

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA MECÂNICA**

**UMA METODOLOGIA PARA INTEGRAÇÃO
CAD/CAPP/CAM VOLTADA PARA MANUFATURA
REMOTA DE PEÇAS ROTACIONAIS BASEADA NA
INTERNET**

Tese submetida à

UNIVERSIDADE FEDERAL DE SANTA CATARINA

para a obtenção do grau de

DOUTOR EM ENGENHARIA MECÂNICA

ALBERTO JOSÉ ÁLVARES

Florianópolis, outubro de 2005

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA MECÂNICA**

**UMA METODOLOGIA PARA INTEGRAÇÃO
CAD/CAPP/CAM VOLTADA PARA MANUFATURA REMOTA
DE PEÇAS ROTACIONAIS BASEADA NA INTERNET**

ALBERTO JOSÉ ÁLVARES

Esta tese foi julgada adequada para a obtenção do título de

**DOUTOR EM ENGENHARIA
ESPECIALIDADE ENGENHARIA MECÂNICA**

sendo aprovada em sua forma final.

Prof. João Carlos Espíndola Ferreira, Ph.D. - Orientador

Prof. José A. Bellini da Cunha Neto, Dr. - Coordenador do Curso

BANCA EXAMINADORA

Prof. João Carlos Espíndola Ferreira, Ph.D. - Presidente

Prof. Lourival Boehs, Dr. Eng.

Prof. Ricardo José Rabelo, Dr.

Prof. Marcelo Teixeira dos Santos, Dr.

Prof. Osiris Canciglieri Junior, Ph.D.

*À minha amada Lillian e aos meus maravilhosos
filhos Manuella, Carolina, Marcos Estevam, João
Alberto e Luis Felipe.*

Agradecimentos

Ao Departamento de Engenharia Mecânica da UFSC por oferecer as condições necessárias para elaboração desta tese.

À Universidade de Brasília, Faculdade de Tecnologia e ao Departamento de Engenharia Mecânica por terem aprovado meu afastamento, o que viabilizou minha dedicação integral ao doutoramento.

Ao Prof. João Carlos Espíndola Ferreira pela orientação e apoio.

Aos Bolsistas do Graco, Magno e Jones, e do Grima, Roman, pela valiosa colaboração na implementação do sistema WebMachining.

À MANET (Rede de Automação da Manufatura) pelo apoio oferecido para elaboração deste trabalho.

Ao IFM (Instituto Fábrica do Milênio) e ao CNPq pela disponibilização de três bolsas de iniciação científica.

Abstract

This thesis describes a methodology for CAD/CAPP/CAM integration for the remote manufacture of cylindrical parts using Internet, especially through the protocols associated with the World Wide Web. This methodology proposed specifies the modules of the system, and how these modules interact with each other. Described within this work is the architecture for implementing the integrated system CAD/CAPP/CAM applied to the remote manufacture of cylindrical parts, consisting of a group of specifications, techniques and algorithms used for the functional definition of the modules of the system.

This work presents a description of the implementation of the WebMachining methodology and system (<http://WebMachining.AlvaresTech.com>) developed in a context of e-Mfg and Concurrent Engineering, aiming at integrating CAD/CAPP/CAM for the remote manufacturing of cylindrical parts through the Internet.

The methodology and its implementation are conceived starting from the modelling paradigm based on synthesis of design features, in order to allow the integration among collaborative design (WebCADbyFeatures), generative process planning (WebCAPP) and manufacturing (WebTurning).

The system is implemented in a distributed environment of agents (agents' community), whose architecture is stratified in three layers: design, Process Planning and Production; being adopted Knowledge Query and Manipulation Language (KQML) as the language pattern of messages among the design, process planning and production agents.

In this work a system will be presented, which was developed for the collaborative design of feature-based cylindrical parts through the Internet, integrated with a CAD/CAPP/CAM system, using an approach based on multi-agent systems. The information referring to the features is manipulated through a relational database management system. The Graphic User Interface (GUI) is implemented in Java and HTML.

The main objective of this research is to develop a methodology and the implementation of a system CAD/CAPP/CAM integrated and intelligent, oriented to production of cylindrical parts through Web. The systems developed starting from this methodology can be applied in the industry and in the academy. In academic applications it can be used in distance teaching in a context of remote laboratories. In industrial applications it can be used as part of a service of rapid prototyping for try-out of parts or for the supply of functional prototype for the own company or remote customers in a context of telemanufacturing.

The life cycle of the development of the system WebMachining is based on objects-oriented programming. It was also used the modelling methodology IDEF (IDEF0 and IDEF1X) associated to UML, for designing and implementation of the methodology of the system WebMachining.

Resumo

Esta tese de doutorado apresenta uma metodologia para integração CAD/CAPP/CAM voltada para manufatura remota de peças rotacionais via Internet, em especial os protocolos associados à *World Wide Web*, bem como a implementação de um sistema computacional baseado na arquitetura proposta, que permitiu a validação da metodologia concebida. Esta metodologia, denominada WebMachining (<http://WebMachining.AlvaresTech.com>), descreve uma arquitetura para implementação do sistema integrado CAD/CAPP/CAM via Web, definindo um arcabouço para um sistema integrado CAD/CAPP/CAM via Internet.

A metodologia é concebida a partir do paradigma de modelagem colaborativa baseada em síntese por *features* de projeto, a fim de permitir a integração das atividades de projeto detalhado colaborativo, planejamento do processo e manufatura. Utiliza-se um ambiente distribuído de agentes de softwares interoperáveis denominado de Comunidade de Agentes de Manufatura, sendo sua arquitetura estratificada em três camadas: Projeto, Planejamento do Processo e Fabricação. O fluxo de informações é realizado através do encapsulamento das informações, adotando-se o *Knowledge Query and Manipulation Language* (KQML) como linguagem de mensagens entre os agentes.

O objetivo principal desta tese é desenvolver uma metodologia e a implementação de um sistema CAD/CAPP/CAM integrado e inteligente, voltado para fabricação de peças rotacionais via Web. O sistema é inédito no que concerne à abordagem de projeto colaborativo por síntese de *features* e no processo de fabricação utilizado, sendo voltados para fabricação de peças rotacionais simétricas (*features* concêntricas) e assimétricas (*features* não-concêntricas). Inova também na incorporação de funcionalidades associada à teleoperação via Web da máquina-ferramenta CNC permitindo a integração CAD/CAPP/CAM no chão-de-fábrica. O sistema CAPP concebido e implementado utiliza uma nova abordagem de mapeamento de *features* de projeto para *features* de torneamento e usinagem, denominada de “decomposição orientada ao *setup*, geometria e operação”, sendo aderente ao STEP-NC, usando uma estrutura de dados semelhante e trabalhando com planos de processos não-lineares.

Os sistemas desenvolvidos a partir desta metodologia poderão ser aplicados na indústria e na academia. Em aplicações acadêmicas poderão ser utilizados em ensino a distância em um contexto de laboratórios remotos. Em aplicações industriais poderão ser utilizados como parte de um serviço de prototipagem rápida para *try-out* de peças ou para o fornecimento de protótipo funcional para a própria empresa ou clientes remotos em um ambiente de telemanufatura; podendo também ser integrado ao sistema de Gestão da Produção da empresa, fornecendo planos alternativos de processos para o sistema de programação da produção em "tempo real".

O ciclo de vida do desenvolvimento do sistema WebMachining é baseado em técnicas de programação orientada a objetos. Utilizou-se também a metodologia de modelagem IDEF (IDEF0 e IDEF1X) associada à UML, para projeto e implementação da metodologia do sistema WebMachining.

Sumário

ABSTRACT	v
RESUMO	vi
LISTA DE TABELAS	xiv
LISTA DE FIGURAS	xv
LISTA DE SÍMBOLOS E ABREVIATURAS	xxii
Capítulo 1: Introdução	1
1.1 Resumo Orientativo da Tese	2
1.1.1 Questão a Responder	2
1.1.2 Hipótese a Comprovar	2
1.1.3 Objetivos	2
1.1.4 Contexto e Condições de Contorno	3
1.1.5 Enfoque	3
1.2 Formulação do Problema	3
1.3 Oportunidade do Tema	4
1.4 Contribuições do Trabalho	5
1.5 Estrutura do Documento	6
Capítulo 2: Manufatura Eletrônica (<i>e-Mfg</i>)	7
2.1 Manufatura Eletrônica (<i>e-Mfg</i>)	7
2.2 Manufatura Remota	8
2.3 Manufatura Virtual	8
2.4 Trabalho Colaborativo	9
2.4.1 Realidade Virtual	9
2.4.2 Laboratório Virtual	9
2.5 Teleoperação	10
2.6 Teleoperação e o CAM	10
2.7 Taxonomia: Laboratórios Virtuais e Remotos	11
2.8 Infra-estrutura de Rede de Alta Velocidade	11

2.9	WWW (World Wide Web) e SOA	12
-----	--------------------------------------	----

Capítulo 3: Revisão de Literatura: Planejamento do Processo **13**

3.1	Projeto Lógico do Plano de Processo	13
3.2	Planejamento do Roteamento - Macroplanejamento	15
3.3	Planejamento das Operações de Usinagem - Microplanejamento	15
3.4	Determinação do Tipo de Operação	15
3.4.1	Capacidade do Processo e Seus Limites	16
3.4.2	Método para Seleção das Operações de Corte X Tabelas de Roteamento	16
3.4.3	Determinação de Sobremateriais de Usinagem	17
3.4.4	Determinação de Condições de Usinagem e Tempos Padrão	17
3.5	Fixação em um Torno	18
3.6	Planejamento do Processo com Alternativas	19

Capítulo 4: Revisão de Literatura: CAPP e Projeto por *Features* **21**

4.1	Abordagens de Sistemas CAPP	21
4.1.1	Variante	22
4.1.2	Generativo	22
4.1.3	Semi-generativo ou Híbrido	23
4.2	Integração Projeto e Manufatura Baseada em <i>Features</i>	23
4.2.1	O Elo de Ligação Entre as Atividades de Projeto e Manufatura	24
4.2.2	Reconhecimento de <i>Features</i>	25
4.2.3	Mapeamento de <i>Features</i>	25
4.2.3.1	Projeto com <i>Features</i> de Forma	25
4.2.3.2	Projeto com <i>Features</i> de Manufatura	27
4.3	Projeto Por <i>Features</i>	28
4.3.1	ISO10303-224: Definição de Produto para Planejamento de Processo	29
4.3.2	ISO10303-48: <i>Features</i> de Forma	30
4.4	Aderência à Norma ISO 14649 (STEP-NC)	31
4.5	Sistemas CAPP Voltados Para Operações de Torneamento	32
4.6	Outros Trabalhos Associados ao Planejamento de Processos	34
4.6.1	Estratégias de Movimentação de Ferramentas e Superfícies Intermediárias	34
4.6.2	Seleção da Seqüência de Operações de Usinagem	35
4.7	Métodos e Ferramentas Para CAPP	37
4.7.1	Banco de Dados	38
4.7.2	Inteligência Artificial	38
4.7.3	Agentes de Software	39
4.7.3.1	Uma Noção Fraca para Agentes	40
4.7.3.2	Uma Noção Forte para Agentes	40
4.7.4	Arquitetura de Agentes	40
4.7.5	Tipologia de Agentes	41

4.7.6	Linguagens de Agentes	42
4.7.7	Objetos e Agentes	44
4.7.8	O Desenvolvimento de Sistemas de Agentes (ou Multiagentes)	44
4.7.9	Qual Modelo de Tomada de Decisão Utilizar ?	44
Capítulo 5: Metodologias e Sistemas CAD/CAPP/CAM Baseados na Web		46
5.1	Introdução	46
5.1.1	CAD/CAM Colaborativo	46
5.1.2	CAD Colaborativos	48
5.2	Ferramentas CAD com Área de Trabalho Compartilhada	49
5.2.1	WebSpiff (http://www.webspiff.org)	52
5.2.2	WebCAD3D e CyberCut	52
Capítulo 6: WebMachining: Integração CAD/CAPP/CAM Baseada na Web		54
6.1	Metodologia WebMachining	54
6.2	Arquitetura Multiagente	56
6.3	Sistema WebMachining: Modelagem IDEF0	59
6.3.1	Atividade WebCADbyFeatures	61
6.3.2	Atividade WebCAPP	67
6.3.2.1	Abordagem Para Planejamento do Processo	68
6.3.2.2	Metodologia CAPP Para Peças Rotacionais	69
6.3.2.3	Mapeamento de Features de Projeto para Features de Manufatura	71
6.3.2.4	Operações de Usinagem	73
6.3.2.5	Seqüência de Operações de Usinagem	74
6.3.2.6	Determinação das Condições de Usinagem	75
6.3.2.7	Geração do Programa NC	76
6.3.3	WebTurning: CAM Execução	76
6.4	Modelagem IDEF1X	78
Capítulo 7: Interface CAD-CAPP: CAPP Baseado em Mapeamento de Features		82
7.1	Mapeamento de Features	82
7.1.1	Decomposição Orientada ao Setup	85
7.1.2	Decomposição Orientada à Geometria	88
7.1.3	Decomposição Orientada à Operação	89
7.2	Determinação das Operações de Usinagem Associadas às Features: Workingsteps	91
7.2.1	Grafos E/Ou Para Operações de Desbaste	95
7.2.2	Grafos E/Ou Para Operações de Acabamento e Semi-acabamento	100
7.3	Representação do Plano de Operações Não-Linear por Features de Usinagem: Workingsteps	102

7.4	Determinação da Seqüência de Operações: <i>Workplan</i>	103
7.5	Estratégias para Geração de Trajetórias de Ferramentas	108
7.5.1	Faceamento: Desbaste	111
7.5.2	Torneamento Externo e Interno: Desbaste da Direita para Esquerda (Ferramenta para Esquerda)	113
7.5.3	Torneamento Externo e Interno de Reentrância com Superfície Intermediária à Direita: Desbaste da Direita para Esquerda	114
7.5.4	Torneamento Externo e Interno de Reentrância (Canal Largo) com Superfície Intermediária Triangular à Direita: Desbaste Esquerda para Direita	116
7.5.5	Torneamento Externo e Interno de Reentrância (Canal Largo) com Superfície Cônica à Esquerda: Desbaste Direita para Esquerda	117
7.5.6	Torneamento Externo e Interno de Reentrância (Canal Largo) com Superfície Circular (Côncava ou Convexa) à Esquerda: Desbaste Direita para Esquerda	118
7.5.7	Torneamento Externo e Interno de Ranhuras: Desbaste Canal Estreito	120
7.5.8	Torneamento Externo e Interno de <i>Features</i> de Transição: Desbaste Chanfros, Arredondamentos e Filetes: Usinagem Direita para Esquerda	120
7.5.9	Operações de Acabamento: Perfilamento Externo e Interno de <i>Features</i> e Outras Operações	121
7.5.10	Operações de Fresamento e Furação: Eixo C	122
7.6	Determinação das Ferramentas de Corte (Insertos e Suportes)	122
7.7	Modelo de Tempos Para Determinação das Condições de Corte Otimizadas Para Operações de Usinagem	126
7.7.1	Formulação do Problema	128
7.7.2	Modelos Matemáticos para Cálculo de Tempos de Usinagem	128
7.7.3	Metodologia de Otimização	129
7.7.4	Algoritmo Genético	131
7.8	Linearização do Plano de Processos com Alternativas	133
7.8.1	Metodologia de Otimização	134
7.8.2	Algoritmo Genético	136
7.9	Geração do Programa NC (ISO 6983)	138
7.10	Geração da Documentação Final do Plano de Processo Linearizado	140

Capítulo 8: Abordagem Lógica Para CAPP: Métodos e Algoritmos Concebidos **142**

8.1	Peça Exemplo: Estudo de Caso	142
8.2	Mapeamento de <i>Features</i> de Projeto para <i>Features</i> de Usinagem	142
8.2.1	Decomposição Orientada ao Setup	145
8.2.2	Decomposição Orientada à Geometria	147
8.2.3	Decomposição Orientada à Operações	151
8.3	Determinação dos <i>Workingsteps</i> de Usinagem e de Torneamento com Alternativas	153

8.3.1	Macroplanejamento: Algoritmos e Estruturas para Determinação do Roteamento	156
8.3.2	Microplanejamento: Algoritmos e Estruturas para Determinação das Operações de Usinagem	158
8.3.3	Montagem dos Turning_workingsteps e Machining_workingsteps	158
8.4	Determinação e Representação da Sequências de Operações por Setup	158
8.5	Estratégias para Geração de Trajetórias de Ferramentas	161
8.5.1	Primeiro Setup: Setup1	163
8.5.2	Segundo Setup: Setup2	164
8.6	Determinação das Ferramentas de Corte	164
8.6.1	Ferramentas Para Torneamento Geral, Sangramento e Rosqueamento	164
8.7	Determinação dos Modelos de Tempos e Custos	167
8.8	Determinação das Condições de Usinagem com Algoritmos Genéticos	169
8.8.1	Determinação das Propriedades, Coeficientes e Constantes do Modelo de Otimização	169
8.8.2	Primeiro Estágio	170
8.8.3	Segundo Estágio	173
8.8.4	População Inicial: n Passes de Desbaste + 1 Passe Acabamento	175
8.9	Linearização do Plano de Processos com Alternativas (Workplan)	177
8.10	Geração do Código G e Plano de Processo	180

Capítulo 9: Projeto Colaborativo: WebCADbyFeatures 182

9.1	Introdução	182
9.2	Arquitetura Multiagente Para Projeto Colaborativo	183
9.3	Modelagem da Informação	184
9.3.1	WebCADbyFeatures: Modelagem Conceitual da Informação	184
9.3.2	WebCADbyFeatures: Modelagem Física - MySQL	185
9.4	Sistema Computacional WebCADbyFeatures	185
9.5	Modelagem Colaborativa	191
9.6	Discussão	193

Capítulo 10: Planejamento do Processo Não-Linear: WebCAPP 197

10.1	Introdução	197
10.2	Implementação Computacional do WebCAPP	199
10.2.1	Via Servlet	199
10.2.2	Via Applet	199
10.3	Discussão	203

Capítulo 11: Teleoperação Centro de Torneamento: WebTurning 206

11.1	Introdução	206
------	----------------------	-----

11.2	Arquitetura WebTurning	207
11.3	Interface com o Usuário WebDNC: GUI Cliente-Servidor	209
11.4	Funções FOCAS1/Ethernet Para Teleoperação do Centro de Torneamento	210
11.5	Servidor WebDNC	210
11.6	Servidores de Vídeo e Áudio: WebCam	212
11.7	Discussão	212

Capítulo 12: Estudos de Casos: Peças Exemplos **216**

12.1	Introdução	216
12.2	Peças Jogo de Xadrez	217
12.2.1	WebCADbyFeatures	217
12.2.2	WebCAPP	219
12.2.3	WebTurning	223
12.3	Peça com Tolerâncias	223
12.3.1	WebCADbyFeatures	223
12.3.2	WebCAPP	223
12.3.3	WebTurning	227
12.4	Peça Romi	227
12.4.1	WebCADbyFeatures	227
12.4.2	WebCAPP	227
12.4.3	WebTurning	231
12.5	Peça Complexa	231
12.5.1	WebCADbyFeatures	231
12.5.2	WebCAPP	238
12.5.3	WebTurning	238
12.6	Peça Espelho	239
12.6.1	WebCADbyFeatures	239
12.6.2	WebCAPP	239
12.6.3	WebTurning	242

Capítulo 13: Conclusões **243**

13.1	Contribuições do Trabalho	243
13.2	Implementação Computacional WebMachining	246
13.2.1	WebCADbyFeatures	246
13.2.2	WebCAPP	247
13.2.3	WebTurning	247
13.3	Sugestões para Trabalhos Futuros	248

Apêndice A: Publicações e Projetos Associados à Tese	250
Apêndice B: URLs Associadas à Metodologia WebMachining	252
B.1 Metodologia Para Teleoperação Via Web	252
B.2 Metodologias IDEF	252
B.3 Modelagem Física e Conceitual do Banco de Dados MySQL	252
B.3.1 Modelagem Conceitual	252
B.3.2 Modelagem Física	253
B.4 Biblioteca de <i>Features</i>	253
Apêndice C: Modelos Matemáticos Para Otimização	254
C.1 Modelo Matemático Passe Único para Cálculo de Tempos de Usinagem: Faceamento e Torneamento Cilíndrico, Cônico e Circular	254
C.2 Modelo Matemático Multi-passe para Tempo de Corte de Perfil Contínuo	256
C.3 Modelo Matemático Multi-passe para Tempo de Corte de Torneamento Longitudinal	260
C.4 Modelo Matemático Multi-passe para Tempo de Corte de Sangramento, Corte de Peças e Faceamento	261
C.5 Modelo Matemático Passe Único para Tempo de Corte de Recartilhamento	262
C.6 Modelo Matemático Passe Único para Tempo de Corte de Furação e Alargamento . .	262
C.7 Modelo Matemático Passe Único para Tempo de Corte de Rosqueamento Cilíndrico .	263
C.8 Modelos de Otimização de Usinagem Propostos	263
C.8.1 Modelo Baseado no Mínimo Custo	263
C.8.2 Modelo Baseado na Máxima Taxa de Produção	266
C.8.3 Restrições Práticas de Usinagem	267
Apêndice D: Ciclo de Vida do Software WebMachining Modelado em UML	270
D.1 Introdução	270
D.2 WebCADbyFeatures	271
D.2.1 Use-Case Diagram	271
D.2.2 Activity Diagram	272
D.2.3 Modelo Estático	272
D.2.4 Modelo Dinâmico	272
D.2.5 Implementação do WebCADbyFeatures	276
D.3 WebCAPP	277
D.3.1 Decomposição Orientada ao Setup	277
D.3.2 Decomposição Orientada à Geometria	278
D.3.3 Decomposição Orientada à Operação	278
D.3.4 Geração do Código G	280
D.3.5 Modelo Estático: <i>Class Diagrams</i>	280
D.4 WebTurning	280

Lista de Tabelas

4.1	Troca de mensagens utilizando KQML.	43
7.1	Operações de usinagem modeladas na decomposição orientada à operação.	90
7.2	Classificação das <i>features</i> reconhecidas pelo método.	93
7.3	Tabela montagem cromossomo associado ao algoritmo genético.	132
8.1	Tabela <i>Workingsteps</i> de torneamento e usinagem (W_{ID}).	155
8.3	Operações de usinagem modeladas na decomposição orientada à operação.	165
8.4	Condições de usinagem para passe simples em operações de acabamento, considerando um comprimento de corte de 60,1 mm (L).	171
8.5	Condições de usinagem para passe simples em operações de desbaste, considerando um comprimento de corte de 60,1 mm (L).	172
8.6	Faixa de avanços máximos recomendado pela SANDVIK (2003) em relação ao raio de arredondamento da ponta do inserto.	172
8.7	Possibilidades de a_{ps} e a_{pr} para a peça exemplo: $a_{pr} = \frac{12,5 - a_{ps}}{4}$	175
8.8	População inicial de cromossomos, função <i>fitness</i> , valor esperado e valor real baseado no método da roleta (probabilidade de um indivíduo ser melhor que outro).	176
8.9	Operadores de reprodução e <i>crossover</i> com probabilidade de 80%.	176
8.10	Operador de mutação com probabilidade de 80%.	177
8.11	Tabela <i>Workingsteps</i> para linearização <i>Workplan</i> (W_{ID}).	178
8.12	Matriz adjacente relativa ao tempo de troca de <i>setup</i> entre <i>Workingsteps</i> (W_{ID}).	179
12.1	<i>Features</i> presentes nas peças estudadas.	217
12.2	Ferramentas Sandvik disponíveis no Galaxy 15M.	217
13.1	Síntese das contribuições da tese.	244
13.2	Especificações, funcionalidades e requisitos do sistema WebMachining.	246
D.1	<i>Use-cases</i> para os três sub-sistemas do ambiente integrado WebMachining.	271

Lista de Figuras

3.1	Tipos de placas utilizadas (HUANG, 1988): estilo 1, 2, 3 e 4.	19
4.1	Abordagem de criação de <i>features</i> utilizando projeto por <i>features</i> (SHAH e MANTYLA, 1995).	29
4.2	Estrutura de modelo de dados para STEP-NC representada por um diagrama Express_G (ISO 14649 - PART 1, 2003).	32
6.1	Camadas da arquitetura multiagente MAC.	56
6.2	Arquitetura Multiagente WebMachining.	58
6.3	Diagrama IDEF0: WebMachining nível A0.	60
6.4	Diagrama IDEF0: principais atividades do sistema WebMachining.	62
6.5	Diagrama IDEF0: Atividades WebCADbyFeatures.	63
6.6	Componentes GUI.	65
6.7	Taxonomia de <i>features</i> de forma utilizada (CAM-I, 1986).	65
6.8	Diagrama de classes para <i>Projeto</i>	66
6.9	Diagrama IDEF0: Atividades CAPP.	70
6.10	Diagrama IDEF0: sub-atividades do WebCAPP associadas ao mapeamento de <i>features</i> de projeto para <i>features</i> de manufatura.	72
6.11	Árvores de decisão para <i>features</i> concêntricas baseada em HUANG (1988).	75
6.12	Diagrama IDEF0: Atividades CAM.	77
6.13	Modelagem da informação de algumas <i>features</i> de forma utilizadas através da metodologia IDEF1X.	79
6.14	Diagrama de classes para <i>features</i> de tolerâncias.	80
6.15	Diagrama de classes para tecnologia de usinagem (recursos de manufatura).	81
7.1	Grafo e/ou <i>setup</i> : alternativas para <i>setup</i> da peça quando $\frac{L_{peça}}{D_{peça}} > 4$	86
7.2	Grafo e/ou <i>setup</i> : alternativas para <i>setup</i> da peça quando $\frac{L_{peça}}{D_{peça}} < 2$	87
7.3	Grafo e/ou <i>setup</i> : alternativas para <i>setup</i> da peça quando $2 \leq \frac{L_{peça}}{D_{peça}} \leq 4$	87
7.4	Poliforma fechada representando o perfil 2D da peça bruta e peça acabada.	90
7.5	(a) Ferramentas: direita para esquerda, esquerda para direita e neutra (CHANG <i>et al.</i> , 1998) (b) Operações mais usuais em peças de revolução (MAZIERO e NASCIMENTO, 2003) (c) métodos de usinagem de canal/reentrância (YANG e LEE, 1998).	92

7.6	Um exemplo do procedimento associado a uma árvores de decisão (HUANG, 1988), apresentando alternativas de volumes e operações de usinagem para <i>features</i> concêntricas.	92
7.7	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo OD (Diâmetro Externo) cilíndrica/retilínea.	95
7.8	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo OD (Diâmetro Externo) cônica para dentro (<i>cut-in</i>).	96
7.9	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo OD (Diâmetro Externo) cônica para fora (<i>cut-out</i>).	96
7.10	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo canal estreito.	96
7.11	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo canal largo, reentrância, com as duas faces perpendiculares.	97
7.12	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo canal largo, reentrância, com uma face perpendicular à esquerda e uma superfície cônica à direita, para dentro da peça (<i>cut-in</i>).	98
7.13	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo canal largo, reentrância, com uma face perpendicular à direita e uma superfície cônica à esquerda, para fora da peça (<i>cut-out</i>).	98
7.14	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo face.	99
7.15	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo chanfro.	99
7.16	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo filete ou arredondamento.	99
7.17	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> ID.	100
7.18	Grafo E/OU apresentando alternativas de usinagem para perfil com <i>features</i> OD e ID cilíndricas.	101
7.19	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo Perfil Externo para fora e para dentro com reentrância (<i>cut-in</i>).	101
7.20	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo Perfil Interno para fora e para dentro com reentrância.	101
7.21	Grafo E/OU apresentando alternativas de usinagem para <i>feature</i> do tipo perfil para fora (<i>cut-out</i>) e para dentro (<i>cut-in</i>) com filetes para superfícies internas e externas.	102
7.22	Exemplo de peça STEP-NC (2003): (a) Geometria da peça (b) lista de <i>workingstep</i> , (c) grafo E/OU, (d) lista da estrutura de dados para operações de usinagem.	104
7.23	(a) Relação de <i>features</i> de usinagem por <i>setup</i> para operações de desbaste, semi-acabamento e acabamento (b) Tabela de relacionamentos por <i>setup</i> mostrando a precedência de <i>features</i> de usinagem (c) Matriz Adjacente por <i>setup</i>	105
7.24	Opções de <i>Workingsteps</i> : <i>features</i> de usinagem, operações de usinagem representados no formato de Matriz Operações Teóricas: operação, tempo e custo teórico da operação (TP), precedência (prioridade) e seqüência final de execução de <i>workingstep</i> .	107

7.25	Geração lógica do caminho de ferramenta para torneamento (SHAH e MANTYLA, 1995) destacando estratégias para movimentação de ferramenta da esquerda para direita (1, 2, 3, 8 e 9) e da direita para esquerda (4 e 5) nas operações de desbaste e de acabamento (perfilamento).	110
7.26	Trajectoria de ferramenta para perfil cônico e convexo: corte paralelo x corte bi-axial.	118
7.27	Caminho de ferramenta para desbaste de ranhura.	120
7.28	Caminho da ferramenta para desbaste de <i>features</i> de transição: chanfro, arredondamento e filete, respectivamente.	120
7.29	Caminho da ferramenta para acabamento: operação de perfilamento (movimento bi-axial) com ferramenta para esquerda e torneamento com ferramenta para direita (direção do avanço).	121
7.30	Modelo de usinagem multi-passe (CHEN e SU, 1998): (a) trajetória da ferramenta para dois estágios de desbaste e um passe de acabamento (b) Modelo interpolação circular (c) Modelo interpolação linear.	129
7.31	ToolBox Algoritmo Genético do MatLab, utilizado na determinação das condições de usinagem otimizadas.	133
7.32	Entidades associadas à representação de planos de processos não-lineares segundo STEP-NC (Part 10), em Express-G.	134
7.33	<i>Workplans</i> alternativos e <i>workingsteps</i> associados, onde os valores das células estão associados ao custo ou tempo relativo à operação de usinagem.	136
7.34	Exemplo: (a) peça (b) <i>features</i> de usinagem e operações (c) grafo de precedência de <i>workingsteps</i> (d) matriz de <i>setup</i> de <i>workingstep</i> (e) <i>strings</i> (f) operador <i>crossover</i> (g) operador de mutação.	137
7.35	Documentação final do plano de processo linearizado em XML.	141
8.1	Peça exemplo em 2D e 3D sendo constituída por 39 <i>features</i> de forma do tipo OD, ID, canal, face, arredondamento e chanfro, modelado pelo sistema WebCADbyFeatures.	143
8.2	Posicionamento das linhas delimitadoras (LDs) em antes de depois das <i>features</i> F6 e F39 e valor dos pesos de F1-F6.	150
8.3	Roteamento para o <i>Setup1</i> e <i>Setup2</i> da peça exemplo, relacionando as <i>features</i> de projeto (F1 - F39), sendo representado na ordem inversa.	152
8.4	Volumes de usinagem associados à peça exemplo.	154
8.5	Grafo E/OU de <i>workingsteps</i> associado à peça exemplo.	156
8.6	Grafo de precedência de <i>features</i> de usinagem associado aos <i>Workingsteps</i>	161
8.7	Definição Zero Peça (ZP) no <i>Setup1</i> , peça exemplo.	163
8.8	Chaves de código do porta-ferramenta DDJNR/L (SANDVIK, 2003).	165
8.9	Páginas HTML da Sandvik utilizadas para pesquisar as condições de usinagem para operações de acabamento e desbaste.	170
8.10	Código G gerado para a peça exemplo.	181

9.1	Janela principal do WebCADbyFeatures, apresentando o perfil de uma peça rotacional em 2D. A peça é constituída por 29 features, incluindo features do tipo OD, ID, Face, Groover, Splines, entre outras. É apresentado sólido VRML correspondente, gerado pelo sistema.	183
9.2	Resumo da modelagem UML da biblioteca de features.	185
9.3	As três camadas do acesso ao banco de dados.	186
9.4	Arquitetura computacional WebMachining: módulos do sistema WebCADbyFeatures.	186
9.5	Acesso ao sistema WebCADbyFeatures (http://WebMachining.AlvaresTech.com).	188
9.6	Etapas da geração da peça no CAD: a) Janela de Opções (Options); b) Janela de dados de um novo projeto (New Project); c) Janela de dados do blank barra sólida (SolidBar); d) Janela de opções de modelagem (Modelling Options).	189
9.7	Modelagem da feature de Eixo C padrão de furos axial.	190
9.8	Interface WebCADbyFeatures: 2D e VRML.	191
9.9	Inteface do cliente para projeto colaborativo: "WebCADbyFeatures Collaborative Design IPLayre Router Client", apresentando o painel "Register".	192
9.10	Interface do cliente para projeto colaborativo: Request do Agent de Interface "Alvares" no servidor Router do ambiente JATLite.	192
9.11	Modelagem colaborativa entre os agentes "Alvares" e "jones". Agente "Alvares" envia modelo por features para agente "jones".	193
9.12	Modelagem colaborativa entre os agentes "Alvares" e "jones". Agente "Alvares" recebe o modelo por features do agente "jones", após efetuar as alterações no modelo de features.	194
10.1	Normalização das features de projeto.	198
10.2	Decomposição orientada à geometria.	198
10.3	Plano de processo linearizado e programa NC gerado pelo WebCAPPServlet em XML.	200
10.4	Simulação do programa NC capturado pelo sistema de processamento de imagens WebCam do sistema WebDNC.	200
10.5	Resultado da atividade de decomposição orientada ao setup, geometria e operação gerado pelo sistema WebCAPPApplet, sendo a peça constituída por 29 features.	201
10.6	a) Lista de Workingsteps; b) grafo E/OU.	202
10.7	Programa NC gerado pelo sistema WebCAPPApplet.	202
11.1	Arquitetura detalhada: módulos do sistema WebTurning.	207
11.2	Hardware: placa ethernet, protocolo e API FOCAS1/DNC1 (Fanuc, 2003).	208
11.3	Biblioteca DLL FOCAS1/DNC1:fluxo de dados (Fanuc, 2003).	208
11.4	WebDNC: teleoperação e monitoramento remoto do centro de torneamento Galaxy 15M (http://WebDNC.graco.unb.br).	209
11.5	Arquitetura, fluxo de dados e algumas funções disponibilizadas pelo WebTurning.	211
11.6	Sistema de monitoração por imagem WebCam (NetCam), constituído por quatro applets, que monitoram a tela do CNC, o interior da máquina e o exterior.	213

12.1	Modelagem torre utilizando <i>feature</i> Eixo C e o VRML associado às <i>features</i> concêntricas.	218
12.2	Modelagem bispo utilizando <i>feature ODconvex</i> e o VRML associado às <i>features</i> concêntricas.	218
12.3	Modelagem do cavalo utilizando <i>feature</i> Eixo C e o VRML associado às <i>features</i> concêntricas.	219
12.4	Modelagem do peão utilizando <i>feature spline</i> e o VRML associado às <i>features</i> concêntricas.	220
12.5	Modelagem rainha e duas visões do VRML associado às <i>features</i> concêntricas.	220
12.6	Modelagem do rei e VRML associado às <i>features</i> concêntricas.	221
12.7	Procedimento de normalização realizado pelo WebCAPP identificando as <i>features</i> de projeto que serão mapeadas.	222
12.8	Decomposição orientada ao <i>setup</i> com a identificação das <i>features</i> de fixação, superfícies delimitadoras e <i>features</i> de Eixo C.	223
12.9	Decomposição orientada à geometria.	224
12.10	Decomposição orientada à operação.	225
12.11	Programa NC gerado para a peça torre.	226
12.12	Simulação dos programas NC no CNC do Galaxy 15M para as peças rei e torre.	227
12.13	Simulação dos programas NC no CNC do Galaxy 15M para as peças bispo, cavalo, peão e rainha.	228
12.14	GUI WebDNC mostrando a fabricação da torre e realizando o <i>upload</i> do arquivo NC da peça rei para o Galaxy 15M. Também são apresentadas simulações e imagens da usinagem.	229
12.15	GUI WebDNC mostrando a usinagem do bispo e do cavalo.	229
12.16	GUI WebDNC mostrando a usinagem do peão e da rainha e telas associadas ao <i>download</i> e <i>upload</i> de programas NC.	230
12.17	Peças do jogo de xadrez usinadas pelo Galaxy 15M.	230
12.18	Detalhes das peças torre e rainha.	231
12.19	Modelagem de peça com <i>features</i> de tolerâncias de cilindricidade e <i>features</i> de referências <i>A</i> e <i>M</i> e os VRMLs associados.	232
12.20	Simulação dos programas NC no CNC do Galaxy 15M para a peça tolerância no <i>Setup1</i> e <i>Setup2</i>	233
12.21	GUI WebDNC realizando a monitoração da usinagem da peça tolerância.	233
12.22	<i>Features</i> externas da peça tolerância.	234
12.23	<i>Features</i> internas da peça tolerância, lado esquerdo e lado direito.	234
12.24	Modelagem da peça romi constituída por <i>features</i> concêntricas e não-concêntricas, e os modelos VRMLs associados destacando-se a rosca métrica e os canais.	235
12.25	Simulação dos programas NC no CNC do Galaxy 15M para a peça romi no <i>Setup1</i> e <i>Setup2</i>	235
12.26	GUI WebDNC realizando a monitoração da usinagem da peça romi.	236
12.27	Detalhes <i>features</i> internas e externas da peça romi.	236
12.28	<i>Features</i> externas (lado esquerdo e direito) da peça romi.	237

12.29	Modelagem da peça complexa com <i>features</i> de Eixo C, padrão de furos Axial, associada à <i>feature</i> de face F18.	237
12.30	Modelagem VRML da peça complexa apresentando as <i>features</i> concêntricas em <i>wire-frame</i> e isométrica.	238
12.31	Simulação dos programas NC no CNC do Galaxy 15M para a peça complexa no <i>Setup1</i> e <i>Setup2</i>	238
12.32	GUI WebDNC realizando a monitoração da usinagem da peça complexa.	239
12.33	<i>Features</i> externas e internas da peça complexa.	240
12.34	Modelagem da peça espelho hiperbólico aproximada por <i>spline</i> e o VRML associado.	240
12.35	Simulação do programa NC no CNC do Galaxy 15M para a peça espelho hiperbólico modelada por <i>spline</i>	241
12.36	GUI WebDNC realizando a monitoração da usinagem da peça espelho.	241
12.37	Peça espelho usinada.	242
C.1	Modelo de trajetória de ferramenta para sangramento (SANDVIK, 2003).	262
D.1	Diagrama Use-Case do WebCADbyFeatures.	272
D.2	Diagrama de Atividade do WebCADbyFeatures.	273
D.3	Diagrama de Atividade do projeto colaborativo.	273
D.4	Diagrama Estático WebCADbyFeatures.	274
D.5	Diagrama de classe (modelo estático) para a classe <i>ODStraight</i> (super classe <i>Feature</i> e pacote <i>features</i>) e seu relacionamento com as demais classes e pacotes WebCADbyFeatures.	274
D.6	Modelo estático para a classe <i>CADAgentFrame</i> (pacote <i>ProtocolLayer.Example.CAD</i>) associado ao <i>JATLite</i> (projeto colaborativo, agente de interface) e seu relacionamento com as demais classes e pacotes WebCADbyFeatures.	275
D.7	Diagrama de Seqüência para use-case de inserção de uma <i>feature</i> qualquer do WebCADbyFeatures.	276
D.8	<i>Component and Deployment Diagrams</i> do WebCADbyFeatures.	276
D.9	<i>Use-Case</i> do WebCAPP.	277
D.10	<i>Use-Case</i> decomposição orientada ao <i>setup</i>	278
D.11	<i>Use-case</i> decomposição orientada à geometria.	279
D.12	<i>Use-case</i> decomposição orientada à operação.	279
D.13	Diagrama de classe principal (pacotes) associado ao módulo WebCAPP e relacionamento com demais pacotes.	281
D.14	Diagrama de classe <i>Workingstep</i> que é composto pelas classes <i>FeatureDeUsinagem</i> e <i>OperaçãoDeUsinagem</i>	282
D.15	Diagrama de classe <i>FeatureDeUsinagem</i>	282
D.16	Diagrama de classe <i>OperacaoDeUsinagem</i> e seus relacionamentos com as classes <i>FerramentaUtilizada</i> e <i>TecnologiaDeUsinagem</i>	283
D.17	Diagrama de classe <i>CodigoG</i> e seus relacionamentos com as classes <i>FerramentaUtilizada</i> , <i>Maquina</i> e <i>InfoDecomposicaoOrientadaAOperacao</i>	284

- D.18 Diagrama de classe *EncontrandoSuperficiesDelimitadoras* e seus relacionamentos com a classe *InfoDecomposicaoOrientadaAOperacao*. 285
- D.19 Diagrama de classe principal (pacotes) associado ao módulo NetCam e relacionamento com demais pacotes, que permite a visualização de vídeo e imagens em tempo real e vídeo gravado do chão-de-fábrica (GUI NetCAM). 285
- D.20 Modelo estático (diagrama de classes) para a classe *NetCAMAVI* (pacotecom.AlvaresTech.NetCAM) e seu relacionamento com as demais classes e pacotes do módulo WebCam, que permite a gravação de vídeo em formato *.avi*. 286

Lista de Símbolos e Abreviaturas

CAD	Computer Aided Design - Projeto Auxiliado por Computador	1
CAPP	Computer Aided Process Planning - Planejamento de Processo Auxiliado por Computador	1
CAM	Computer Aided Manufacturing - Manufatura Auxiliada por Computador	1
GUI	Graphical User Interface - Interface Gráfica do Usuário	2
CIM	Computer Integrated Manufacturing - Manufatura Integrada por Computador	2
B2B	Business to Business - Negócio para Negócio	2
B2C	Business to Costumer - Negócio para Consumidor	2
WWW	World Wide Web - Rede Mundial de Computadores	2
CNC	Comando Numérico Computadorizado	3
KQML	Knowledge Query and Manipulation Language	3
e-Mfg	Electronic Manufacturing - Manufatura Eletrônica	4
SOA	Service-oriented Architecture - Arquitetura Orientada a Serviço	5
TCP/IP	Transmission Control Protocol/Internet Protocol - Protocolo de Controle de Transmissão/Protocolo Internet	6
IDEF	Integration Definition for Function Modeling	6
IDEF0	Integration DEFinition language 0	6
IDEF1X	Integration Definition for Information Modeling	6
NC	Comando Numérico	8
ERP	Enterprise Resources Planning - Planejamento dos Recursos da Empresa	8
RV	Reality Virtual - Realidade Virtual	9
FTP	File Transfer Protocol - Protocolo de Transferência de Arquivo	10
HTTP	Hypertext Transfer Protocol - Protocolo de Transferência de Hypertexto	10

DNC	Distributed Numeric Control - Comando Numérico Distribuído	11
MRP	Material Resource Planning - Planejamento dos Recursos de Materiais	11
CAPPC/CAP	Planejamento e Controle da Produção Auxiliado por Computador	11
DXF	Data Exchange Format - Formato de Troca de Dados	11
IGES	Initial Graphics Exchange Specification - Especificação Inicial de Troca de Dados Gráficos	11
HTML	Hiper Text Markup Language - Linguagem Simbólica de Hipertexto	11
CGI	Comun Gateway Interface - Interface de Porta Comum	11
VRML	Virtual Reality Modeling Language - Linguagem de Modelagem de Realidade Virtual	12
PNs	Petri Networks - Redes de Petri	20
AI	Artificial Intelligence - Inteligência Artificial	23
STEP	Standard Exchange of Product Model Data - Padrão de Troca de Modelo de Dados de Produto	25
PDES	Product Data Exchange Using STEP	25
CSG/DSG	Constructive/Destructive Solid Geometric - Sólido Geométrico Construtivo/Destrutivo	27
ADDL	Linguagem de Descrição de Projeto	27
FDL	Linguagem de Descrição Funcional	27
PDGL	Linguagem Gráfica para o Projeto de Peças	27
PSG	Process Sequence Graphic - Gráfico de Seqüência de Processo	32
SNLP	Systematic NonLinear Planning - Planejamento Não-Linear Sistemático	37
DAI	Distributed Artificial Intelligence - Inteligência Artificial Distribuída	39
MAS	MultiAgent Systems - Sistemas MultiAgentes	39
ACL	Agent Communication Language - Linguagem de Comunicação de Agentes	40
FIPA-ACL	Foundation for Intelligent Physical Agents - Agents Communication Language	43
KIF	Knowledge Interchange Format	43
FIPA-CLL	FIPA Content Language Library	43
FIPA-SL	FIPA Semantic Language	43
FIPA-DRF	FIPA Resource Description Framework	43

FIPA-CCL	FIPA Constraint Choice Language	43
FIPA-KIF	FIPA Knowledge Interchange Format	43
SQL	Structured Query Language - Linguagem de Pesquisa Estruturada	45
JATLite	Java Agent Template Lite	45
AMR	Agent Message Router	45
Jade	Java Agent DEvelopment Framework	45
CORBA	Common Object Request Broker Architecture	49
API	Application Protocol Interface - Interface de Protocolo de Aplicação	52
V_c	velocidade de corte	128
f	avanço	128
ap	profundidade de corte	128
FOCAS1	Fanuc Open CNC API Specifications	183
VOIP	Voice Over Internet Protocol - Voz sobre Protocolo Internet	196
UML	<i>Unified Modeling Language</i> - Linguagem de Modelagem Unificada	270

Lista de Algoritmos

1	Caminho de ferramenta para faceamento (torneamento transversal).	112
2	Caminho de ferramenta para desbaste torneamento longitudinal da direita para esquerda.	113
3	Algoritmo para determinação do caminho de ferramenta para desbaste de reentrância em torneamento longitudinal da direita para esquerda.	115
4	Algoritmo para determinação do caminho de ferramenta no torneamento, desbaste da esquerda para direita em uma superfície intermediária.	116
5	Programa em Java para construção de blocos em código G.	140
6	Principais passos para implementação do WebCAPP.	143
7	Algoritmo para decomposição orientada ao <i>setup</i>	145
8	Algoritmo para definição do número de fixações da peça.	145
9	Algoritmo para definir candidatas à superfícies de fixação (<i>features</i> internas e externas) para o <i>Setup1</i> e <i>Setup2</i>	146
10	Algoritmo para agrupamento de <i>features</i> no <i>Setup1</i> e <i>Setup2</i>	148
11	Algoritmo para determinação das linhas delimitadoras (LD) e agrupamento de <i>features</i> .	149
12	Algoritmo para detecção de <i>features</i> secundárias, para evitar seu uso como superfície de fixação no <i>Setup2</i>	149
13	Estrutura para algoritmo macroplanejamento.	157
14	Algoritmos para atividade de macroplanejamento de operações de usinagem.	157
15	Estruturas para microplanejamento de operações de usinagem.	159
16	Estruturas para microplanejamento de operações de usinagem.	160
17	Estrutura <i>Workingstep</i>	160
18	Estratégia para movimentação de ferramenta associada aos <i>Workingsteps</i> utilizando ciclos fixos.	163
19	Cálculo de tempos para <i>Workingsteps</i> baseado na área de usinagem da <i>feature</i> de torneamento.	169
20	Fluxograma apresentando algoritmo genérico para otimização das condições de usinagem.	174
21	Fluxograma detalhado para seleção de condições de usinagem para passe simples: a) operações de desbaste b) operações de acabamento.	174
22	Fluxograma determinação de movimentação de ferramenta baseado em ciclo fixo ou movimentação manual.	180

Capítulo 1

Introdução

A globalização tem afetado diretamente as empresas, os consumidores e as economias dos países, provocando a introdução de novos modelos organizacionais, levando em conta a gestão da produção e o fluxo de informação, para diminuir o tempo de resposta em relação à demanda de mercado. O rápido desenvolvimento da Internet tem facilitado a comunicação no âmbito da manufatura, criando novas possibilidades e oportunidades para aumentar a eficiência deste processo, diminuindo o tempo de reação às demandas e oscilações do mercado.

Componentes e produtos variados e de boa qualidade, fabricados em diferentes países, têm sido disponibilizados às pessoas em qualquer parte do mundo, muitas vezes a preços iguais ou inferiores aos produtos nacionais. A Internet tem contribuído significativamente para a globalização, pois ela permite uma comunicação mais rápida entre pessoas e também empresas. Uma prova disto é a quantidade de recursos que têm sido gastos no comércio eletrônico. Analistas da Câmara Brasileira de Comércio Eletrônico (<http://Camara-e.net>) estimam que em 2005 o comércio eletrônico associado ao B2B (*business to business*) nos USA estará em torno de US\$ 1,8 trilhões e o B2C (*business to customer*) no Brasil na ordem de R\$ 9,5 bilhões. Além da venda de itens como carros, jóias e livros através da Internet, esta tecnologia tem sido utilizada com sucesso em outras atividades importantes, como a manutenção remota de equipamentos (ROCKWELL, 2002) e a telecirurgia efetuada por um robô comandado remotamente por um cirurgião localizado à centenas de quilômetros de distância (SCIENTIFIC AMERICAN, 2000).

Esta tese de doutorado tem como tema a proposição de uma nova metodologia de integração CAD/CAPP/CAM voltada para manufatura remota de peças rotacionais utilizando a Internet, em especial os protocolos associados à *World Wide Web*, visando a aplicação e domínio de ferramentas relacionadas à Internet, bem como a implementação de um sistema computacional. O sistema denominado WebMachining irá disponibilizar as funções para modelagem de produto colaborativo (WebCADbyFeatures), Planejamento de Processo Auxiliado por Computador Generativo (WebCAPP) e de Manufatura Auxiliada por Computador (WebTurning), para que um usuário remoto possa modelar uma peça rotacional, gerar o planejamento do processo com alternativas e fabricar a peça em um centro de torneamento, realizando todo o ciclo de desenvolvimento do produto, relacionado ao projeto detalhado colaborativo, planejamento do processo e fabricação, de forma integrada e via Internet.

A motivação deste trabalho de doutoramento está associada à importância e à evolução tecnológica da Internet nos últimos anos, visando uma contribuição para a concepção de um novo paradigma para

desenvolvimento de produto colaborativo via Web. O objetivo principal desta pesquisa é desenvolver uma metodologia e a implementação de um protótipo computacional, sistema CAD/CAPP/CAM integrado e inteligente, voltado para fabricação de peças rotacionais via Web.

Os Sistemas desenvolvidos a partir desta metodologia poderão ser aplicados na indústria e na academia (ÁLVARES e FERREIRA, 2003). Em aplicações acadêmicas poderão ser utilizados em ensino a distância, em um contexto de laboratórios remotos. Em aplicações industriais poderão ser utilizados como parte de um serviço de prototipagem rápida para *try-out* de peças ou para o fornecimento de protótipo funcional para a própria empresa ou clientes remotos em um ambiente de tele-manufatura; podendo também ser integrado ao sistema de gestão da produção da empresa, fornecendo planos alternativos de processos para o sistema de programação da produção (*scheduling*) em "tempo real".

1.1 Resumo Orientativo da Tese

1.1.1 Questão a Responder

Como conceber e implementar um sistema integrado CAD/CAPP/CAM voltado para manufatura remota de peças baseado na Web, a fim de atender o novo paradigma de desenvolvimento de produto colaborativo, envolvendo as fases de ciclo de vida do produto associadas ao projeto detalhado colaborativo, planejamento do processo e fabricação ?

1.1.2 Hipótese a Comprovar

As necessidades primárias para a próxima geração de sistemas de *softwares* CAD/CAPP/CAM são a interoperabilidade¹ entre as ferramentas de *softwares*, colaboração entre projetistas e equipes de projeto de forma distribuída, integração de dados e conhecimento através do ciclo de desenvolvimento de produto (do projeto à fabricação), bem como a captura, troca e reutilização do conhecimento. Para atender estas demandas torna-se necessário o desenvolvimento de metodologias que definam os módulos do sistema e como estes módulos interagem entre si, estabelecendo um novo paradigma de ambientes CAD/CAPP/CAM integrados e colaborativos.

1.1.3 Objetivos

O objetivo principal desta pesquisa é desenvolver uma metodologia e a implementação de um sistema CAD/CAPP/CAM, integrado e inteligente, voltado para fabricação de peças rotacionais via Web, que poderá ser utilizado pela indústria de manufatura e por instituições acadêmicas.

Será implementado um sistema computacional, disponibilizado via Web (<http://WebMachining.AlvaresTech.com>), que possibilitará aos usuários remotos o desenvolvimento de produto de forma integrada, colaborativa e distribuída. O desenvolvimento da metodologia e a sua implementação tem como requisitos-chave a modularidade, padronização, facilidade de utilização, compartilhamento de informações e recursos.

¹Interoperabilidade deve ser entendida como a capacidade de dois ou mais sistemas ou produtos, usualmente hardware ou software, trabalharem harmoniosamente.

1.1.4 Contexto e Condições de Contorno

Esta metodologia para fabricação de peças remotas, bem como a sua implementação, limita sua abrangência ao projeto detalhado colaborativo, planejamento do processo e fabricação de peças rotacionais simétricas e assimétricas (operações eixo C, ferramenta rotativa), utilizando-se de tornos CNC e centros de torneamento. A fabricação das peças rotacionais deverá ocorrer, preferencialmente, em uma única máquina de comando numérico, em uma, duas ou três fixações. O detalhamento dos demais processos que sejam necessários (retificação, tratamento térmico e superficial, por exemplo) não serão objeto da metodologia e da implementação. A metodologia concebida poderá ser aplicada ao contexto de peças prismáticas, com as devidas adequações para esta categoria de peças.

Sua implementação utiliza como plataforma de desenvolvimento o sistema operacional Linux, as linguagens de programação Java, C e C++ e *softwares open source* como o Apache, JatLite, MySQL, PHP e TomCat. A aplicação desenvolvida é uma candidata natural à utilização da Internet 2, conectando o Graco/UnB ao Grima/UFSC com maior largura de banda e qualidade do serviço de telemanufatura, bem como as demais universidades brasileiras.

1.1.5 Enfoque

A metodologia é concebida a partir do paradigma de modelagem colaborativa baseada em síntese por *features* de projeto (adição e subtração de *features* de forma), a fim de permitir a integração das atividades de projeto colaborativo (CAD), planejamento do processo generativo (CAPP) e manufatura (CAM planejamento e CAM execução). O procedimento inicia-se na modelagem colaborativa de uma peça por *features*, num contexto de manufatura remota utilizando a Web como meio de comunicação, num modelo computacional cliente-servidor. O sistema é concebido num ambiente distribuído de agentes de *softwares* interoperáveis denominado de Comunidade de Agentes de Manufatura, sendo sua arquitetura estratificada em três camadas: Projeto, Planejamento do Processo e Fabricação. O fluxo de informações entre as três camadas da arquitetura é realizado através do encapsulamento das informações, adotando-se o *Knowledge Query and Manipulation Language* (KQML) como padrão de linguagem de mensagens entre os agentes.

1.2 Formulação do Problema

O problema objeto desta tese pode ser sintetizado da seguinte forma:

“Definir uma metodologia para um sistema integrado CAD/CAPP/CAM voltado para manufatura remota de peças rotacionais simétricas e assimétricas baseado na *Web*, bem como a sua implementação, a fim de atingir um novo paradigma de desenvolvimento de produto colaborativo via Internet, de forma a atender às necessidades primárias para a próxima geração de sistemas de softwares CAD/CAPP/CAM, que são a interoperabilidade entre as ferramentas de softwares, colaboração entre projetistas e equipes de projeto de forma distribuída, integração de dados e conhecimento através do ciclo de desenvolvimento de produto (do projeto à fabricação), bem como a captura, troca e reutilização do conhecimento. Para atender estas demandas torna-se necessário o desenvolvimento de metodologias que definam os módulos do sistema e como estes módulos interagem entre si, estabele-

cendo um novo paradigma de ambiente integrado CAD/CAPP/CAM para a era de *e-Mfg* (*electronic-Manufacturing*), levando em conta o balanceamento da funcionalidade do modelo cliente-servidor e a largura de banda disponível no sistema de comunicação de dados a ser utilizado, no caso a rede Internet.”

1.3 Oportunidade do Tema

Está em curso uma nova revolução no sistema de trabalho adotado nas empresas de manufatura, migrando das atividades auxiliadas pelo uso do computador (CAD, CAPP, CAM, CAP, etc) baseadas em sistemas integrados por computador, desenvolvidas nas décadas de 80 e 90 do século 20, para as atividades baseadas em *e-work* (*electronic-work*), que caracterizam o princípio de trabalho da era da informação, com uso intensivo da tecnologia da informação.

A Tecnologia da Informação, em especial a tecnologia de redes de comunicação e a convergência de tecnologias baseadas em redes sem fio e Internet, está abrindo um novo domínio para construção dos futuros ambientes de manufatura denominados *e-Mfg*, usando métodos de trabalho baseados em *Collaborative e-Work*, em especial para as atividades desenvolvidas durante o ciclo de desenvolvimento de produto em ambientes CAD/CAPP/CAM integrados e colaborativos. Este é um novo paradigma para estes sistemas computacionais baseados em ambiente globalizados, centrados em rede e espacialmente distribuídos, possibilitando o desenvolvimento de atividades usando *e-Work*. Isto permitirá que os desenvolvedores de produtos, projetistas, tenham maior facilidade de comunicação possibilitando o compartilhamento e o projeto colaborativo durante o desenvolvimento do produto, bem como, a teleoperação e monitoração dos dispositivos de fabricação.

A intensa competitividade entre as empresas e os novos desenvolvimentos na área da tecnologia da informação impõem uma nova abordagem crítica com relação aos sistemas CAD/CAM oferecidos atualmente à indústria e à academia. Para tirar vantagem da aplicação interativa baseada na *Web* é necessário mudar a abordagem para conceber e explorar novos paradigmas para a integração CAD/CAPP/CAM.

Tendo em vista a importância e popularidade da Internet, é apresentada uma metodologia de integração CAD/CAPP/CAM voltada para manufatura remota de peças rotacionais utilizando a Internet, em especial os protocolos associados a *World Wide Web*, visando a aplicação e domínio de ferramentas relacionadas à Internet, bem como a disponibilização das funções CAD/CAPP/CAM para que um usuário possa remotamente realizar o projeto colaborativo de uma peça rotacional (modelagem em 2D e visualização em 2D e 3D), gerar o planejamento do processo e fabricar a peça em um centro de torneamento.

Esta tese pretende ser uma contribuição para o desenvolvimento da linha de pesquisa denominada de telemanufatura (*e-Mfg*), que congrega áreas multidisciplinares envolvendo a Tecnologia da Informação, Engenharia Mecânica e Engenharia Mecatrônica.

1.4 Contribuições do Trabalho

A metodologia idealizada utiliza a tecnologia Internet e de comunicação associada à SOA (*Service-oriented Architecture*) para oferecer um novo paradigma para o desenvolvimento dos futuros ambientes integrados CAD/CAPP/CAM. Estes ambientes serão globais, centrados em rede e espacialmente distribuídos, tendo como *front-end* com o sistema CAD/CAPP/CAM navegadores *Web* e a linguagem Java, permitindo a independência da plataforma computacional do usuário.

Os Sistemas desenvolvidos a partir desta metodologia poderão ser aplicados na indústria e na academia. Esta metodologia de desenvolvimento de produto permite que uma empresa (aqui representada pelo usuário remoto) não necessariamente possua os equipamentos para a projeto, planejamento e fabricação de um lote de peças, além da empresa fabricante das peças poder ter contato direto com o cliente.

O sistema proposto apresenta muitas contribuições, inovações, para o desenvolvimento de sistemas de manufatura eletrônica e telemanufatura baseados em serviços *Web*, integrando atividades de projeto detalhado de forma colaborativa, planejamento de processos e fabricação através da modelagem por *features*, utilizando os protocolos TCP/IP, SOA e sistemas distribuídos. O sistema é inédito no que concerne à abordagem de projeto colaborativo por síntese de *features* e no processo de fabricação utilizado, sendo voltados para fabricação de peças rotacionais. Inova também na incorporação de funcionalidades associada à teleoperação via *Web* da máquina-ferramenta CNC permitindo a integração CAD/CAPP/CAM. O sistema CAPP, proposto e implementado, utiliza uma nova abordagem de mapeamento de *features* de projeto para *features* de torneamento e usinagem, denominada de “*decomposição orientada ao setup, geometria e operação*”, sendo utilizados conceitos associados à norma STEP (STEP, 2003), e também aderente ao STEP-NC (STEP-NC, 2003).

Outras contribuições específicas estão associadas à implementação do sistema utilizando a tecnologia Java *Applet* e *servlets* que impõem uma maior segurança e restrições associadas à rede de comunicação. Esta implementação permite a utilização de uma estrutura de plataforma transversal (*cross plataform*) reusável, possibilitando a sua operação com qualquer sistema operacional que suporte Java.

A seguir são apresentadas algumas contribuições do trabalho relacionadas à integração CAD/CAPP/CAM e a Tecnologia da Informação:

- CAD por *features* voltado para *Web*: é inovador na sua concepção, bem como, na aplicação voltada à modelagem colaborativa de peças rotacionais simétricas e assimétricas.
- CAPP generativo voltado para *Web* e aderente à STEP-NC: é inovador, principalmente na aplicação escolhida, voltada para planejamento do processo não-linear de peças rotacionais simétricas e assimétricas. Vários métodos propostos e implementados para resolução das atividades relacionadas ao planejamento de processo não-linear são inovadores, como a “*decomposição orientada ao setup, geometria e operação*”, “*geração de planos de processos não-lineares*”, “*otimização das condições de usinagem baseada em algoritmos genéticos*” e “*linearização do plano de processo baseado em algoritmos genéticos*”.
- Supervisão, monitoração e teleoperação do centro de usinagem Galaxy 15M da Romi (CNC Fanuc 18i-ta) via *Web*. O sistema de supervisão e teleoperação via *Web* está integrado aos

servidores de *streaming* de imagens e de som, que permitem uma imersão do usuário remoto no chão-de-fábrica.

- integração baseada em banco de dados relacional compartilhando toda a informação do sistema aliada à arquitetura multiagente e sistemas distribuídos, utilizando-se de *applets* e *servlets* em uma arquitetura cliente-servidor.
- identificação e discussão das técnicas disponíveis para telemanufatura via *Web*. Isto inclui técnicas e protocolos tais como CGI, HTTP, HTML, Java, sistemas multiagentes, VRML, *servlets*, entre outros.
- análise detalhada de características da Internet para aplicação de telemanufatura, levando em conta o balanceamento da funcionalidade disponível na arquitetura cliente-servidor em relação à largura de banda, a fim de minimizar o *delay* inerente ao protocolo TCP/IP.

1.5 Estrutura do Documento

Inicialmente é apresentada uma revisão de literatura, do capítulo dois ao cinco. O capítulo dois apresenta uma introdução à manufatura eletrônica baseada na Internet. O capítulo três apresenta uma introdução ao planejamento do processo. O capítulo quatro aborda o uso de *features* de projeto e manufatura como elemento de integração entre o projeto e a manufatura. O capítulo cinco aborda as metodologias e sistemas CAD/CAPP/CAM baseados na *Web*.

Do capítulo seis ao oito são apresentados os aspectos associados à metodologia concebida para viabilizar a integração CAD/CAPP/CAM via *Web*, denominada de WebMachining. O Capítulo seis apresenta a arquitetura e os módulos do sistema WebMachining. O capítulo sete apresenta a interface CAD-CAPP associada, detalhando as dez atividades concebidas para o planejamento de processo. O capítulo oito descreve a abordagem lógica para CAPP, descrevendo a aplicação dos métodos e algoritmos concebidos em cada uma das atividades previstas, através de um estudo de caso aplicado à uma peça exemplo.

Os capítulos nove, dez e onze apresentam os resultados obtidos na implementação computacional dos sistemas WebCADbyFeatures, WebCAPP e WebTurning, respectivamente, através da descrição e apresentação de telas dos sistemas desenvolvidos.

O capítulo doze apresenta cinco estudos de casos constituídos por dez peças exemplos, para todo o ciclo de vida da metodologia WebMachining, validando assim a metodologia proposta.

No capítulo treze são apresentadas as conclusões, contribuições e sugestões de trabalhos futuros visando a evolução do sistema WebMachining.

No Apêndice A são apresentadas uma síntese das publicações e projetos aprovados por agências de fomentos gerados a partir do trabalho de doutoramento.

No Apêndice B são apresentadas as URLs relativas aos modelos IDEF0 e IDEF1X concebidos, à modelagem conceitual e física da base de dados relacional e à biblioteca de *features* CAM-I utilizada.

No Apêndice C são apresentados os modelos matemáticos para otimização de operações de usinagem, complementando assim o capítulo sete.

No Apêndice D é apresentado o ciclo de vida do software WebMachining modelado em UML.

Capítulo 2

Manufatura Eletrônica (*e-Mfg*)

Este capítulo apresenta uma revisão de literatura relativa à Tecnologia da Informação e sua utilização na manufatura, sendo designada atualmente de *Electronic Manufacturing* (Manufatura Eletrônica). São apresentados conceitos de Manufatura Remota e Virtual (Teleoperação e Realidade Virtual), tecnologia *Web*, linguagens de programação voltadas pra *Web*, ferramentas de desenvolvimento, laboratórios remotos e virtuais, que serão utilizados no desenvolvimento da metodologia e do sistema WebMachining.

2.1 Manufatura Eletrônica (*e-Mfg*)

A Tecnologia da Informação, em especial, a tecnologia de redes de comunicação e Internet, está abrindo um novo domínio para construção dos futuros ambiente de manufatura denominados *e-Mfg* (*electronic-Manufacturing*) usando métodos de trabalho baseados em *Collaborative e-Work* (Trabalho Eletrônico Colaborativo), em especial para as atividades desenvolvidas durante o ciclo de desenvolvimento de produto em ambientes CAD/CAPP/CAM integrados e colaborativos (MALEK *et al.*, 1998; LEE, 2003 & NOF, 2004).

E-Work (Trabalho Eletrônico) foi definida pelo PRISM Center (NOF, 2004) como qualquer atividade produtiva que seja colaborativa, suportada por computador e apoiada por comunicação em organizações altamente distribuídas de robôs e/ou de pessoas ou sistemas autônomos. Em essência *e-Work* é compreendida de *e-activities* (*electronic-activities* - atividades eletrônicas), ou seja atividades baseadas e executadas através do uso de Tecnologia da Informação.

Este é um novo paradigma para estes sistemas computacionais baseados em ambiente globalizados, centrados em rede e espacialmente distribuídos, possibilitando o desenvolvimento de atividades usando *e-Work*. Isto permitirá que os desenvolvedores de produtos, projetistas, tenham maior facilidade de comunicação possibilitando o compartilhamento e o projeto colaborativo durante o desenvolvimento do produto, bem como, a teleoperação e monitoração dos dispositivos de manufatura.

Em essência, *e-work* é constituído de atividades eletrônicas, isto é, atividades baseadas e executadas através da Tecnologia da Informação. Essas *e-activities* incluem *v-(virtual)Design* (Projeto Virtual), *e-Business* (e-negócio), *e-Commerce* (e-comércio), *e-Manufacturing* (e-Manufatura), *v-Factories* (v-fábricas), *v-Enterprises* (v-empresendimentos), *e-Logistics* (e-logísticas), e semelhantemente, robótica inteligente, transporte inteligente, e assim por diante. Todas estas *e-activities* se

apoiam em computador e tecnologias de comunicação, e todas requerem colaboração e interações inerentes entre máquinas, pessoas e computadores. *E-work* inclui aplicações como telerobótica, telemanufatura, teleoperação e serviços remotos. *Telework* também faz parte de *e-work* de acordo com a definição geral.

2.2 Manufatura Remota

Segundo MALEK *et al.* (1998), telemanufatura ou manufatura remota pode ser definida como uma atividade onde uma empresa (cliente) utiliza serviços oferecidos por centros especializados (servidores) disponibilizados via rede de comunicação através das supervias da informação (Internet) para executar, em tempo real, operações e processos necessários para o projeto e a produção de bens. Desta forma atividades relativas à telemanufatura estão presentes em todo o ciclo de desenvolvimento do produto, desde a concepção do produto até a sua fabricação e distribuição.

Atualmente já existem empresas com alto nível de especialização que oferecem serviços em um ambiente voltado para telemanufatura. Estes centros especializados detêm o estado da arte na tecnologia de software (sistemas CAE/CAD/CAPP/CAM/ERP, entre outros), especialidades avançadas e acesso às informações atualizadas em seus campos, a fim de auxiliar seus clientes no desenvolvimentos de novos produtos e processos (AHN *et al.*, 1999 e MALEK *et al.*, 1998). Como exemplo cita-se o CyberCut da Universidade de Berkeley (<http://cybercut.berkeley.edu>), que possibilita a modelagem de uma peça prismática, que será usinada, utilizando-se de um sistema CAD/CAPP/CAM desenvolvido em Java (BROWN e WRIGHT, 1998). Outro exemplo foi desenvolvido por ÁLVARES *et al.* (2002) onde o sistema permite a teleoperação de uma máquina de oxi-corte CNC, bem como, a geração do programa NC da peça a ser produzida utilizando um ambiente de modelagem CAD/CAM desenvolvido na linguagem Java (<http://weboxicorte.graco.unb.br>), denominado de WebOxiCorte.

Assim, sistemas de teleoperação de equipamentos industriais enquadram-se no contexto de telemanufatura (MALEK *et al.*, 1998) nos aspectos referentes ao controle da manufatura em operações de chão-de-fábrica, bem como, os ambientes computacionais integrados de CAD/CAPP/CAM para desenvolvimento de produto. Estes sistemas e ambientes de suporte ao desenvolvimento de produto e de teleoperação estão sendo atualmente disponibilizados através da Internet/Intranet, utilizando-se dos protocolos de desenvolvimento para *Web*.

2.3 Manufatura Virtual

É necessário fazer uma distinção do conceito de Manufatura Remota (telemanufatura), que será utilizado no desenvolvimento do trabalho de doutorado, dos conceitos de Manufatura Virtual e Empresa Virtual que muitas vezes são confundidos.

A manufatura virtual pode ser definida, conforme BANERJEE e ZETU (2001), como um ambiente de manufatura sintético, integrado, que utiliza todos os níveis de decisão e controle no projeto de produto e processo, planejamento de processo, planejamento da produção e controle de chão de fábrica.

A empresa virtual é uma rede multi-disciplinar, rapidamente configurável, de pequenas empresas configuradas para projetar e produzir um produto específico. Pode ser definida como um consórcio temporário de empresas que juntam competências e recursos, suportados por redes de computadores, para melhor responder às oportunidades de mercado (CAMARINHA-MATOS, 1998), podendo inclusive utilizar-se de conceitos de telemanufatura onde as empresas com maior competência, em determinadas áreas, disponibilizam serviços especializados para as demais empresas, ajudando-as no desenvolvimento de produtos.

2.4 Trabalho Colaborativo

O trabalho colaborativo¹, como item integrante da estratégia de implantação de uma estrutura de manufatura virtual envolve conceitos de realidade virtual, em ambientes compartilhados, proporcionando aplicações de teleoperação, telerobótica e telepresença, além de uma infra-estrutura de rede de alta velocidade.

2.4.1 Realidade Virtual

O termo realidade virtual (RV) pode ser interpretado como uma forma avançada de interface do usuário de computador (BURDEA & COIFFET, 1994), a qual tem aplicação na maioria das áreas do conhecimento. Pode também ser definida como uma técnica avançada de interface, onde o usuário pode realizar imersão, navegação e interação em um ambiente sintético tridimensional gerado por computador, utilizando canais multi-sensoriais. A interface com realidade virtual envolve um controle tridimensional altamente interativo de processos computacionais, onde o usuário entra no espaço virtual das aplicações e visualiza, manipula e explora os dados da aplicação em tempo real, usando seus sentidos, particularmente os movimentos naturais tridimensionais do corpo. A grande vantagem desse tipo de interface é que o conhecimento intuitivo do usuário a respeito do mundo físico pode ser transferido para manipular o mundo virtual. PENG *et al.* (2000) apresentam um sistema CAPP Generativo baseado em realidade virtual voltado para o processo de usinagem.

A vantagem principal dessa tecnologia é o desenvolvimento e análise do projeto colaborativo, habilitando grupos de engenheiros a visualizar e manipular, em tempo real, um objeto virtual tão facilmente como eles poderiam fazer com um objeto físico.

2.4.2 Laboratório Virtual

Um laboratório virtual (BANERJEE e ZETU, 2001), (SCHIMID, 1998) e (ZHAI e MILGRAM, 1991) é um meio heterogêneo e distribuído de resolução de problemas que permite a um grupo de pessoas de diversos lugares no mundo trabalharem em um conjunto comum de projetos. No projeto e manufatura, uma empresa envolvida na produção de um produto grande e complexo, tal como um avião, deve ser capaz de simular processos diretamente para interagir com as bases de dados de projeto, as quais contêm especificações técnicas e de manufatura.

¹Nesta tese será utilizado o termo colaborativo traduzido do Inglês *collaborative*. Pode-se utilizar o termo cooperativo como sinônimo.

2.5 Teleoperação

Existem diversas situações em que a operação de um determinado dispositivo deve ser realizada através de um operador remoto, o qual encontra-se fisicamente distante do objeto de controle. Estes dispositivos remotamente operados vêm sendo utilizados em diversas áreas tais como robótica, medicina, pesquisa submarina, etc.

Geralmente, uma tarefa de manipulação remota requer uma máquina de propósito geral que possa realizar uma variedade de manipulações físicas. Muitas destas tarefas são altamente variáveis. As tarefas remotas variam em termos do local onde a tarefa tem que ser realizada e se a tarefa é conhecida a priori. Um sistema ideal para realizar estas tarefas tem que ter a mesma capacidade de um operador humano para reconhecer qual tarefa tem que ser realizada e quais passos devem ser tomados para sua realização, além de possuir flexibilidade para realizar manipulações físicas, de maneira análoga à operação de um humano.

Aspectos importantes no efetivo controle de dispositivos remotos envolvem: o atraso entre uma ação de controle do operador e sua correspondente visualização, como informação de realimentação de sua ação, mostrada em seu *display*; largura de banda do sistema de comunicação, que determinam uma parcela importante do atraso relacionado à ação de controle; experiência na tarefa de controle contínuo, manual, do dispositivo controlado; questões de segurança e erros de posicionamento e autonomia (SHERIDAM, 1992), (ZHAI, 1992) e (NOF, 1999).

A questão da autonomia é particularmente importante, já que define o quanto o dispositivo teleoperado pode funcionar independentemente do operador humano. Isto pode ser obtido através da implementação de parte da estratégia de controle em um computador localizado no *site* remoto. O operador pode estabelecer altos níveis de controle ao computador remoto, o qual pode executar os níveis mais baixos de controle.

2.6 Teleoperação e o CAM

Manufatura Auxiliada por Computador é o estágio final do auxílio do computador na produção de peças mecânicas. Dessa forma, em referências mais antigas (pré-década de 90), CAM é definido como o uso de sistemas de computação para planejar, gerenciar e controlar as operações de uma planta, através de interface computadorizada direta ou indireta com a produção (GROOVER, 1985). Dentro dessa definição, GROOVER dividiu o CAM em duas grandes categorias:

1. Execução (monitoramento e controle por computador): Estas são as aplicações diretas nas quais o computador se conecta diretamente com o processo de manufatura com o propósito de monitorar ou controlar o processo. A teleoperação está intimamente ligada a esta etapa do CAM.
2. Planejamento (aplicações de suporte à manufatura): Estas são as aplicações indiretas nas quais o computador é usado como suporte das operações de produção na planta, mas não há interface direta entre o computador e o processo de manufatura. O ambiente integrado CAD/CAPP/CAM se insere nesta categoria.

O monitoramento do processo envolve uma interface direta com o processo de manufatura, tendo o propósito de observar o processo e o equipamento associado e coletar dados do processo. O computa-

dor não é usado para controlar a operação diretamente. O controle continua nas mãos de operadores humanos, que podem ser guiados pelas informações compiladas pelo computador. O controle do processo por computador vai um passo além de apenas observar o processo, controlando-o a partir das observações. Novamente atividades associadas à teleoperação da máquina-ferramenta CNC, robô industrial, entre outros estão presentes no CAM. Ou seja, as atividades de chão-de-fábrica que são executadas pelo CAM, são agora atribuições do sistema de teleoperação como: *download* e *upload* de programas, inicialização da máquina, status da máquina, e outras funções associadas ao DNC (Comando Numérico Distribuído).

2.7 Taxonomia: Laboratórios Virtuais e Remotos

Muitos cursos da engenharia, incluindo cursos de Controle e Automação da Manufatura, normalmente usam a *Web* para demonstrações de *softwares*, tutoriais e gerência básica do curso. Entretanto, a necessidade de aulas práticas em laboratório está levando ao desenvolvimento de duas modalidades de uso da *Web* na área de Ensino a Distância (EAD): laboratórios virtuais e laboratórios remotos. Este uso da *Web* voltado ao ensino pode também ser voltado para desenvolvimento de produto, planejamento e programação da produção e manufatura, usando a mesma metodologia em especial nos chamados laboratórios remotos, que também podem ser designados de plantas industriais remotas em um contexto de telemanufatura e manufatura virtual.

Os laboratórios virtuais são simulações de dispositivos físicos por meio de software. Estes podem ser considerados demos interativos sofisticados. Os laboratórios virtuais podem ser uma bancada de testes de um sistema visando elaborar métodos para o seu controle, ou mesmo a simulação de um sistema de manufatura baseado em teoria de filas. Se a simulação for muito detalhada, pode ser um bom substituto para um laboratório real, especialmente se acompanhado de animação.

Os laboratórios virtuais acessíveis através da Internet/Intranet estão se transformando em uma maneira popular para reduzir custos de equipamento e disponibilizar conceitos através do laboratório em cursos de EAD. Estes tipos de laboratórios usam geralmente *softwares* comerciais como o LabView, MatLab (LEE *et al.*, 1998), Arena, AutoMod, Sistemas CAD/CAM, entre outros. Por exemplo, o laboratório virtual para ensino de Processamento de Sinais da Universidade de Carnegie Mellon (STONICK, 1993) e o laboratório CAD/CAM desenvolvido em Java da Universidade de Berkeley (cybercut <http://cybercut.berkeley.edu>) são utilizados em Ensino a Distância (EAD) em algumas disciplinas dos cursos de Graduação e Pós-Graduação destas Universidades. Outro exemplo mais recente é o laboratório remoto denominado TeleLabs, que utiliza a arquitetura LabView e está disponível na URL <http://www.mech.uwa.edu.au/jpt/tele>.

2.8 Infra-estrutura de Rede de Alta Velocidade

A infra-estrutura de Rede de Alta Velocidade deve disponibilizar diversas tecnologias para suportar diferentes aplicações, as quais utilizam parâmetros diversos de Qualidade de Serviço como (YE *et al.*, 2005): velocidade de transmissão, atraso (*delay*), *throughput*, agendamento e taxa de perda.

A velocidade de transmissão define a faixa de valores na qual a velocidade de uma determinada

conexão deve estar; o atraso define o máximo de interrupção aceitável para um sinal na rede, de modo a garantir o fluxo contínuo de transferência da informação; o *throughput* define a quantidade de dados transmitidos em uma unidade de tempo; o agendamento define o horário em que uma determinada conexão deverá estar disponível e a taxa de perda define a taxa máxima esperada de perda de pacotes na unidade de tempo.

No Brasil esta infra-estrutura de rede é disponibilizada pelo *backbone* da RNP que conecta as instituições de ensino superior, em especial a UnB e a UFSC. Futuramente ter-se-á conectividade através da Internet 2, onde o *backbone* terá uma maior largura de banda, sendo mais adequado para aplicações de manufatura remota e virtual.

2.9 WWW² (World Wide Web) e SOA

O serviço WWW surgiu em 1989 (TORNINCASA, 2001) como um integrador de informações, dentro do qual a grande maioria das informações disponíveis na Internet podem ser acessadas de forma simples e consistente em diferentes plataformas. A forma padrão das informações do WWW é o hipertexto, o que permite a interligação entre diferentes documentos, possivelmente localizados em diferentes servidores, em diferentes partes do mundo.

O hipertexto é codificado com a linguagem HTML (*Hypertext Markup Language*) e suas extensões, que possui um conjunto de marcas de codificação que são interpretadas pelos clientes WWW (que são os *browsers*, como o Netscape), em diferentes plataformas. O protocolo usado para a transferência de informações no WWW é o HTTP.

Ao se avaliar a efetividade do uso do WWW na manufatura remota é imprescindível a realização de um estudo e avaliação das várias ferramentas disponíveis para prover recursos avançados no WWW, tais como interatividade, animações, simulações e mundos virtuais a este serviço.

Em ÁLVARES (2003 e 2002) e SU *et al.* (2005) são apresentadas as principais ferramentas de desenvolvimento para *Web*, bem como discutido os fatores relevantes na avaliação destas ferramentas para verificar se o seu uso tem o potencial de prover um hiperdocumento interativo, eficiente e de alta de qualidade. São analisadas as seguintes ferramentas: Form/CGI, servlets, Java, Javascript e VRML (*Virtual Reality Modeling Language*).

Nos últimos anos um novo paradigma para organização de *software* e intercomunicação tem emergido, denominado de SOA (Arquitetura Orientada a Serviço), sendo um protótipo da infra-estrutura de TI dos próximos anos (MCINTOSH, 2004). SOA potencializa a proposta de serviços *Web* (MCINTOSH, 2004 & ESTREM, 2003) fornecendo orientação sobre como a infra-estrutura de TI deve ser arquitetada para utilizar serviços *Web* (<http://www.ibm.com/br/products/software/info/topic/webservices>).

²Também denominado de *Web*. Neste trabalho muitas vezes a terminologia WWW será designada simplesmente de *Web*, ou seja, a rede Internet utilizando o protocolo HTTP.

Capítulo 3

Revisão de Literatura: Planejamento do Processo

O planejamento do processo é uma função vital nas indústrias de manufatura pois fornece as informações necessárias para a transformação de matéria-prima ou produtos semi-acabados em produtos final (CHANG *et al.*, 1998). Os requisitos associados a seleção de um processo podem ser divididos em requisitos para a *peça* e para o *processo* (SMITH, 1999).

Os requisitos da peça estão associados aos aspectos funcionais sendo definidos no projeto da peça sem levar em conta a quantidade a ser produzida. Por outro lado, os requisitos de processo levam em conta os aspectos de negócio/econômico da manufatura (quanto, qual o custo e quando?). Os requisitos da peça normalmente especificados no projeto mecânico são: tolerância dimensional, tolerância de forma e posição, rugosidade superficial, forma geométrica e material. Os requisitos de processo são: tamanho do lote, custo de *setup*, tempo de *setup*, custo por peça e taxa de produção. O projeto mecânico de um novo produto deve incluir estes atributos para possibilitar à atividade de planejamento do processo a especificação do plano de processo, preferencialmente, com alternativas que será utilizado na fabricação da peça. As decisões associadas ao planejamento do processo (WANG & LI, 1991) podem ser divididas em atividades associadas ao planejamento do roteamento (macroplanejamento) e ao planejamento das operações de usinagem (microplanejamento).

3.1 Projeto Lógico do Plano de Processo

De acordo com HALEVI E WEIL (1995) o projeto lógico do plano de processo deve seguir a seguinte ordem:

- Análise preliminar da peça mecânica;
- Seleção dos processos de usinagem (operações elementares), ferramentas e parâmetros de corte;
- Agrupamento dos processos dentro de operações (*jobs*);
- Seleção de máquinas-ferramenta;

- Seqüenciamento das operações de acordo com as relações de precedência segundo as restrições de ordem: dimensional, tolerâncias geométricas, tecnológica (desbaste antes de pré-acabamento) e econômicas (redução do custo de produção e desgaste/quebra de ferramenta, por exemplo);
- Seleção dos dispositivos de fixação e referências dimensionais;
- Preparação final do arquivo com o plano de processo para a peça.

Nesta abordagem a seleção da máquina-ferramenta é feita após ter-se escolhido os parâmetros de corte através dos métodos baseados no mínimo tempo de produção (maior produtividade) ou menor custo de usinagem. Por este método as restrições para escolha da máquina baseiam-se nas condições tecnológicas de usinagem que irão definir a potência e a velocidade requerida para a máquina a ser selecionada e outras restrições disponíveis.

Segundo WANG & LI (1991) o planejamento do processo pode ser dividido em dois estágios: planejamento do roteamento e projeto da operação de usinagem. No primeiro caso faz-se um planejamento global dos processos de fabricação de uma peça tendo como objetivo a determinação do conteúdo e das seqüências de operações no plano de processos. Os fatores considerados são: material, especificações técnicas da peça, características da matéria-prima, volume de produção, máquinas, ferramentas, sistemas de medição, dispositivos de fixação, entre outros. Tendo-se identificado as operações a serem executadas no planejamento de roteamento de operações parte-se para o segundo estágio, o projeto das operações de usinagem.

A partir da listagem da ordem lógica das operações fornecida pelo roteamento deve-se realizar um detalhamento do plano de processos com as informações necessárias para as instruções de fabricação da peça. De forma bastante semelhante à abordagem de HALEVI & WEIL (1995) têm-se as seguintes atividades no estágio associado ao planejamento do roteamento, também denominado de macroplanejamento (ÁLVARES, 2002):

- Análise do desenho da peça;
- Seleção dos processos e rotas de usinagem para cada superfície da peça a ser usinada (*feature*);
- Determinação do conteúdo de cada operação e o número de operações de usinagem.
- Determinação das seqüências de operações;
- Seleção das referências de apoio para cada operação;
- Arranjo de operações de tratamento térmico;
- Arranjo de operações auxiliares (inspeção, limpeza, etc);

As seguintes atividades são efetuadas no projeto (planejamento) de operações de usinagem, também denominado de microplanejamento (ÁLVARES, 2002):

- Seleção da máquina e ferramental para cada operação;
- Arranjo das seqüências de operações elementares em cada operação;

- Determinação dos sobremateriais para cada superfície a ser usinada;
- Determinação das cotas e tolerâncias de fabricação para cada operação;
- Determinação das condições de usinagem para cada operação;
- Estimativa de tempos padrões para cada operação.

3.2 Planejamento do Roteamento - Macroplanejamento

É realizado um planejamento global dos processos de fabricação de uma peça tendo como objetivo a determinação do conteúdo e das seqüências de operações no plano de processos. Os fatores considerados são: material, especificações técnicas da peça, características da matéria-prima, volume de produção, máquinas, ferramentas, sistemas de medição, dispositivos de fixação, entre outros.

Em ÁLVARES (2001) são detalhadas as principais atividades associadas ao macroplanejamento, listadas a seguir: seleção método de usinagem; divisão da rota de processo em etapas; concentração e separação de operações.

3.3 Planejamento¹ das Operações de Usinagem - Microplanejamento

É um planejamento específico, onde cada operação definida no roteamento do processo é detalhada em termos de máquinas, ferramentas, superfícies de referência para a fabricação, dispositivos de fixação, condições de corte, etc.

Em ÁLVARES (2001) são detalhadas as principais atividades associadas ao microplanejamento: seleção de máquinas-ferramenta; seleção de ferramentas; seleção de dispositivos de fixação; seleção de superfícies de referência para a fabricação; determinação de sobremetais; determinação de dimensões e tolerâncias de usinagem; seleção de condições de corte; estabelecimento de tempos padrão; documentação do plano de processo.

3.4 Determinação do Tipo de Operação

Segundo HALEVI & WEILL (1995) muito esforço tem sido desenvolvido nos últimos anos para estabelecer teorias e algoritmos para a completa otimização do processo de fabricação com remoção de material. Entretanto a maioria dos esforços concentram-se nos parâmetros tecnológicos de usinagem, em especial na velocidade de corte. Caso a operação a ser executada seja supérflua nenhum ganho será obtido com a otimização da velocidade de corte que está diretamente relacionada com o tempo de usinagem.

As operações necessárias e sua seqüência devem ser determinadas antes da escolha das condições de usinagem e da máquina a ser utilizada, caso haja opções de máquinas. A seguir são apresentados alguns métodos para determinar os tipos de operações necessárias para fabricação da peça.

¹Também denominado de Projeto das Operações de Usinagem.

3.4.1 Capacidade do Processo e Seus Limites

A seleção da operação deve ser baseada na capacidade e nas condições de limite do processo. A profundidade de corte é um dos parâmetros que mais influenciam na tomada desta decisão. A estratégia de limites de contorno é baseado no conceito da existência de restrições técnicas e considerações econômicas na seleção das operações de corte ótimas. O método proposto por HALEVI & WEILL (1995) estabelece um conjunto de restrições técnicas como limites de contorno, e então, levando em consideração aspectos econômicos, seleciona-se as condições de trabalho dentro destes limites.

Estas restrições são usadas para estabelecer os limites da profundidade de corte, avanço e velocidade de corte. As restrições que determinarão as condições de contorno (limites) são expressas através de fórmula e tabelas como por exemplo:

- acabamento superficial como função da profundidade de corte e avanço;
- força de corte como função da profundidade de corte e avanço;
- limite da profundidade de corte como função da dureza do material e acabamento superficial;
- profundidade de corte como função do avanço;
- profundidade de corte como função de uma operação elementar a ser selecionada, entre outros.

3.4.2 Método para Seleção das Operações de Corte X Tabelas de Roteamento

Este método desenvolvido por HALEVI E WEISS (1995) tem por finalidade otimizar a definição das operações de usinagem. Baseia-se no axioma onde a otimização é buscada através da minimização do número de passes de usinagem, definindo os maiores valores possíveis das condições de usinagem (avanço e profundidade), dando preferência às estratégias de otimização bem conhecidas baseadas nos limites de contorno calculados e tabelados.

A partir dos limites da profundidade de corte calculados em relação às diversas restrições que são levadas em consideração, determina-se a quantidade de passes a ser efetuada escolhendo os parâmetros de corte para cada operação de usinagem, avanço e profundidade de corte. A velocidade de corte é determinada por outro método. Cabe destacar que a utilização deste método é mais genérico pois ainda não se escolheu a ferramenta de corte a ser utilizada bem como a máquina-ferramenta. Este método determina inicialmente o sistema e dispositivo de fixação da peça, a seguir determina as operações elementares (número de passes e condições de usinagem) e finalmente faz um agrupamento das operações elementares em operações (*jobs*) utilizando a matriz das anteriores. Após a determinação das operações define-se a velocidade de corte, escolhe-se a máquina através da Matriz de Halevi e finalmente seleciona-se as ferramentas de corte. Neste método parte-se das operações de usinagem, condições tecnológicas de corte e por último define-se as máquinas e ferramentas, minimizando as restrições que se impõe a cada tomada de decisão. Desta forma as restrições impostas pelas várias decisões tomadas ao longo das etapas do planejamento do processo são minimizadas.

Outra abordagem muito popular WANG e LI (1991) e CHANG e WYSK (1985) é a utilização de tabelas que já apresentam roteamentos típicos de usinagem para superfícies cilíndricas externas, internas e planas.

3.4.3 Determinação de Sobremateriais de Usinagem

Outra estratégia que pode ser utilizada é baseada no conceito de sobrematerial de usinagem² que é apresentado em WANG e LI (1991) e FERREIRA (1996). A quantidade de sobrematerial é a diferença entre as cotas de fabricação obtidas nas operações atual e anterior. O sobrematerial total (incluindo todas as operações) é igual à soma dos sobremateriais referentes a cada operação, desde a matéria-prima até a peça na forma final.

Existem diferentes métodos para a determinação de sobremateriais WANG E LI (1991) e (FERREIRA, 1996):

1. Cálculo: pouco usado devido à ausência de dados experimentais.
2. Estimativa baseada na experiência: normalmente pouco eficiente.
3. Tabelas de sobremateriais: dados disponíveis de manuais normalmente compilados de dados de chão-de-fábrica e experimentos. Esta é a abordagem mais utilizada na determinação de sobrematerial.

Cada cota e tolerância de projeto é obtida através de uma série de operações de usinagem. Em cada operação, a superfície é usinada até que se atinja uma certa cota e precisão de fabricação. A cota e tolerância de fabricação devem ser indicadas no plano de operações. A determinação das cotas de fabricação de outras operações de usinagem relaciona-se com os sobremateriais de usinagem. Os sobremateriais são determinados em ordem inversa, isto é, da peça acabada até a matéria prima.

3.4.4 Determinação de Condições de Usinagem e Tempos Padrão

As condições de usinagem podem ser obtidas de *handbooks* ou catálogos de fabricantes de ferramentas, ou através do método *seqüencial*. No caso da utilização de *handbooks* ou catálogos de fabricantes de ferramentas faz-se a seleção das condições de corte para uma dada combinação de materiais de peça e ferramenta, bem como, da geometria da ferramenta WANG E LI (1991).

As condições de corte utilizadas numa operação influenciam significativamente na precisão dimensional, geométrica e na qualidade superficial obtida; assim como na vida da ferramenta e na produtividade de um processo. Um grande número de pesquisas tem sido realizadas para determinação dos parâmetros de usinagem ótimos, tendo iniciado com GILBERT (1950).

A seleção de condições de corte em um modelo multi-passe envolve a determinação de quatro parâmetros: profundidade, avanço, velocidade de corte e o número de passes de usinagem.

Um modelo multi-passe generalizado descrito em TAN e CREESE (1995) & CHANG *et al.* (1998) é utilizado para seleção do número de passes de usinagem ótimo, bem como, da velocidade de corte, avanço e profundidade de corte para cada passe individual. Ao contrário de um modelo de passe simples, onde a profundidade de corte é fixa e apenas um único passe é utilizado para produzir a peça, no modelo multi-passe é assumido vários passes de usinagem para produzir a peça. Assim profundidade de corte (a_p) torna-se uma variável de controle.

²Sobrematerial de usinagem é uma camada de material que deve ser removida da superfície da peça para que a precisão e a qualidade da superfície sejam obtidas.

Normalmente um modelo multi-passe é dividido em várias operações de desbaste, com uma mesma profundidade de corte de usinagem ou com diferentes profundidades para cada passe, e um único passe de acabamento. O primeiro passo do problema associado à determinação das condições de corte é o cálculo da profundidade de corte (a_p). Se for possível a execução da operação de desbaste em um único passe, então a profundidade de corte é determinada em função do sobremetal de acabamento que deverá ser deixado para o passe de acabamento. Em operações de desbaste, a profundidade de corte é função da potência da máquina-ferramenta, do comprimento da aresta de corte, da rigidez da peça, do método de fixação, entre outras. Desta forma, podem ser necessários vários passes de usinagem para a remoção do sobremetal, com profundidades de corte menores.

CHEN e TSAI (1996) relatam várias abordagens para resolver o problema de otimização de usinagem multi-passe, incluindo métodos baseados em programação geométrica, multiplicadores de Lagrange, programação linear, entre outros, que foram utilizados no passado. SHIN e JOO (1992) apresentaram um modelo matemático para otimização das condições de corte de operações de torneamento longitudinal com restrições práticas, que influenciou a maioria dos métodos propostos a partir de então, sendo um dos trabalhos mais citados na área. Entretanto este modelo matemático só pode ser aplicado em operações de torneamento longitudinal linear, não contemplando operações de faceamento, torneamento cônico, torneamento circular, entre outras operações, normalmente presentes em peças rotacionais.

CHEN e SU (1998) apresentaram um novo modelo matemático que inclui operações de torneamento longitudinal (trajetória da ferramenta retilínea), torneamento cônico, faceamento e torneamento circular, que são considerados simultaneamente no modelo matemático, sendo baseado em um trabalho de SPEEDING *et al.* (1998). Atualmente vários métodos de otimização utilizam abordagens heurísticas baseadas em *simulated annealing*, como descritos em CHEN e TSAI (1996), KHAN *et al.* (1997) e CHEN e SU (1998), e em algoritmos genéticos, destacando REDDY *et al.* (1988), ONWUBOLU e KUMALO (2001), WANG e JAWAHIR (2001) e CHEN e CHEN (2003).

A fase final do planejamento do processo consiste em estabelecer o tempo padrão para cada operação. Os tempos padrões num plano de processo fornecem os dados iniciais para a programação da produção, escalonamento de pessoal e cálculo de custos. O tempo padrão é o tempo necessário para executar uma tarefa bem definida (p.ex. uma operação) sob certas condições. Um tempo padrão é composto pelo tempo de *setup*, tempo de processamento, tempo de manuseio, tempo de serviço e tempo de compensação. Em WANG E LI (1991) são apresentadas tabelas e fórmulas para o cálculo dos diversos componentes do tempo padrão, em especial os modelos de tempo para as principais operações de usinagem normalmente utilizadas e que dependem das condições de usinagem definidas para a operação.

3.5 Fixação em um Torno

Na usinagem por comando numérico três métodos são usados para que as dimensões entre duas superfícies, *features*, de uma peça sejam obtidas (ZHANG *et al.*, 1996):

1. usinagem das duas *features* em um mesmo *setup* (dimensões tipo I - erros tipo I) ;

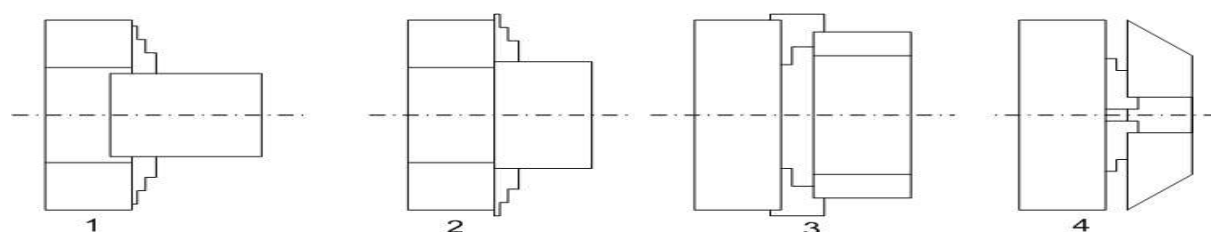


Figura 3.1: Tipos de placas utilizadas (HUANG, 1988): estilo 1, 2, 3 e 4.

2. usando a *feature* de referência como superfície de fixação de *setup*, em relação a outra *feature* (dimensões tipo II - erros tipo II);
3. usando uma referência intermediária de *setup* para máquina e duas *features* em diferentes *setups* (dimensões tipo III - erros tipo III).

Estes três tipos são denominados como *setup* método I, *setup* método II e *setup* método III; consistindo em diferentes fontes de erros de fabricação. Os erros geométricos relacionados às superfícies de usinagem estão associados principalmente ao projeto geométrico. Já os relacionamentos dimensionais associados às superfícies de usinagem estão ligados à precisão dimensional do sistema de controle da máquina. O relacionamento entre as superfícies usinadas levando em conta os erros dimensionais e geométricos, em um mesmo *setup*, está associado à capacidade máquina/processo. Assim as dimensões obtidas pelo *setup* método I são dimensões do tipo I, consistindo no menor erro de fabricação entre os três métodos. Este é o método a ser utilizado para facilitar o controle de tolerância.

Quando o *setup* método II é utilizado na fabricação NC introduz-se um erro de localização na dimensão obtida (dimensões tipo II), sendo menos precisa que as dimensões tipo I. O método III é o menos desejado dos três, pois tem-se a formação de uma cadeia dimensional associada a cada dimensão do tipo III, devendo-se evitá-lo.

Uma descrição detalhada sobre planejamento de *setup* e modelos geométricos de placa de fixação e fixações validas são apresentados em (ZHANG *et al.*, 1996) e (MASSAROPPI e MASIERO, 2001), respectivamente.

Existem muitos métodos de fixação de peças simétricas em um torno. HALEVI & WEILL (1995), ÁLVARES (2001) e (MASSAROPPI e MASIERO, 2001) apresentaram algumas orientações para selecionar o tipo de fixação mais econômica para operações de torneamento, como o uso de placas com três castanhas com e sem contra-ponta. A figura 3.1 apresenta os quatro tipos de fixação em placas com três castanhas mais utilizadas (HUANG, 1988).

3.6 Planejamento do Processo com Alternativas

A chave para a fabricação de qualquer peça é o plano de processo, o qual contém as informações necessárias à fabricação da peça, como por exemplo, operações, máquinas, ferramentas de corte, dispositivos de fixação e condições de corte (FERREIRA e WYSK, 2001). O plano de processo para uma peça ou lote deve satisfazer os requisitos de tolerância considerando também o carregamento da máquina e a eficiência do processamento. Assim, o planejamento do processo é uma atividade

complexa, e no caso de sistemas de manufatura que estão sujeitos às incertezas de demanda, o planejamento do processo deve ser executado freqüentemente.

Devido aos aspectos dinâmicos do planejamento do processo que influenciam o comportamento do ambiente, cada plano de processo deve conter alternativas para cada operação. Isto é recomendado porque fatores como carregamento excessivo de certas máquinas, mudanças constantes nos produtos, mudanças necessárias no processamento entre lotes de fabricação, entre outros, exigem que se tenha planos/métodos alternativos a fim de melhorar a performance do sistema de manufatura, como em uma célula ou um sistema flexível de manufatura.

Razões para Utilização de Planos de Processo Alternativos

A seguir são apresentados os principais motivos para geração de métodos alternativos para fabricar uma peça:

1. Uma determinada máquina pode estar em manutenção ou pode estar ocupada fabricando outra peça ou um lote. Uma máquina alternativa pode ser usada reduzindo o tempo de processamento da peça e também o trabalho em processo. Com a disponibilidade de alternativas, no caso de uma falha no chão-de-fábrica, por exemplo quebra de uma máquina, não será necessário refazer o planejamento do processo da peça, pois existe um plano alternativo para fabricação da peça.
2. A indisponibilidade de uma determinada ferramenta de corte no magazine da máquina especificada, pode ser substituída por uma ferramenta alternativa especificada no planejamento do processo com alternativas.
3. O tamanho do lote pode influenciar a escolha da operação; isto é, para pequenos lotes uma certa operação pode resultar em um tempo de fabricação menor, por outro lado para lote com quantidade maior outra operação pode ser mais apropriada para diminuir o tempo de fabricação, até mesmo se operações posteriores necessitarem de *setup* de uma nova ferramenta. Por exemplo, se apenas uma peça está sendo produzida (peça protótipo) a estratégia mais eficiente é produzir a peça usando um único *setup* de ferramentas no magazine da máquina, se possível. *Setup* de novas ferramentas na máquina demanda mais tempo do que usar as ferramentas disponíveis na máquina, sendo inclusive menos eficiente do ponto de vista de usinagem, por exemplo menor velocidade de corte a ser usada.

Existem diferentes modos de representar planos de processos com alternativas (não-lineares), destacando-se: Redes de Petri; grafos E/OU; Estruturas em árvores; gráficos direcionados. KRUTH e DETAND (1992) propõem o uso de Redes de Petri (PNs) para representar planos de processos não-lineares. PNs são usadas para representar *features*, operações e máquinas. KIRITSIS e PORCHET (1996) propõem uma abordagem baseada em PNs para planejamento do processo dinâmico e seqüenciamento. Em ÁLVARES (2001) são apresentados exemplos detalhados da utilização da modelagem do planejamento do processo através de PNs e grafos E/OU.

Capítulo 4

Revisão de Literatura: CAPP e Projeto por *Features*

Planejamento do Processo Auxiliado por Computador (CAPP) é o elo de ligação entre CAD e CAM. CAPP tem sido pesquisado desde os anos 60 (ALTING e ZHANG, 1989). Na década de setenta foi lançada a primeira aplicação comercial sendo direcionada para o armazenamento e recuperação de planos de processo para usinagem convencional.

O planejamento do processo auxiliado por computador (CAPP) é tido como parte fundamental de um sistema de Manufatura Integrado por Computador (CIM) por ser responsável pela ligação de dados de projeto (CAD) e fabricação (CAM). No passado, o desenvolvimento dos sistemas auxiliados por computador foi principalmente concentrado nos sistemas CAD (projeto) e CAM (manufatura). Devido à sua importância no contexto de um sistema CIM, nas últimas décadas grandes esforços têm sido empreendidos no sentido de um maior desenvolvimento de sistemas CAPP.

O ano de 1976 talvez seja o grande marco na corrida pelo desenvolvimento destes sistemas. Naquele ano, dois sistemas foram apresentados, um desenvolvido pela CAM-I (*Computer Aided Manufacturing-International*) e outro desenvolvido pela OIR (*Organization of Industrial Research*). Nos anos que se seguiram houve a apresentação de diversos outros sistemas (ALTING e ZHANG, 1989), mas até os dias de hoje a sua aplicação industrial ainda não é uma realidade. Um sistema comercial de CAPP famoso é o MetCapp® (<ftp://graco.unb.br/pub/capp/metcapp>), sendo utilizado a tecnologia de *features* como forma de viabilizar a integração entre as atividades de projeto e de fabricação.

4.1 Abordagens de Sistemas CAPP

Para a construção de sistemas CAPP duas abordagens básicas são seguidas (GROOVER, 1987) e (DETAND, 1993): variante e generativo. Contudo, alguns sistemas que foram construídos apresentam uma combinação das anteriores dando origem a um terceiro tipo, a abordagem semi-generativa, também chamada de híbrida.

4.1.1 Variante

A abordagem variante para o planejamento de processo é comparável à forma manual utilizada por um processista, onde o plano de processo para uma nova peça é obtido através de pequenas alterações executadas em um plano de processo já existente para uma peça similar. Em alguns sistemas variantes, as peças são agrupadas em famílias, caracterizadas por similaridades no processo de fabricação (Tecnologia de Grupo). Para cada família de peças, um plano de processo padrão que contém todas as possíveis operações, é armazenado no sistema. Através da classificação e codificação, um código é definido para cada peça, a partir de uma série de quesitos. Este código é então utilizado para definir a qual família a peça pertence e qual deve ser o plano padrão associado. Em comparação com o planejamento de processo manual, a abordagem variante é bastante vantajosa, pois a manipulação de informações se torna bem mais simples e os planos de processos podem ser padronizados. Contudo, neste tipo de abordagem, a qualidade final do plano de processo ainda depende da habilidade do processista que realiza as modificações necessárias às particularidades de cada peça. Sendo assim, nos sistemas variantes, o computador é apenas uma ferramenta de auxílio às atividades de planejamento do processos. Em *ÁLVARES et al. (2002)* é apresentado a concepção de um sistema CAPP variante concebido para *Web*.

O CAPP variante está associado com a codificação e classificação de peças baseadas na Tecnologia de Grupo¹. Nesta abordagem, as peças são classificadas e codificadas baseada em várias características ou atributos. Um sistema de codificação de Tecnologia de Grupo pode ser usado para a recuperação de planos de processos para peças semelhantes. Grande parte dos sistemas CAPP variantes, relatados na literatura, se utilizam da tecnologia de grupo como ferramenta de implementação.

4.1.2 Generativo

Neste tipo de abordagem, um novo plano de processo é gerado para cada peça do sistema, sem a intervenção de um processista. São utilizadas tabelas de decisão, árvores de decisão, fórmulas, regras de produção, sistemas especialistas, redes semânticas, etc, para definir quais são os procedimentos necessários para converter um material bruto em uma peça acabada. A entrada de informações sobre a peça para um sistema generativo pode ser do tipo texto, onde o usuário responde a uma série de questões predefinidas, ou do tipo gráfica, onde as características da peça são definidas através de um módulo de CAD. Quando se pensa em um sistema CIM, a utilização de uma interface gráfica para a definição da peça é a maneira mais interessante, pois desta forma a comunicação entre os módulos de CAD e CAPP fica prontamente estabelecida. A grande vantagem deste tipo de abordagem é que os planos de processos gerados são padronizados e completamente automatizados. Este tipo de abordagem torna-se bastante atraente para empresas que trabalham com uma grande variedade de produtos que são produzidos em pequenos lotes.

O objetivo desta abordagem é a geração automática de planos de processo, a partir da descrição de uma nova peça. Frequentemente, a descrição da peça é um modelo sólido em CAD, por ser um modelo de produto sem ambigüidades. Um banco de dados de manufatura, algoritmos e lógicas de

¹A Tecnologia de Grupo (GT) pode ser definida como o estudo de uma grande população de itens aparentemente diferentes e a sua divisão em grupos com características similares. A utilização típica da tecnologia de grupo aparece no conceito de famílias de peças, onde a codificação e a classificação são utilizadas.

tomada de decisão são os ingredientes principais de um sistema de CAPP generativo. Na década de oitenta, os sistemas de CAPP baseados em conhecimento usando técnicas de Inteligência Artificial (AI) foram desenvolvidos, sendo os sistemas especialistas uma das técnicas mais utilizadas.

De acordo com ALTING e ZHANG (1989) um sistema especialista pode ser definido como uma ferramenta que tem a capacidade de absorver conhecimento em um domínio específico, e utilizar este conhecimento para propor alternativas de solução. Até o começo dos anos 80, apesar dos esforços empreendidos, o desenvolvimento de sistemas CAPP não havia apresentado resultados muito satisfatórios. Isto porque o planejamento do processo é uma área onde não existem soluções algorítmicas, a capacidade de raciocinar é essencial e as ferramentas computacionais existentes até aquele momento eram muito limitadas neste sentido. Os sistemas especialistas, devido à capacidade de simular o processo de raciocínio de um ser humano, se apresentam como uma das ferramentas mais adequadas para o desenvolvimento de sistemas CAPP generativos.

4.1.3 Semi-generativo ou Híbrido

Este tipo de sistema aparece devido à dificuldade encontrada em se criar sistemas puramente generativos. Estes sistemas são um misto de sistemas variante e generativo. Aqui, várias tarefas que seriam realizadas pelo processista, num sistema variante, são automatizadas, mas algumas modificações no plano de processos gerado ainda são necessárias. Um sistema híbrido (generativo/variante) é descrito por DETAND (1993), sendo aplicado para geração de planos de processos não-lineares.

4.2 Integração Projeto e Manufatura Baseada em *Features*

As *features* podem ser consideradas como um elemento de integração potencial entre o projeto e a manufatura. A integração entre as etapas do ciclo produtivo é um dos caminhos que devem ser explorados na busca pela redução de custos e tempos de produção. De acordo com SHAH e MÄNTYLÄ (1994) a modelagem do produto é o ponto central para a promoção de tal integração.

Num sistema de manufatura integrado, o modelo do produto, definido no módulo de CAD, deve estar disponível para outros módulos (CAE, CAPP, CAM, CAQ, etc) para que estes possam realizar suas funções. Estes módulos devem ser capazes de enviar informações de realimentação para o módulo de CAD, permitindo que as alterações necessárias na peça possam ser efetuadas ainda na etapa de projeto, como por problemas detectados na fabricação. A utilização de *features* como base de informação para a modelagem do produto é o caminho para se atingir esta integração (TÖNSHOFF *et al.*, 1994). De acordo com SALOMONS *et al.* (1993) a tecnologia de *features* é o caminho mais adequado para se promover a integração entre as atividades de projeto, planejamento do processo, fabricação, inspeção, etc.

De acordo com SHAH, MÄNTYLÄ e NAU (1994), o primeiro trabalho relacionado com *features* foi realizado por Grayer durante seu doutorado em Cambridge, em 1976, onde *features* foram utilizadas para a automatização da geração de programas NC com base em desenhos feitos em um CAD.

Como as pesquisas em *features* são relativamente recentes, várias definições são apresentadas, cada uma formulada com base em conceitos de uma área específica. SHAH *et al.* (1993) apresentaram o conceito de *features de forma* como sendo elementos físicos de uma peça que podem ser identifi-

cados por uma forma e por alguns atributos. MAYER *et al.* (1994) apresentaram várias definições de *feature*, cada uma aplicada a uma área distinta:

- *feature de forma*: entidades relacionadas com a geometria e topologia de uma peça;
- *feature de tolerância*: entidade relacionada com os desvios aceitáveis nas dimensões de uma peça;
- *feature de material*: entidade relacionada com as propriedades mecânicas de uma peça;
- *feature funcional*: entidade relacionada com a funcionalidade da peça;
- *feature de montagem*: entidade relacionada às operações de montagem.

Segundo SHAH e MÄNTYLÄ (1995) *feature* é uma forma geométrica definida por um conjunto de parâmetros que têm significado especial para engenheiros de projeto e fabricação. IRANI *et al.* (1995) definem *feature* do ponto de vista de planejamento do processo: *feature* pode ser identificada como uma modificação na forma, no acabamento superficial ou nas dimensões de uma peça, produzida por um determinado conjunto de operações. ERVE (1988) apresenta uma definição do ponto de vista de planejamento do processos, onde *features* de forma são tratadas como características de uma determinada peça, com uma forma geométrica definida, que podem ser utilizadas para especificação de processos de usinagem, fixação e medição.

4.2.1 O Elo de Ligação Entre as Atividades de Projeto e Manufatura

Projeto e manufatura talvez sejam as etapas do ciclo produtivo que tiveram o maior avanço tecnológico das últimas décadas, com o desenvolvimento de sistemas CAD (de auxílio ao projeto) e dos sistemas CAM (de auxílio à geração de programas NC). Contudo, este desenvolvimento se deu de forma isolada, e a comunicação de sistemas CAD/CAM é hoje um grande problema, ocasionando um aumento exagerado no tempo de desenvolvimento de qualquer produto.

Este problema ocorre devido ao pequeno desenvolvimento dos sistemas CAPP, que na verdade têm a tarefa de promover a ligação entre dados de projeto e fabricação. A passagem de dados de projeto para o planejamento do processo e deste para a fabricação deve então ser o ponto estudado.

Para a geração de um plano de processo, é necessário que uma análise detalhada da peça seja empreendida. Quando se deseja utilizar o computador para a geração de planos de processo, a utilização da tecnologia de *features* facilita a análise da peça (SALOMONS, 1995). Sendo assim, é interessante que os dados manipulados por um sistema CAPP estejam na forma de *features* de manufatura. Associado às *features* de manufatura, no caso de processos de fabricação com remoção de material, tem-se as *features* de usinagem². Uma *feature* de usinagem é constituída por:

1. *Features* volumétricas: é o volume de material removido pela operação de usinagem para transformar a peça bruta em peça acabada. O volume removido é denominado de volume delta ou volume removido;

²*Feature* de usinagem é a porção ou parte da peça afetada pela operação de usinagem.

2. *Features* de superfície: é uma coleção de faces na peça que resulta da usinagem (subtração) de uma *feature* volumétrica.

Segundo HALEVI e WEILL (1995) e HOUTEN (1991) a modelagem baseada em *feature* é definida como um elemento físico da peça que tem algum significado específico de engenharia, devendo satisfazer as seguintes condições:

- ser um elemento físico da peça;
- se mapeável para uma forma genérica;
- ter um significado para engenharia;
- ter propriedades que se possa prever.

De acordo com SHAH e MÄNTYLÄ (1994 e 1995) existem essencialmente duas formas de se fazer a preparação de dados de um produto, com base em *features*, para o planejamento de processos:

- reconhecimento de *features* de manufatura a partir de um modelo sólido;
- mapeamento de *features* de projeto em *features* de manufatura.

4.2.2 Reconhecimento de *Features*

Neste tipo de abordagem, a peça criada no CAD é representada em termos de um modelo sólido. As *features* de manufatura são identificadas, com base neste modelo sólido, de forma automática ou de forma interativa.

Alguns pesquisadores relatam restrições com relação ao reconhecimento de *features* como BRONSVOORT e JANSEN (1994), que afirmam que o reconhecimento é de certa forma redundante, pois durante a etapa de projeto, informações de alto nível sobre o produto são transformadas em informações geométricas de baixo nível. Durante o reconhecimento de *features* as informações geométricas são reprocessadas com a finalidade de recuperar as informações de alto nível perdidas.

4.2.3 Mapeamento de *Features*

Neste tipo de abordagem, uma biblioteca de *features* de projeto ou de manufatura é colocada à disposição do projetista, que cria a peça através da instânciação das *features* presentes nesta biblioteca (SHAH e BHATNAGAR, 1989) e (HAN e REQUICHA, 1998). Desta forma pode-se distinguir duas categorias de sistemas de projetos baseados em *features*: projeto com *features* de forma e projeto com *features* de manufatura (HAN e REQUICHA, 1996).

4.2.3.1 Projeto com *Features* de Forma

Neste caso o modelo da peça é representado em termos de *features* de projeto. As *features* de manufatura são obtidas através da conversão ou mapeamento das *features* de projeto para o domínio da

manufatura. De acordo com SHAH e MÄNTYLÄ (1994 e 1995) o mapeamento de *features* pode ser feito segundo as seguintes classe de mapeamento³:

1. *mapeamento um-para-um* (1 => 1): quando a *feature* resultante do mapeamento é idêntica à *feature* mapeada, no caso *feature* de manufatura (do outro domínio). São exemplos as operações de recartilhamento, rosqueamento e furação;
2. *mapeamento de agregação discreta* (m => 1): quando duas ou mais *features* de um domínio são mapeadas para uma única *feature* em outro domínio. Por exemplo, “m” *features* de projeto são combinadas em uma *feature* de usinagem simples (operações de torneamento cilíndrico e perfilamento);
3. *mapeamento especializado* (1 => n) onde uma *feature* de projeto é mapeada em muitas *features* de usinagem, ou seja, alternativas de operações de usinagem, mapeamento um para “n” *features* (ranhuras e superfícies cônicas);
4. *decomposição discreta* (1 => m): quando uma *feature* é mapeada para duas ou mais *features* em outro domínio. Por exemplo um furo escareado (*feature* de projeto) é produzido por uma seqüência de operações, onde a primeira operação realiza a furação e a segunda o escareamento do furo (m *features* de usinagem);
5. *mapeamento conjugado* (m => n): quando uma *feature* (obtida após o mapeamento) é resultante de apenas algumas partes de duas ou mais *features* de um outro domínio. A *features* são decompostas em suas faces e as faces são reagrupadas, formando novas *features* com porções de *features* diferentes, assim “m” *features* produzem “n” *features*, sendo esta a mais complexa de todas as classes de *features* (vários cilindros, cones e cavidades de uma peça rotacional, decompostos e reagrupados para serem usinados com determinadas ferramentas);
6. *reparametrização variante*: quando diferentes conjuntos de atributos são utilizados para representar a mesma *feature* em diferentes domínios.

Uma revisão mais detalhada sobre mapeamento de *features* pode ser encontrada nas referências GADH (1994) e SHAH & MÄNTYLÄ (1994 e 1995).

Para que seja possível ter o modelo da peça em termos de *features* de projeto, é necessário que se realize um projeto por *features*. De acordo com FINGER e DIXON citados por SALOMONS (1995), a utilização do computador no auxílio às atividades de projeto pode se dar em três etapas, quais sejam:

- projeto conceitual ou preliminar;
- projeto estrutural ou de configuração;
- projeto paramétrico ou detalhado.

³O modelo de *features* de forma instanciado, para um determinado projeto/peça (bruta e acabada), pode ser mapeado em *features* de usinagem. Uma *feature* de usinagem representa um volume de material usinado em uma operação de usinagem com uma determinada ferramenta e *setup* da peça.

Sistemas de projeto por *features* têm sido construídos, como os relatados em SHAH e MÄNTYLÄ (1995) e WRIGHT & WANG (1998), mas são mais adequados para a etapa de detalhamento. As etapas de projeto estrutural e conceitual ainda não dispõem de sistemas baseados em *features* suficiente maduros.

O planejamento do processo necessita, além da definição da geometria da peça, de dados como tolerâncias, acabamentos superficiais e especificações de material (SHAH e MÄNTYLÄ, 1995). Se estes dados já estão prontos no modelo derivado de um CAD baseado em *features*, então, quando comparado a um sistema de reconhecimento de *features*, esta abordagem permite uma redução significativa no esforço empreendido para a comunicação CAD-CAPP. Por outro lado, ao utilizar um sistema de projeto por *features*, o projetista deve se limitar à utilização das *features* presentes na biblioteca. Esta seria então uma desvantagem do projeto por *features* em relação ao reconhecimento de *features*.

SHAH E MÄNTYLÄ (1995) descrevem várias técnicas de transformação/conversão de *features* de projeto em *features* de usinagem, destacando-se os métodos heurísticos, estruturas de nível intermediária, mapeamento baseado em células e mapeamento baseado em grafos. Cabe destacar uma técnica de mapeamento de *features* para processo muito utilizada, denominada *Shallow Knowledge* (Conhecimento Superficial), segundo SHAH e MÄNTYLÄ (1995). Nesta técnica o sistema de planejamento de processo utiliza mapas superficiais (*shallow*) com representação simbólica de formas para representação de processos de fabricação. As formas são classificadas em formas de grupos de formas similares e cada grupo é associado com processos que podem produzir a peça.

4.2.3.2 Projeto com *Features* de Manufatura

Normalmente a abordagem baseada em *features* de manufatura é empregada na modelagem de peças prismáticas, associando o volume a ser removido de material da peça diretamente à *feature* de usinagem, através da subtração de material da peça bruta em função da ferramenta de corte selecionada. Utiliza-se um modelador sólido do tipo CSG (*Constructive Solid Geometry*), por trabalhar diretamente com operações booleanas de subtração de *features* a partir da peça bruta até chegar à geometria desejada da peça acabada. Por exemplo, no processo de furação o volume de material removido está associado diretamente ao diâmetro da ferramenta, bastando definir uma operação booleana de subtração de um volume cilíndrico com determinadas dimensões de diâmetro (ferramenta) e comprimento (profundidade de corte).

Em peças cilíndricas/rotacionais esta associação não é tão óbvia como nas peças prismáticas quando se utiliza modeladores CSG, pois o projetista deve associar o volume a ser removido de material à forma geométrica desejada, sendo esta estratégia pouco utilizada nas peças rotacionais. Como é possível representar uma peça rotacional através de 2D, não é necessário utilizar-se de um modelador sólido (WANG e LI, 1991). No projeto de peças rotacionais é mais adequado utilizar a abordagem de projeto baseado em *features* de projeto trabalhando-se com uma biblioteca de *features* para criar a geometria desejada. As peças rotacionais podem ser formadas por dois grupos de *features* de projeto (HAM e LU, 1998) e SALOMONS, KAPPERT *et al.* (1993): concêntricas (*features* internas e *features* externas) e não-concêntricas (*features* de Eixo C, ferramenta rotativa).

4.3 Projeto Por *Features*

Segundo SHAH e MÄNTYLÄ (1995) duas metodologias de projeto por *features* são comumente utilizadas:

1. *Destruction by Machining Features*, também é conhecida por *Destructive Solid Geometry* ou *Deforming Solid Geometry* (DSG);
2. Síntese por *Features* de Projeto.

A abordagem *Destruction by Machining Features* inicia-se com um modelo da peça na qual a peça desejada será usinada. O modelo da peça é criado pela subtração na peça bruta de *features* que correspondem ao material removido por operações de usinagem. O sistema Cybercut (<http://cybercut.berkeley.edu>) utiliza esta abordagem, obrigando o usuário a remover entidades geométricas de uma peça de formato regular através do fresamento e furação da peça bruta, de tal forma que o processo de manufatura da peça é incorporado, de forma inerente, ao projeto. Alguns sistemas comerciais também suportam esta abordagem. Estes sistemas usam um conjunto de *features* pré-definidas que são subtraídas do sólido base (peça bruta).

Na abordagem com *features* de manufatura o projetista é forçado a definir a geometria da peça usando um conjunto de *features* associadas com um processo de manufatura específico. Exemplos de utilização desta abordagem são os sistemas *Quick Tournaroud Cell - QTC* (CHANG *et al.*, 1988), *First Cut* e Cybercut (SMITH e WRIGHT, 2000). Para usinagem, as *features* disponíveis para o projetista são limitadas as *features* negativas e todas são subtraídas da peça bruta. A vantagem deste método é que as *features* de usinagem estão diretamente disponíveis no modelo da peça não sendo necessário o reconhecimento ou mapeamento de *features*. A imposição deste método é a sua restrição, pois assume que o projetista tem um amplo conhecimento de manufatura e força o projetista a pensar em termos de *features* de manufatura. O projetista está interessado, inicialmente, na forma da peça e nos aspectos funcionais.

A segunda abordagem, síntese por *features* de projeto, difere da abordagem anterior com relação ao modelo que pode ser construído tanto pela adição quanto pela subtração de *features*, não sendo necessário iniciar com um modelo de uma peça bruta.

Em todas as abordagens de Projeto por *Features*, as peças são criadas diretamente usando *features* e o modelo geométrico é gerado do modelo de *features*. Isto requer que o sistema de projeto (CAD) tenha definições genéricas de *features* disponibilizadas pela biblioteca de *features*, permitindo a instanciação das *features* pela especificação de dimensões, parâmetros de localização, a *feature/face/aresta* sobre a qual está localizada e vários outros atributos (propriedades), restrições e relacionamentos.

A figura 4.1 apresenta um diagrama da técnica de criação de *features* utilizada na metodologia. O usuário interage com o modelador de *features* e constrói o modelo de *features* a partir da instanciação de *features* disponibilizadas pela biblioteca de *features*. A partir do modelo de *features* o modelador geométrico cria o modelo geométrico da peça. O modelo de *features* define a forma, atributos dimensionais, posição das *features*, restrições geométricas e atributos não-geométricos (*features* de materiais, *features* de Tolerância, entre outros). Já o modelo geométrico é constituído por entidades

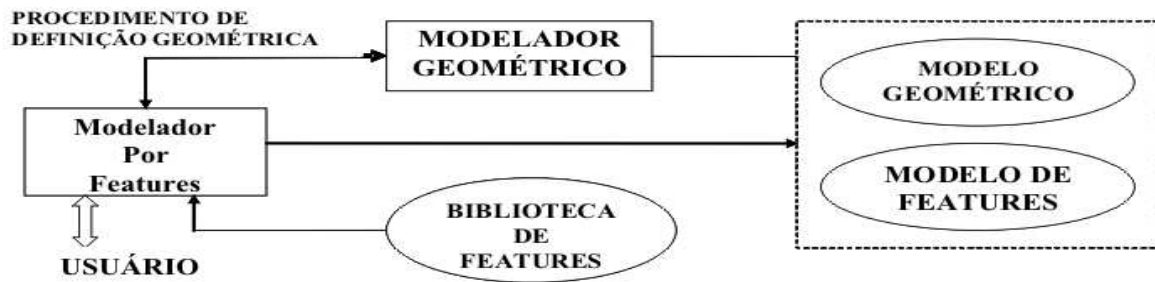


Figura 4.1: Abordagem de criação de *features* utilizando projeto por *features* (SHAH e MANTYLA, 1995).

topológicas, gráficos topológicos, entidades geométricas e ponteiros topológicos-geométricos. Modelos geométricos são representados normalmente por modelos gráficos 2D e 3D (arame), modelos de superfície e modelos sólidos.

4.3.1 ISO10303-224: Definição de Produto para Planejamento de Processo

O Protocolo de Aplicação Part 224 do STEP (ISO 10303-224, 1997) foi desenvolvido para definição de produtos mecânicos voltado para o planejamento de processo computadorizado usando *features* de forma. Part 224 inclui tolerâncias, propriedades do material e de superfícies e dados de controle da produção e administrativos. Uma *Feature* de manufatura é definida como uma informação de produto necessária no planejamento de processo automatizado. Existem três categorias de *features* de manufatura, segundo Part 224 (SHAH e MÄNTYLÄ, 1995): *features* de transição, *features* repetidas e *features* de usinagem.

Uma *feature* de usinagem é definida como uma forma que representa volumes para serem removidos por usinagem. Contudo, uma saliência também é tratada como uma *feature* de usinagem. Uma *feature* de transição especifica uma área de transição entre duas superfícies; diferindo das *features* de usinagem na qual nenhuma orientação de posicionamento é necessário. Uma *feature* repetida é especificada por uma *feature* base e um arranjo geométrico de cópias da *feature* base.

Features de usinagem são definidas em termos de objetos de aplicação, que inclui a definição do tipo de *features* (saliência, furo, recartilhado, canal, cavidades, perfis 2D, rosca, etc), perfil, localização/orientação, dimensões e tolerâncias. *Features* de transição aplicam-se apenas à quinas/cantos, como: filetes que são arredondamentos aplicados à arestas côncavas, chanfros (é plano) e cantos arredondados (é curvo) são aplicados à quinas convexas. *Features* repetidas podem incluir padrões retangulares ou circulares.

Pode-se definir também uma *feature* composta como sendo a união de uma ou mais *features* em uma seqüência específica. O Protocolo de Aplicação Part 224 especifica perfis 2D, abertos e fechados, comumente utilizados em operações de torneamento, como: T, V, N_gon, Circular e Retangular.

O Protocolo de Aplicação Part 224 especifica onze categorias de Unidades de Funcionalidade que são necessárias para a atividade de planejamento de processo, como "machining_features" e "manufacturing_part_properties".

A norma STEP (ISO 10303-224, 1997) define uma *machining_feature* (nomenclatura STEP)

como sendo uma “unidade de funcionalidade (UoF) que contém as informações necessárias para identificar as formas que representam os volumes de material que devem ser removidos da peça por um processo de usinagem”. Já um *volume_feature* é visto como o “volume adicionado ou subtraído de um volume pré-existente”.

4.3.2 ISO10303-48: *Features* de Forma

Segundo SHAH e MÄNTYLÄ (1995) existem vários tipos de *features* geométricas como: *features* de forma (descrevem porções da geometria nominal/idealizada da peça); *features* de tolerância (descrevem variação geométrica em relação a forma nominal); *features* de montagem (descrevem relacionamentos entre peças em uma montagem mecânica). Além das *features* geométricas temos as *features* funcionais e *features* de materiais.

Em função da aplicação, ou seja de um particular ponto de vista, visão, uma *feature* de forma pode ser chamada de *feature* de projeto, *feature* de manufatura, *feature* de inspeção, *feature* de torneamento, *feature* de usinagem, *feature* de fixação, *feature* de custo, etc.

Modelos de *features* são dependentes do domínio, ou seja da aplicação. Quando uma peça é projetada por *features*, o modelo resultante nem sempre é conveniente para uma dada aplicação, como para o planejamento de processo. O Protocolo de Aplicação Part 48 (ISO 14649, 2003) define um modelo de *features* de forma de propósito geral tendo por objetivo fornecer uma definição de *features* independente do domínio, ao contrário de Part 224 que é limitado ao escopo para produzir *features* pelos processos de torneamento e fresamento.

Uma “*form_feature*” é um “*shape_aspect* o qual conforma-se para algum padrão pré-concebido ou estereótipo e é, para propósito de aplicação, tratado como uma ocorrência deste estereótipo”. As *features* de forma podem ser classificadas, segundo Part 48, como *Features* de Volume, *Features* de Transição e Padrão de *Features*. Já um “*volume_feature*” é visto como o “volume adicionado ou subtraído de um volume pré-existente”.

Observa-se que existe um conflito evidente na definição de *features* segundo os Protocolos de Aplicação Part 48 e Part 224 (SHAH e MÄNTYLÄ, 1995). Na metodologia desenvolvida neste trabalho é utilizado como referência a definição de *features* associada ao Protocolo de Aplicação Part 224, utilizando sempre que possível os conceitos associados ao Protocolo de Aplicação Part 48, desde que não estejam em conflito.

As técnicas de mapeamento a serem utilizadas são derivadas de:

1. *métodos heurísticos*, onde as *features* podem ser combinadas ou decompostas em função de um conjunto de regras de transformação de uma visão à outra (mapeamento 1 => 1, por exemplo);
2. técnica de *mapeamento baseado em células*, onde volumes são combinados dentro de diferentes grupos extraíndo-se as *features* de usinagem de *features* de projeto. Muitos métodos de decomposição de célula volumétrica tem sido propostos (SHAH e MANTYLA, 1995), sendo que todos os métodos envolvem quatro fases:
 - determinar o volume a ser removido por usinagem (peça acabada menos peça bruta);
 - particionar cada volume de remoção de material em sub-volumes elementares;

- combinar os volumes elementares em *features* de usinagem;
- combinar as *features* de usinagem para operações de usinagem com base na classificação anterior.

4.4 Aderência à Norma ISO 14649 (STEP-NC)

ISO 14649 (2003) é basicamente uma representação estruturada de um plano de processo para operações de torneamento, fresamento, eletroerosão, entre outras, sendo baseada em *features* de usinagem (SUH *et al.*, 2003). ISO 14649 está em desenvolvimento para operações de torneamento, sendo denominada de ISO 14649 - Part 12.

Enquanto o padrão ISO 6983 (código G) preocupa-se em especificar a trajetória de movimentação de ferramenta, o STEP-NC especifica o plano de usinagem por meio de “*Workingstep*”, como a entidade central. Um *Workingstep* associa uma *feature* de usinagem à uma operação de usinagem (ferramenta de usinagem, condições de corte, funções da máquina-ferramenta e estratégia de usinagem associada à movimentação de ferramenta).

As informações que constituem o ISO 14649 - Part 12 são compostas por quatro grupos (figura 4.2):

1. descrição da tarefa: descreve a seqüência lógica de tarefas executáveis (*Machining_workingstep*, *Turning_workingstep* e *NC_function*). Os detalhes de cada *workingstep* são apresentados na descrição da tecnologia em referência à uma descrição da ferramenta, associados à uma *feature* de usinagem;
2. descrição da tecnologia: refere-se aos detalhes de cada *workingstep* associando-o à uma *feature* de usinagem e à operação de usinagem. Assim descreve-se a tecnologia de usinagem utilizada, listando as ferramentas, estratégias de usinagem e condições tecnológicas de usinagem;
3. descrição da ferramenta: define os detalhes da ferramenta de corte especificada em um *workingstep*;
4. descrição da geometria: define a geometria associada a uma *feature* de usinagem.

STEP-NC também suporta seqüências de processos não-lineares definidas através das entidades presentes em objetos executáveis de um programa STEP-NC. Estas são entidades “*SUBTYPE*” de “*program_structure*”:

- operações seletivas/*Selective*: entidade que define um conjunto de executáveis em que apenas um será executado, correspondendo ao “OU”;
- paralelas/*Parallel*: entidade que permite executar vários executáveis em paralelo e disparados ao mesmo tempo;
- não seqüenciais/*Non_sequential*: entidade que define um conjunto de executáveis que serão executados sem uma ordem pré-estabelecida, correspondendo ao “E”.

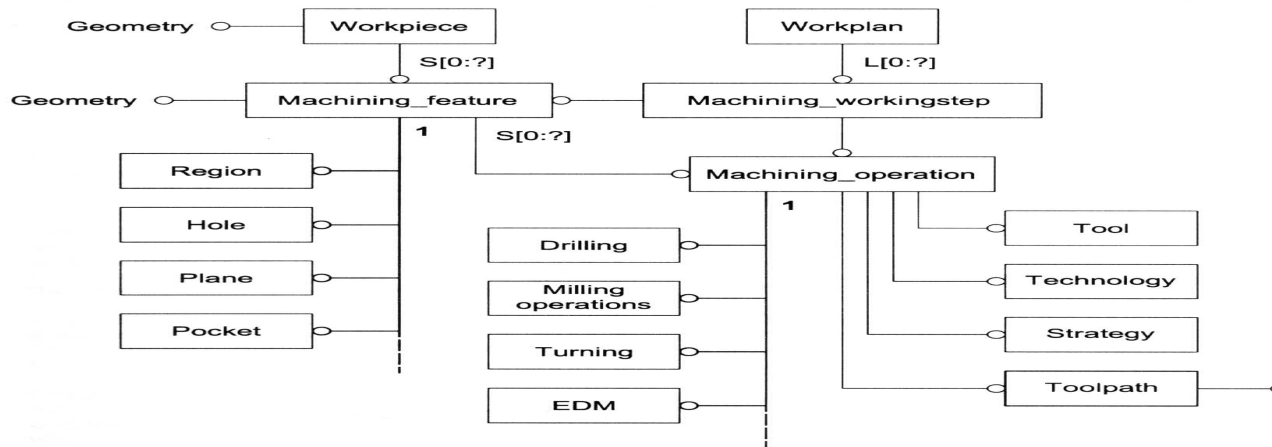


Figura 4.2: Estrutura de modelo de dados para STEP-NC representada por um diagrama Express_G (ISO 14649 - PART 1, 2003).

Assim pode-se gerar planos de processos não-lineares usando um esquema de representação baseado em Gráfico de Sequência de Processo (PSG), que nada mais é do que um grafo E-OU (SUH *et al.*, 2003). Um PSG é uma representação gráfica da seqüência de *workingsteps* descritas em termos de *features* de usinagem (*Machining_features*) e operações de usinagem (*Machining_operations*) usando relacionamento E-OU.

Atualmente tem-se três tipos previstos de controladores aderentes à STEP-NC (SUH *et al.*, 2002):

1. controlador convencional: pós-processamento do ISO 14649 para código G;
2. novo controlador STEP-NC: interpretação do programa ISO 14649 e geração da trajetória de ferramenta;
3. novo controlador STEP-NC inteligente: executa de forma autônoma várias funções inteligentes baseado na ISO 14649, por exemplo: seleção automática de ferramentas, monitoração automática, detecção de *features* de usinagem, seleção automática das condições de corte, entre outras.

Por não se dispor atualmente de controladores comerciais STEP-NC e pelo escopo do trabalho não ser o STEP-NC, o programa NC será gerado apenas no padrão ISO 6983, códigos G e M.

4.5 Sistemas CAPP Voltados Para Operações de Torneamento

ERICKSON (1988) revisou mais de 127 sistemas CAPP e projetos associados. Ele observou que a incompatibilidade nos *softwares*, no *hardware* e nas diferentes representações de produto, recursos e planos de processo inibiram o desenvolvimento de um sistema integrado de propósito geral. O autor sugere que os futuros sistemas CAPP deverão ser modulares, fáceis de integrar, amigáveis e de fácil manutenção. HAM e LU (1988) sugerem que as futuras pesquisas deverão incluir a integração de projeto e manufatura aplicando técnicas de inteligência artificial.

SHUNMUGAM *et al.* (2002) destacam que poucos sistemas CAPP reportados na literatura levam em consideração aspectos de otimização da seqüência das operações ou sugerem seqüências alternativas de operações ou de planos de processo. Nesta linha, uma abordagem muito utilizada é a matriz de precedência (RHO *et al.*, 1992) e o *Hamiltonian Path* (HP), que é análogo aos problemas de planejamento de processo baseado no gráfico de precedência e matriz de custo de operação (IRANI *et al.*, 1995).

GIUSTI *et al.* (1989) apresentaram o CAPP KAPLAN descrevendo uma abordagem baseada em conhecimento para planejamento de processo de peças rotacionais. CHO *et al.* (1991) descrevem o desenvolvimento de um sistema integrado de planejamento de processo e monitoração para operações de torneamento. SHYU *et al.* (1987) descrevem um sistema integrado de CAD/CAPP/CAM para centros de torneamento. DESAI e PANDE (1991) apresentaram um sistema de modelagem baseado em *features* para CAPP voltado à fabricação de peças rotacionais. RICO *et al.* (1997) descrevem um CAPP para peças rotacionais baseado em *features* e perfis 2D. KRUTH e DETAND (1992) descrevem um sistema de CAPP para geração de planos de processos não-lineares.

CAY e CHASSAPIS (1997) fazem um levantamento do estado da arte associado ao CAPP e concluem que sistemas de engenharia distribuídos baseados em agentes irão permitir a integração das atividades de desenvolvimento de produto de forma concorrente. Destacam também a falta de boas metodologias para desenvolvimento de sistemas de engenharia concorrente que incluam planejamento de processo. Sugerem que as metodologias IDEF (HARRINGTON, 1984) devem ser utilizadas para descrever os modelos das atividades de produção.

LI *et al.* (2005) e LI (2005) apresentaram um sistema de planejamento de planos de processos baseado na *Web* aplicado em projeto distribuído, sendo voltado para peças prismáticas. São apresentados três módulos de otimização: seleção de máquinas e ferramentas, determinação de *setups* e seleção da seqüência de operações de usinagem.

Chung e Peng (2004) apresentaram um sistema de seleção de ferramentas e máquinas baseado na *Web* voltado para o domínio de peças rotacionais.

Sistemas de planejamento do processo automatizados (CAPP) relatados na literatura como PART, PARTS, ROUND, RNDFIX e XPLANES (HOUNTEN, 1991 e ERVE, 1985) implementam as seguintes funções de planejamento de operações e processos: interpretação do modelo de produto; seleção de máquinas-ferramenta; seleção de ferramentas; determinação de *setups*; projeto de fixações; determinação dos métodos/operações de usinagem; seleção de ferramentas de corte; determinação das seqüências de usinagem; cálculo das trajetórias de ferramenta; cálculo das condições de usinagem; geração do programa NC; planejamento de capacidade.

Normalmente nestes sistemas, custo e *Throughput* são objetivos secundários a serem obtidos e os recursos disponíveis como máquinas-ferramenta, ferramentas de corte e mão-de-obra são as restrições. A partir destes recursos disponíveis deve-se buscar a elaboração de um plano de processo exequível e realista.

Diversas arquiteturas de sistemas CAPP baseados na abordagem de *features* e desenvolvidos para operações de torneamento, tanto de caráter acadêmico como comercial, são detalhados em ÁLVARES (2001 & 2002), dentre os quais se incluem os seguintes sistemas:

- ROUND: sistema concebido a partir do sistema BID, que era um programa interativo desenvolvido para cálculo econômico das condições de usinagem em operações de torneamento

(HOUTEN, 1984);

- **TECHTURN** (*Technological Oriented Turning System*): é um sistema de planejamento do processo generativo para peças torneadas (HUANG, 1988).
- **CAPP Grima** (REZENDE, 1996): se aplica à peças rotacionais que serão executadas em células de manufatura com estratégia de usinagem bem definida.
- **Seicos Sigma Σ 10L Multi Control**: é um *software* integrado ao CNC⁴ de centros de torneamento da Hitachi Seiki que consiste de uma interface gráfica (*front-end*) com o usuário para realizar as atividade de planejamento do processo (<http://www.hitachiseikiusa.com/controls>)).

4.6 Outros Trabalhos Associados ao Planejamento de Processos

4.6.1 Estratégias de Movimentação de Ferramentas e Superfícies Intermediárias

A ISO 14649 - PART 12 (2003) descreve as seguintes estratégias, usadas como referência:

- **Estratégia Unidirecional**: utilizada em operações de torneamento longitudinal e faceamento em uma única direção de corte (movimento linear) usando ferramenta para direita ou para esquerda, onde a ferramenta vai de um lado ao outro, e então retorna para a posição de partida, podendo ser do tipo *unidirecional_perpendicular* (movimento de retorno é perpendicular a direção de avanço) ou *unidirecional_diagonal* (movimento de retorno/afastamento é definido por um ângulo não perpendicular com a direção de corte, normalmente 45°);
- **Estratégia Bidirecional**: utilizada em operações de torneamento longitudinal e faceamento em duas direções de corte (movimento linear) usando ferramenta neutra, onde a ferramenta executa o corte no movimento de ida e no movimento de retorno, na direção oposta, a 180° do movimento de ida;
- **Estratégia de Contorno**: utilizada em operações de perfilamento ao longo do contorno externo ou interno da peça. Utilizado normalmente para operações de semi-acabamento e acabamento;
- **Estratégia para Rosqueamento**: utilizada em operações de rosqueamento com a especificação de várias profundidades de corte por camada, para remoção de material. Pode ser do tipo profundidade de corte constante, profundidade de corte variável ou profundidade de corte removida constante determinada pelo CNC. A direção do rosqueamento pode ser esquerda, direita, central, esquerda zig-zag e direita zig-zag;
- **Estratégia de Sangramento** (*Grooving*): usada para qualquer operação de sangramento, podendo ser de um único passe ou multi-passes;

⁴Este CNC é na realidade um computador industrial baseado na arquitetura PC.

- Estratégia Explícita: faz-se uma definição exata de todos os movimentos necessários para a trajetória da ferramenta, quando não se pode utilizar as estratégias anteriores. Ou seja, o fabricante do CNC pode definir estratégias de movimentação de ferramenta proprietária.

IBRAHIM *et al.* (1994) demonstram que para um volume de material associado a um perfil de peça cônica, onde o envelope cônico é representada em 2D por uma reta inclinada, utiliza-se duas alternativas de movimentação de ferramenta: abordagem de cortes paralelos; abordagem de corte bi-axial onde a trajetória da ferramenta é paralela ao perfil da peça. Um melhor resultado é obtido para a abordagem de corte bi-axial, pois minimiza o caminho de deslocamento de ferramenta, diminuindo o tempo de usinagem. IBRAHIM *et al.* (1994) também demonstram que para um perfil circular da peça o corte bi-axial é mais eficaz que o corte paralelo.

Operações de torneamento exigem a capacidade de se tratar com o conceito de geometrias intermediárias, que são geradas durante as várias operações. Por exemplo, em função da geometria da ferramenta determinados segmentos não poderão ser executados durante uma operação de usinagem associado a uma fixação/*setup*, pois haverá interferência entre a ponta da ferramenta e a peça durante a movimentação da ferramenta, inviabilizando a operação. Devido a este fato uma nova geometria associada à cinemática peça-ferramenta será necessária, sendo denominada geometria intermediária. É necessário a usinagem desta geometria a partir de um novo *setup*.

4.6.2 Seleção da Seqüência de Operações de Usinagem

Uma peça é composta por um determinado número de *features* que é produzida por uma sucessão de diferentes operações de usinagem. Esta abordagem *top-down* tem como objetivo apresentar um método que pode ser implementado computacionalmente a fim de agrupar as operações elementares em operações (*jobs*), realizar o seqüenciamento destas operações elementares agrupadas⁵ de acordo com seu relacionamento de precedência (anteriores). Este método de determinação da seqüência de operações de usinagem foi desenvolvido por SUNDARAM (1986) e também, descrito por HALEVI & WEILL (1995) que apresenta uma pequena diferença na resolução da matriz de relacionamento de precedência.

Esta metodologia estabelece um procedimento sistemático para seleção da seqüência de operações de usinagem. As operações elementares de usinagem podem ser agrupadas em uma seqüência e o grupo de operações elementares pode ser processado em uma única máquina. A partir da especificação de operações de usinagem é montada uma tabela de relacionamento de precedência de operações baseada nos seguintes fatores:

- Tolerância Dimensional;
- Tolerância Geométrica (acabamento superficial, tolerância de forma e posição);
- Considerações Tecnológicas (incluir desbaste antes de acabamento, por exemplo);
- Aspectos Econômicos (incluir semi-acabamento, por exemplo).

⁵Operações elementares agrupadas denominadas a partir de agora simplesmente de *operações*.

O trabalho de SUNDARAM (1986) detalha o método e apresenta a solução através de manipulação de matrizes, o que é facilmente implementado em um computador. A abordagem da solução proposta por HALEVI & WEILL (1995) não permite uma implementação computacional sendo mais adequada como demonstração de uma solução didática. Outra abordagem de otimização da seqüência de usinagem é baseada em algoritmo genético, sendo apresentado por USHER e BOWDEN (1996).

Uma diferença básica nesta proposta em relação à USHER e BOWDEN (1996) e SUNDARAM (1986) está associada aos elementos da matriz adjacente, que nesta proposta é composta por *features* de usinagem. Já em USHER e BOWDEN (1996) é composta por *features* de forma e em SUNDARAM (1986) por operações de usinagem.

ZHAO *et al.* (2002) relacionam quatro condições nas quais uma *feature* de forma rotacional irá necessitar de uma operação de acabamento:

1. se a superfície da *feature* envolve operações secundárias como rosqueamento, sangramento ou corte;
2. se a superfície da *feature* está relacionada a um requisito de tolerância dimensional $\leq \pm 0,6$ mm.
3. se a superfície da *feature* tem um requisito de tolerância geométrica;
4. se a superfície da *feature* tem um requisito de acabamento superficial $\leq 12,5 \mu m$.

As principais tolerâncias dimensionais consideradas são associadas a ângulo, diâmetro, comprimento, raio e largura. As tolerâncias geométricas (forma e posição) são consideradas de dois tipos: *features* simples e *features* inter-relacionadas. As simples incluem cilindricidade, planicidade, perfil de linha, perfil de superfície, circularidade e retilineidade. As inter-relacionadas/ligadas incluem paralelismo, angulosidade, ortogonalidade, posição, concentricidade, simetria e batimento.

As tolerâncias associadas às *features* inter-relacionadas envolvem duas *features*, sendo a segunda, usualmente, uma *feature* de referência. Normalmente as *features* inter-relacionadas são processadas em uma mesma fixação da peça, a fim de garantir a restrição geométrica definida (ZHAO *et al.*, 2002).

Segundo CAM-I (1986), na geração da definição dos passes de corte para remoção de material deve-se utilizar a abordagem lógica denominada *outside-in*, de fora para dentro, trabalhando do contorno externo da peça bruta (*blank*) para as *features* internas. Neste tipo de cenário certas *features* devem ser representadas como sendo *filhas* de uma *feature* pai, estabelecendo um relacionamento de precedência; fazendo com que a *feature* pai seja gerada antes da *feature* filha, como ocorre em uma cavidade (*feature pai*) com um furo na sua base (*feature filha*). Assim é necessário definir as restrições de precedência de volumes a serem removidas utilizando um grafo E/OU.

VARVAKIS (1991) apresenta um ambiente de modelagem de produto baseado em *features* para peças rotacionais, o qual é constituído por uma GUI que permite a modelagem de produto utilizando uma biblioteca de *features*. O modelador de produto está conectado ao módulo denominado “Pacote de Aplicações” que realiza as atividades de planejamento do processo e a geração do código G, sendo integrado a uma base de dados. Utiliza-se a mesma estratégia definida em CAM-I (1986) para determinação/reconhecimento das regiões de usinagem, avaliando as regiões externas e as regiões internas da peças, varrendo o perfil associado ao modelo de *features*, e determinando os relacionamentos entre

as *features* de projeto e mapeando-as em *features* de usinagem. Utiliza-se a abordagem CAM-I (1986) baseada no relacionamento Pai-Filho entre *features*, para esta finalidade.

ZHAO *et al.* (2002) apresentaram um conceito para integração de sistemas CAD e sistemas baseados em conhecimento para seleção de ferramentas de corte e condições de usinagem voltado para operações de torneamento, denominado EXCATS. Relatam também o desenvolvimento de um segundo sistema denominado CADEXCATS que integra o EXCATs a um sistema CAD comercial utilizando a abordagem de reconhecimento de *features* a partir de um arquivo IGES.

ZHANG *et al.* (1997) apresentaram uma proposta de planejamento de *setup* baseado em grafo, relacionando as alternativas de *setup*, que inclui o agrupamento de *features* em *setups*, seleção de referências de *setup* para cada *setup* e determinação da seqüência de *setups*.

AVILA e WEBER (2000) apresentaram uma especificação para o domínio de planejamento de processo para peças rotacionais simétricas utilizando a abordagem SNLP (*Systematic NonLinear Planning*).

USHER e BOWDEN (1996) apresentaram uma abordagem baseada em algoritmo genético para determinação da seqüência de operações para peças rotacionais. Utiliza regras de produção e gráficos de precedência. As *features* são divididas em *features* primárias e secundárias, onde uma *feature* primária é a forma básica de uma parte da peça (cone e cilindro, por exemplo) e as *features* secundárias determinam aspectos de forma detalhada no formato de atributos à uma forma primária, como roscas e ranhura.

Uma *feature* secundária é definida como residente sobre uma *feature* primária. As restrições de seqüenciamento observadas e critérios de otimização de seqüenciamento são classificados como:

1. restrições de exequibilidade: referência de localização, acessibilidade, não destruição, tolerância geométrica e estrita precedência.
2. critério de otimização: número de *setups*, continuidade do movimento e perda de precedência.

Features de forma em peças rotacionais apresentam relacionamentos entre *features* basicamente de dois modos: adjacente e subordinada (CHERNG *et al.*, 1998). As *features* de forma são divididas em *features* principais/primárias (cilindro, cone, face, etc) e *features* auxiliares/secundárias (filete, chanfros, entalhe, rasgo, etc). O relacionamento das *features* principais são adjacentes, isto é, anterior e posterior. Já o relacionamento entre as *features* principal e secundária são subordinados, isto é, Pai-Filho.

SCHÜTZER *et al.* (1992) desenvolveram dois protótipos para a modelagem semântica, geométrica e tecnológica de peças prismáticas e rotacionais utilizando o conceito de "*manufacturing features*" (FINDES-P) e de "*design features*" (FINDES-R), respectivamente, os quais foram implementados usando o Sistema CAD/CAM Euclid3 utilizando o sistema operacional VMS.

4.7 Métodos e Ferramentas Para CAPP

Existem vários métodos para descrever a estrutura de decisão no planejamento do processo. Os métodos de representação do conhecimento relacionam-se diretamente à lógica de decisão nestes sistemas. Os seguintes métodos de lógica de decisão são utilizados no planejamento do processo: tabelas de

decisão e técnicas baseadas em inteligência artificial destacando-se os sistemas especialistas, lógica difusa, redes neurais, sistemas multiagentes e algoritmos genéticos (WANG e LI, 1991).

Uma base de dados relacional é a maneira mais adequada de compartilhar dados para as diversas atividades a serem desenvolvidas para integração CAD/CAPP/CAM. Durante a modelagem por *features*, no cliente, é necessário o armazenamento das instâncias de classes de *features* definidas no processo de modelagem da peça. Estas informações devem ser armazenadas em uma base de dados, no servidor. Com estas informações disponíveis, o sistema de CAPP poderá utilizar diversos métodos para resolução dos problemas referentes ao planejamento de processo. Os diversos módulos do CAPP podem trabalhar utilizando uma arquitetura multiagente em que cada atividade a ser executada pelos módulos do CAPP seria considerada um agente (ULIERU *et al.*, 2000)

A seguir é apresentado uma pequena revisão sobre informações de manufatura em banco de dados relacionais, técnicas baseadas em Inteligência Artificial utilizadas na tomada de decisão de um CAPP generativo, e finalmente uma introdução sobre agentes de softwares, sistemas multiagentes e a arquitetura de desenvolvimento de agentes denominada JATLite. Em ÁLVARES (2003 e 2002) e MANET (2002) estes tópicos são apresentados em detalhes.

4.7.1 Banco de Dados

Segundo JUNIOR (2001) os modelos de banco de dados⁶ são coleções de ferramentas conceituais para a descrição de dados, relacionamentos, semântica e restrições. O modelo conceitual de banco de dados associado às informações de fabricação pode ser dividido em duas grandes áreas (WANG e WALKER, 1989):

1. Conhecimento de manufatura: conhecimento declarativo;
2. Regras de manufatura: conhecimento procedural.

O conhecimento do planejamento do processo pode ser classificado de maneira geral em: conhecimento dos componentes/peças; conhecimento de máquinas; conhecimento de ferramentas; conhecimento de sistemas de fixação; conhecimento de materiais.

Este conhecimento pode ser armazenado em banco de dados e acessado pelas várias atividades de manufatura incluindo o planejamento do processo. Normalmente o modelo de base de dados relacional (JUNIOR, 2001) é utilizado para armazenar e disponibilizar este conhecimento de manufatura para a empresa. Logo o desenvolvimento de sistemas CAPP utilizando ambientes de *softwares* baseados em Sistemas de Gerenciamento de Banco de Dados Relacional (DBMS) é uma abordagem muito utilizada para compartilhamento do conhecimento da manufatura nas diversas atividades necessárias ao planejamento do processo.

4.7.2 Inteligência Artificial

Técnicas ou abordagens emergentes de Inteligência Artificial (AI) como redes neurais, lógica difusa, algoritmos genéticos, sistemas especialistas, etc, oferecem novas oportunidades e abordagens para resolver complexos problemas associados à elaboração automática de planos de processo (DÉPINCÉ *et*

⁶Também denominado de base de dados.

al., 2001). A maioria dos métodos utilizados na pesquisas de CAPP são baseados em lógica difusa ou um *mix* de métodos usando redes neurais, lógica difusa e sistemas especialistas. Atualmente as técnicas de AI são usadas em funções específicas como seleção de ferramentas de corte, seqüenciamento das operações, reconhecimento de padrões, etc.

O uso de técnicas de AI em planejamento do processo tem destacado a necessidade de se ter a abordagem certa aplicada no domínio do problema. Algumas técnicas como algoritmos genéticos ou lógica difusa podem tratar com planos de processos contendo alternativas, gerando planos de processos não-lineares. Considera-se que o próximo desenvolvimento de sistema de CAPP é a integração de técnicas de AI dentro do campo de inteligência artificial distribuído, como em uma arquitetura computacional baseada em Agentes (DÉPINCÉ *et al.*, 2001). Neste caso as atividades são distribuídas através de múltiplos resolvidores de problemas especializados, ou seja o Agente.

HASHMI *et al.* (1998) desenvolveu uma aplicação para seleção das condições de usinagem utilizando-se da abordagem baseada em lógica difusa. HASHMI *et al.* (1998) conclui o estudo indicando que existe uma boa correlação entre os dados utilizados de velocidade de corte recomendados pelo *Machining Data Handbook* e os valores de velocidade previstos pelo modelo em lógica difusa.

O uso de sistemas distribuídos tem mostrado que a inteligência distribuída melhora a eficiência do processo de decisão. Em sistemas distribuídos, o problema original é decomposto em sub-problemas e cada sub-sistema especialista é responsável por uma tarefa específica. Em um único sistema, diferentes fontes de conhecimento coexistem e o uso de Inteligência Artificial Distribuída (DAI) permite o gerenciamento evitando os conflitos devido às várias formas de representação de conhecimento utilizadas. No final as soluções parciais criadas pelos sub-sistemas são colocadas juntas, de forma ordenada, para obter uma solução global.

Outra vantagem de técnicas de DAI é a possibilidade de integrar o usuário na malha de decisão: de maneira a controlar e disparar os subsistemas especialistas por meio de três maneiras: orientado ao usuário, orientado pelo cenário e disparo automatizado.

Em DÉPINCÉ *et al.* (2001) é apresentado uma arquitetura para CAPP generativo baseada na representação multiagentes. MAS distribui as atividades de planejamento do processo para múltiplos agentes especializados e coordena-os de maneira a obter a solução global. Um agente é uma entidade que pode atuar em um ambiente, comunicar-se com outros agentes e cujo o comportamento é resultado de suas observações, conhecimento e interação com outros agentes. Um agente pode ser um modelo de dados (máquinas, ferramentas, dispositivos, etc), um supervisor, funções matemáticas ou um ser humano.

O planejamento de processos de fabricação tem uma característica bastante peculiar, pois não existe um algoritmo predefinido para a geração dos planos de processo. Assim, se faz necessária a utilização de uma metodologia de programação especialmente voltada para a solução de problemas desta natureza. A tecnologia de sistemas especialistas se apresenta como uma alternativa bastante atrativa, também.

4.7.3 Agentes de Software

Agentes são metáforas computacionais para um tipo de componente de software que em seu funcionamento pretende imitar o comportamento de seres humanos no tocante a diversas características, parti-

cularmente no que se refere a um comportamento independente e inteligente (MANET, 2002). Apesar dessa idéia geral, diversas definições de agentes podem ser encontradas na literatura. WOOLDRIDGE e JENNINGS (1995) apresentaram duas propostas de definição de agentes, no que chamam de noção forte e noção fraca, dependendo do grau de "realismo" com que se pretende entender a metáfora de agente. A seguir são apresentadas várias definições associadas ao tema agentes, sendo descritas com mais detalhes em MANET (2002).

4.7.3.1 Uma Noção Fraca para Agentes

Segundo WOOLDRIDGE e JENNINGS (1995), uma noção fraca do termo agente é aquela que o utiliza para denotar qualquer hardware ou sistema de computação baseado em software, que apresente as seguintes propriedades:

1. autonomia: os agentes operam sem a intervenção direta dos humanos ou outros agentes, além de ter algum tipo de controle de suas ações e estados internos;
2. habilidade social: os agentes interagem com outros agentes (e possivelmente com humanos) através de algum tipo de linguagem de comunicação de agentes (ACL);
3. reatividade: os agentes percebem seu ambiente, o qual pode ser o mundo real, um usuário via uma interface gráfica de usuário (GUI), uma coleção de outros agentes, a Internet, ou talvez a combinação de alguns destes ou de todos respondendo de forma oportuna às mudanças que ocorrem neste ambiente;
4. pró-atividade (*pro-activeness*): os agentes não simplesmente reagem em resposta ao ambiente, mas têm a capacidade de exibir condutas baseadas em metas, tomando a iniciativa em relação a suas próprias ações.

4.7.3.2 Uma Noção Forte para Agentes

Para alguns pesquisadores, particularmente aqueles que trabalham na área da Inteligência Artificial, o termo agente deve ter um significado mais específico que o adotado pela noção fraca (MANET, 2002). Em sentido geral, estes pesquisadores descrevem um agente como um sistema de computação que, além de apresentar as propriedades identificadas na noção fraca, deve ser definido ou implementado utilizando-se conceitos que usualmente são aplicáveis aos seres humanos. Por exemplo, é bastante comum na Inteligência Artificial caracterizar os agentes usando noções aplicáveis à mente humana, tais como conhecimentos, crenças, intenções e obrigações. Para FRANKLIN (1996) “agentes inteligentes são entidades de software que realizam um conjunto de operações em nome de um usuário ou outro programa com certo grau de independência ou autonomia, e desta maneira empregam algum conhecimento ou representação das metas e/ou desejos do usuário”.

4.7.4 Arquitetura de Agentes

Um tópico importante relacionado à tecnologia de agentes diz respeito às diferentes arquiteturas que podem ser idealizadas para a implementação de agentes. Entende-se por uma arquitetura de agentes

como sendo o conjunto de especificações e técnicas utilizadas para a definição funcional dos agentes (WOOLDRIDGE e JENNINGS, 1995), onde:

- MAES define uma arquitetura de agentes como: "Uma metodologia particular para a construção de agentes. Especifica como os agentes podem ser decompostos na construção de um conjunto de módulos (componentes) e como estes módulos podem interagir entre si. O conjunto total de módulos e suas interações deve especificar como os dados dos sensores e o estado interno do agente serão utilizados para determinar as ações realizadas pelo agente e seu futuro estado interno. Uma arquitetura envolve técnicas e algoritmos que suportem esta metodologia."
- KAELBLING considera uma arquitetura de agentes como: "Uma coleção específica de módulos de software (ou hardware), tipicamente designados por caixas com setas que indicam os dados e o fluxo de controle entre os módulos. Uma visão mais abstrata de uma arquitetura é uma metodologia geral para projetar a decomposição em módulos particulares direcionados à tarefas particulares."

4.7.5 Tipologia de Agentes

Existem diferentes critérios para a classificação de agentes. Por exemplo, os agentes podem ser classificados por sua mobilidade, ou seja, sua habilidade em mover-se por diferentes nós de uma rede (MANET, 2002). Segundo este conceito os agentes podem ser classificados como agentes estáticos ou agentes móveis. Outra possível classificação pode ser feita segundo sua arquitetura. Assim, eles poderiam ser classificados como deliberativos ou reativos (também conhecidos na literatura como reflexivos).

Os agentes também podem ser classificados segundo os diferentes atributos que possam idealmente exibir (MANET, 2002). Neste sentido, JENNINGS (1994) descreve uma classificação prática dos agentes (baseado em alguns casos no que são os agentes, e outros no papel que eles executam), resultando na seguinte classificação:

1. Agentes colaborativos (*Collaborative Agents*): agentes geralmente estáticos e grandes sobre os quais há ênfase na autonomia e cooperação com outros agentes para executar tarefas em prol de seus proprietários, em ambientes multiagente abertos ou de tempo limitado. Eles podem ter aprendizado, mas este atributo não é geralmente de maior importância em sua operação. Para coordenar suas atividades, eles podem realizar algum tipo de negociação para alcançar acordos mutuamente aceitáveis.
2. Agentes de interface (*Interface Agents*): suportam e fornecem uma ajuda pró-ativa, geralmente para um usuário utilizando um programa de aplicação complexo. Este tipo de agente enfatiza sua autonomia e capacidade de aprendizado para executar as tarefas em nome de seus proprietários. Uma metáfora usada para definir os agentes de interface, é que são assistentes pessoais os quais estão colaborando com o usuário no mesmo ambiente de trabalho. Sua cooperação com outros agentes, se existe, é tipicamente limitada para responder às consultas.
3. Agentes móveis (*Mobile Agents*): processos de software com capacidade de movimentar-se através das redes de longo alcance (*WANs, Wide Area Networks*), como é o caso da WWW

(*World Wide Web*), interagindo com *hosts* externos, executando tarefas em nome de seus proprietários e retornando a sua origem com o resultado das tarefas executadas. Estas tarefas ou obrigações podem ser as mais diversas possíveis, desde fazer uma reserva de vôo até manipular uma rede de telecomunicações.

4. Agentes de informação (*Information Agents*): administradores de informação WWW pró-ativos, dinâmicos, adaptativos e colaborativos que executam o papel de administradores, manipuladores ou coletores de informação de qualquer recurso distribuído.
5. Agentes reativos ou reflexivos (*Reactive Agents*): agentes que não possuem internamente modelos simbólicos de seus ambientes, embora respondam de maneira "estímulo-resposta" ao estado atual do ambiente no qual são colocados.
6. Agentes híbridos (*Hybrid Agents*): agentes cuja constituição é uma combinação de duas ou mais filosofias.
7. Sistemas de agentes heterogêneos (*Heterogeneous Agent Systems*): algum software baseado em agentes que combine dois ou mais agentes das categorias descritas acima.

Neste trabalho de doutorado, devido a sua natureza, os agentes podem ser classificados como do tipo colaborativo ou do tipo de interface, onde as atividades de modelagem colaborativa de produto e planejamento do processo demandam uma autonomia na tomada de decisão e cooperação com outros agentes.

4.7.6 Linguagens de Agentes

Com o desenvolvimento da tecnologia de agentes, uma grande variedade de ferramentas de software encontram-se disponíveis para o projeto e construção de sistemas baseados em agentes. O número emergente de protótipos de linguagens de agentes é um sinal de que a tecnologia de agentes está sendo amplamente utilizada e que muitas aplicações baseadas em agentes estão sendo desenvolvidas.

WOOLDRIDGE e JENNINGS (1995) escreveram a respeito: "por uma linguagem de agente, entende-se um sistema que permita programar hardware ou software de sistemas de computação em termos de alguns dos conceitos desenvolvidos pelos teóricos de agentes. Como mínimo, espera-se que tal linguagem inclua alguma estrutura correspondente a um agente"

Como a questão "O que é um agente?" é muito polêmica e não existe um consenso (e possivelmente nunca existirá) com respeito a esta questão, alguns pesquisadores consideram uma linguagem como linguagem de agentes e outros podem não considerá-la da mesma maneira. O que é certo, é que elas prestam-se em diferentes graus para diferentes tipos de definições e aplicações de agentes.

Até agora foi utilizado a terminologia "linguagem de agentes" para referenciar a linguagem utilizada na criação dos agentes. É importante destacar que existe um outro enfoque para esta terminologia, onde utiliza-se "linguagem de agentes" para referenciar a linguagem que dois ou mais agentes utilizam para comunicar-se entre si, quando interagindo em um ambiente comum (MANET, 2002). Esta terminologia é usual dentro do contexto de sistemas multiagentes, sendo que as "linguagens de agentes" são utilizadas para o intercâmbio de conhecimento a ser compartilhado entre os agentes que

Tabela 4.1: Troca de mensagens utilizando KQML.

Agente A	Agente B
<i>(ask-all</i> : sender AgenteA	<i>(tell</i> :sender AgenteB
:receiver AgentB	:receiver AgenteA
:in-reply-to id0	:in-reply-to id1
:reply-with id1	:reply-with id2
:language Java	:language Java
:ontology: Features	:ontology: Features
:content "[Feature.ID (X,L)]"	:content "[Feature.ID (30,40),Feature.ID (76,10)]"

cooperam/colaboram em sistemas deste tipo. Dois tipos de linguagens de agentes são destacados na literatura (NWANA, 1997): as chamadas Linguagem de Comunicação de Agentes (ACL) e as chamadas Linguagens de Conteúdo (CL).

As ACL são utilizadas para deixar explícito o ato comunicativo relacionado à mensagem, ou seja, destaca-se o ato comunicativo pretendido pelo agente quando de sua comunicação com seu interlocutor. Alguns exemplos desta linguagem são o KQML (*Knowledge Query and Manipulation Language*) e o FIPA-ACL (*Foundation for Intelligent Physical Agents - Agents Communication Language*).

As CL, por sua vez, são utilizadas para expressar o conhecimento que se deseja compartilhar com o destinatário da mensagem. Exemplos de linguagens deste tipo são o KIF (*Knowledge Interchange Format*), FIPA-CLL (*FIPA Content Language Library*), FIPA-SL (*FIPA Semantic Language*), FIPA-RDF (*FIPA Resource Description Framework*), FIPA-CCL (*FIPA Constraint Choice Language*) e o FIPA-KIF (*FIPA Knowledge Interchange Format*).

Uma mensagem na linguagem de comunicação entre agentes KQML consiste de uma performativa (PERFORMATIVE - meta mensagem) que denota o ato de fala ou intenção do agente, seus argumentos obrigatórios, como o conteúdo da mensagem e outros argumentos que descrevem o conteúdo. No exemplo da tabela 4.1, o AgenteA quer saber se o AgenteB pode fornecer a posição (em termos da coordenada X) e comprimento (L) de uma *feature* cilíndrica_interna, no contexto da ontologia⁷ de *Features-cilíndrica_interna* (campo :ONTOLOGY *Features-cilíndrica_interna*). O AgenteA pergunta isto (intenção diretiva), então, ao AgenteB, que retorna as respectivas posição e comprimento das *features* cilíndricas_internas contidas na sua base de dados (intenção informativa). Como se pode observar, as mensagens respondem e conectam-se logicamente umas às outras pelo campo :IN-REPLY-TO. Desta feita, a primeira mensagem (id1) está ligada à mensagem id0, e a segunda (id2) responde à mensagem id1. O conteúdo das mensagens foi codificado no formalismo lógico de representação Java (campo :LANGUAGE).

⁷Uma ontologia (GRUBBER, 1995) é uma maneira de se definir um conteúdo específico sobre um conhecimento a ser compartilhado e reusado entre diferentes agentes, estabelecendo convenções em três níveis: (i) formato de representação da linguagem; (ii) protocolo de comunicação entre agentes (humanos ou computacionais); (iii) especificação do conteúdo do conhecimento compartilhado (conceituação). O processo de conceituação implica em definir um corpo de conhecimento, representado formalmente, que seja baseado nos seguintes elementos: objetos, entidades, relações entre objetos e entre conceitos. Uma ontologia é descrita de forma hierárquica através de uma taxonomia, como a utilizada na Taxonomia de *Features* de Forma CAM-I (1986).

4.7.7 Objetos e Agentes

A engenharia de software baseada em agente é com frequência comparada à programação orientada a objetos (POO), em que os agentes, assim como os objetos, compartilham algumas propriedades tais como: encapsulamento, herança (com certa frequência) e fornecem uma interface baseada em mensagens para suas estruturas de dados internas e seus métodos (ou algoritmos). Na programação orientada a objetos, o significado de uma mensagem pode ser diferente de um objeto para outro (princípio do polimorfismo). Na engenharia de software baseada em agentes, os agentes utilizam uma linguagem comum, que tem uma semântica independente dos agentes, ou seja, os agentes devem ter uma linguagem de comunicação (ACL) comum de modo que todos possam se entender.

Outra diferença entre objetos e agentes é que um objeto tem uma postura passiva diante do mundo (MANET, 2002). Ou seja, um objeto é uma entidade do mundo que somente recebe mensagens, efetuando um comportamento em resposta a elas. Por outro lado, um agente tem uma postura ativa, ou seja, é uma entidade do mundo que possui um ciclo de vida e que durante esse ciclo de vida estará adquirindo continuamente informações do mundo, através da busca ativa por mensagens que se encontrem no ambiente em que está inserido.

4.7.8 O Desenvolvimento de Sistemas de Agentes (ou Multiagentes)

Agentes podem ser integrados à sistemas de software na forma de componentes especialmente desenvolvidos para operar de maneira pró-ativa e contínua. Desta forma, pode-se adotar as mesmas metodologias de projeto de *software* utilizadas, por exemplo, em sistemas orientados a objetos, com apenas pequenas modificações. Essas modificações ocorrem principalmente na fase de *design*, durante a atribuição de funcionalidades (responsabilidades) aos componentes de *software*. Nesse instante, ao invés de se projetar um componente na forma de um autômato (grupo de objetos trocando mensagens entre si, o que seria natural na metodologia orientada a objetos), utiliza-se a metáfora de um agente. Nesse caso, o projeto do componente-agente passará por etapas de definição das responsabilidades do agente (sua motivação - geratriz do comportamento pró-ativo), e ciclo de vida. A atribuição de uma motivação a um agente não é uma tarefa trivial, sendo atualmente fonte de inspiração para toda uma área de pesquisas dentro do tema "sistemas inteligentes".

4.7.9 Qual Modelo de Tomada de Decisão Utilizar ?

A utilização de uma arquitetura baseada em sistemas multiagentes (MAS) aliada a SOA (MCINTOSH, 2004) é muito atrativa atualmente, por permitir o desenvolvimento de sistemas colaborativos e distribuídos utilizando-se redes de comunicação baseadas no protocolo TCP/IP e serviços baseados na *Web*, em uma arquitetura cliente/servidor.

Desta forma pode-se utilizar diversos tipos de agentes trabalhando cooperativamente e de forma distribuída na resolução dos diversos problemas associados ao ciclo de desenvolvimento de produto (projeto, planejamento do processo e fabricação) via *Web*. Por exemplo, pode-se utilizar um sistema de gerenciamento de base de dados relacional (MySQL® ou SQL®) para compartilhar as informações dos recursos disponíveis de manufatura (máquinas, ferramentas, informações de materiais, dispositivos de fixação, etc) e ter os agentes como os resolvidores das atividades de projeto e plane-

jamento do processo. Os agentes podem ser implementados utilizando diversas abordagens na sua lógica de decisão: sistemas especialistas baseado em regras de produção, algoritmos genéticos, redes neurais, tabelas de decisão, entre outros.

Pode-se utilizar a linguagem de comunicação de agentes KQML ou FIPA-ACL como linguagem que os agentes usam para se comunicar. Por exemplo, o sistema Cybercut (AHN *et al.*, 2001) utiliza o KQML como linguagem de comunicação. A arquitetura de agentes utilizada pelo sistema Cybercut pode ser tomada como referência para o trabalho de doutorado. Cybercut foi concebido em um ambiente em rede de agentes de *softwares* interoperáveis. A arquitetura Cybercut não prevê o projeto detalhado colaborativo (CAD colaborativo), ao contrário da metodologia WebMachining.

Cybercut utiliza a ferramenta computacional JATLite (*Java Agent Template Lite*). JATLite é um pacote de programas escritos em Java baseado na teoria de agentes denominada *Typed-Message Agents*, que permite aos usuários a criação de agentes de *softwares* que comunicam-se de forma robusta usando protocolo TCP/IP. Um agente é definido em termos de uma comunidade de agentes que trabalha por meio de computação distribuída usando *typed-messages*. Esta ferramenta pode ser obtida gratuitamente na URL <http://java.stanford.edu/index.html>. JATLite oferece uma infra-estrutura básica na qual agentes registrados com um *Agent Message Router* (AMR) usam um nome e um *password*, conectando-se e desconectando-se da Internet, mandando e recebendo mensagens, transferindo arquivos com FTP, e geralmente trocando informações com outros agentes através dos vários computadores onde eles estão sendo executados.

JATLite foi utilizado na implementação do CyberCut e também será utilizado no desenvolvimento da metodologia e na implementação do sistema WebMachining. O protocolo de mensagem compartilhado utilizado é o KQML. JATLite permite o desenvolvimento de uma infra-estrutura para *Typed-Message Agents*, definido nos termos de uma comunidade de agentes. Maiores informações sobre KQML e sistemas desenvolvidos utilizando-se a arquitetura de agentes podem ser encontradas na URL <http://www.cs.umbc.edu/kqml>. Uma abordagem interessante utilizando o *kernel* de sistema especialista JESS (Clips portado para Java), KQML e JATLite é descrito na URL <ftp://suse.lab.unb.br/pub/papers/Coop1S99-paper.pdf>.

Em CAMARINHA-MATOS *et al.* (2001) é apresentada uma comparação da utilização de plataformas (ambientes) para desenvolvimentos de sistemas multiagentes. São avaliados vários ambientes de desenvolvimentos e analisados três com maior profundidade: JATLite, FIPA-OS e Jade (*Java Agent DEvelopment Framework*) (<http://sharon.cselt.it/projects/jade/>). Jade e FIPA-OS obtiveram uma melhor avaliação que JATLite, por terem uma melhor documentação, compatibilidade FIPA-ACL e maior funcionalidade no desenvolvimento de sistemas multiagentes. Em SHEN e NORRIE (1999) é apresentada uma revisão de literatura sobre sistemas baseados em agentes para manufatura inteligente, sendo que muitos utilizam KQML e JATLite.

Maiores informações sobre outras ferramentas acadêmicas para desenvolvimento de sistemas multiagentes podem ser encontradas na URL <http://www.agentbuilder.com/AgentTools/academic.php> e em SHEN, *et al.* (2001). Ferramentas comerciais podem ser encontradas na URL <http://www.agentbuilder.com/AgentTools/commercial.php>. Na URL <http://www.dbgroup.unimo.it/Miks/selectedAgentSoftware.html> são apresentadas ferramentas para desenvolvimento de agentes de propósito geral, *toolkits* para desenvolvimento de MAS, entre outros.

Capítulo 5

Metodologias e Sistemas CAD/CAPP/CAM Baseados na Web

Neste capítulo é apresentada uma revisão de literatura sobre arquiteturas e sistemas CAD/CAPP/CAM colaborativos e correlatos, centrados em rede e distribuídos. Duas arquiteturas que se destacam são o Cybercut e o WebSpiff, ambas baseadas na tecnologia de *features*.

5.1 Introdução

A Tecnologia da Informação, em especial, a tecnologia de redes de comunicação e Internet, está abrindo um novo domínio para construção dos futuros ambiente CAD/CAPP/CAM (LEE *et al.*, 1999). Este é um novo paradigma para estes sistemas computacionais baseados em ambiente globalizados, centrados em rede e espacialmente distribuídos. Isto permitirá que os desenvolvedores de produtos, projetistas, tenham maior facilidade de comunicação possibilitando o compartilhamento e o projeto colaborativo durante o desenvolvimento do produto, bem como, a teleoperação e monitoração dos dispositivos de manufatura. Com o crescimento da popularidade dos navegadores baseados no WWW está ficando mais evidente que o ambiente de projeto e manufatura orientado à rede se tornará um novo paradigma para o desenvolvimento de produto (ZHOU *et al.*, 2002), (ADAMCZYK, 2002) e (OFFODILE e MALEK, 2002).

No chão-de-fábrica vários exemplos de arquiteturas para protocolos de camada de aplicação do Modelo ISO/OSI, *Manufacturing Message Specifications* (MMS), estão sendo desenvolvidos. Estas arquiteturas utilizam-se do protocolo TCP/IP por apresentar menores custos de implementação e devido ao fato da maioria dos dispositivos de manufatura (CNC, PLC, Robôs, entre outros) não suportarem o MMS. Segundo CHEAH *et al.* (1997) o TCP/IP é uma ótima alternativa para suportar o protocolo MMS como um padrão de fato em redes de computadores. Assim o CAM em sua fase de execução, também pode usufruir dos benefícios do TCP/IP (WANG *et al.*, 2004).

5.1.1 CAD/CAM Colaborativo

A filosofia de aplicações CAD/CAM tradicionais é baseada no ambiente computacional desenvolvido na década de 70 e restrito à aplicações tipo "*single-location*" (KAO e LIN, 1996). A Tecnologia

CAD/CAM foi concebida como uma aplicação "single-user" e um usuário CAD/CAM pode apenas se comunicar com a unidade de processamento central (CPU) do seu computador. Em sistemas CAD/CAM comerciais os usuários não podem, normalmente, se comunicar entre eles, mesmo que a CPU seja um sistema de tempo compartilhado como uma estação de trabalho Unix ou um *main-frame*. Com a popularização das Redes de Comunicação Local (LAN) sistemas CAD/CAM foram implementados em servidores de redes, permitindo que os usuários acessem o servidor de qualquer computador conectado à LAN. Contudo, a interação entre os usuários CAD/CAM ainda ocorre da mesma forma que antes, apesar da utilização dos modernos serviços de comunicação disponíveis como E-mail, fax, ftp, http, entre outras, que diminui o tempo gasto para comunicações.

Passadas mais de três décadas a tecnologia CAD/CAM tem tido um grande sucesso em aplicações industriais tendo como resultado um significativo aumento na produtividade e competitividade. A Engenharia Simultânea ou Concorrente (CE) tem sido reconhecida como uma filosofia de manufatura capaz de possibilitar a concretização da Manufatura Integrada por Computador (CIM) que não pode alcançar todo seu potencial sem CAD/CAM. Já é uma realidade a manufatura global que aumenta de forma acentuada a implementação de empreendimentos multinacionais e multiregionais. Assim, é necessário a utilização de tecnologias CAD/CAM para facilitar a Engenharia Simultânea (CE) nas Empresas Integradas por Computador (CIE) em operações multiregionais ou multinacionais.

Histórias de sucesso sobre CE usualmente enfatizam a ligação entre muitas divisões na empresa para compartilhamento e troca de idéias e experiências. Este estilo de trabalho emprega uma equipe multifuncional, equipe virtual ou uma força tarefa multidisciplinar. Para que isto ocorra os engenheiros deverão trabalhar efetivamente como time/equipe, sendo este um fator crítico para o sucesso do desenvolvimento de produto de forma simultânea (HARTLEY, 1992). O engenheiro de projeto define toda a geometria CAD, passando-a para a divisão de engenharia no estilo tradicional de trabalho, *over-the-face* ou *over-the-wall*. Por outro lado, com a equipe de projeto organizada como uma força tarefa (CE), o engenheiro de projeto e o processista podem sentar-se lado a lado e trabalhar juntos, se eles estiverem lotados no mesmo departamento. Isto resulta em um processo de desenvolvimento contínuo, melhor que um modelo seqüencial onde os projetos são passados de um lado para o outro. Portanto, em empresas que mantêm operações multiregionais ou multinacionais, uma necessidade por sistemas computacionais CAD/CAM colaborativos têm surgido, para facilitar o trabalho em equipe de forma fechada entre engenheiros localizados em diferentes regiões geográficas.

Um sistema CAD/CAM colaborativo deve prover aos projetistas, em duas ou mais localizações remotas, capacidade de trabalhar juntos sobre atividades de projeto comuns, o que significa que um novo ambiente de projeto terá de ser criado para se ter um uso total de todos os conhecimentos relevantes, tecnologias e recursos do mercado mundial. Para atingir esta tarefa desafiadora, tecnologias de rede e computadores devem ser usadas para acentuar a comunicação entre os projetistas. Por exemplo, multimídia, sistemas *whiteboard* 2D e 3D, *browsers*, WWW, ferramentas de aplicação compartilhada, etc, têm demonstrado serem de grande utilidade para acentuar as ligações entre projetistas geograficamente separados, mas ainda não são suficientes para o trabalho colaborativo.

5.1.2 CAD Colaborativos

Na prática de projeto de engenharia, cada vez mais as atividades associadas aos vários aspectos de manufatura estão sendo consideradas durante a fase de projeto. Modelagem baseada em *features* tem sido considerada como um novo paradigma para integração das atividades de engenharia, desde o projeto até a fabricação. Assim o conceito de *features* tem sido usado em uma ampla gama de aplicações como projeto de peças e montagem, projeto para manufatura, planejamento de processo e inúmeras outras aplicações. Estas aplicações estão migrando para ambientes computacionais heterogêneos e distribuídos para suportar o processo de projeto e manufatura, que serão distribuídos tanto na dimensão espacial quanto temporal.

Nota-se que é indesejável e freqüentemente improvável requerer que todos os participantes nas atividades de desenvolvimento de produto usem o mesmo sistema de hardware e software. Assim os componentes devem ser modulares e comunicar-se com os demais através de uma rede de comunicação, para efetiva colaboração.

Muitos esforços de pesquisa têm sido empregado no desenvolvimento de ambientes de projeto orientados à redes de computadores, normalmente, denominados de centrados em rede. SHAH *et al.* (1997) desenvolveram uma arquitetura para padronização da comunicação entre o núcleo de um sistema de modelagem geométrica e as aplicações. HAN e REQUICHA (1998) propuseram uma abordagem similar que possibilita o acesso transparente para diversos modeladores sólidos. WANG e WRIGHT (1998) descreveram um serviço de manufatura distribuído denominado de Cybercut (<http://cybercut.berkeley.edu>). Foi enfatizado como ferramentas de CAD firm desenvolvidas para facilitar o projeto distribuído e o processo de fabricação.

SMITH e WRIGHT (2000) descreveram um serviço de manufatura distribuído denominado de Cybercut, desenvolvido na Universidade de Berkeley (<http://cybercut.berkeley.edu>) que possibilita a concepção de uma peça prismática que será usinada utilizando-se de um sistema CAD/CAM desenvolvido em Java em um contexto de manufatura remota (ÁLVARES *et al.*, 2002 e 2003).

LI, SHAO *et al.* (2004) descreveram um sistema de projeto de produto colaborativo e inteligente orientado ao processo baseado no paradigma de projeto *Analysis-Synthesis-Evaluation* (ASE) e na parametrização do projeto de produto, baseado em agentes.

WebSpiff (BIDARRA *et al.*, 2001) baseia-se numa arquitetura cliente/servidor consistindo, no lado do servidor, de dois componentes principais: Sistema de Modelagem SPIFF que fornece toda a funcionalidade para modelagem baseada em *feature*, utilizando o *kernel* de modelagem ACIS (CORNEY e LIM, 2001); Gerenciador de Sessão que fornece funcionalidade para iniciar, associar-se, sair e fechar uma sessão de modelagem, bem como, gerência todas as comunicações entre o sistema SPIFF e os clientes.

LI *et al.* (2004, 2005), LI (2005), WONG *et al.* (2004) e FUH e LI (2004) relacionaram vários sistemas de projeto colaborativo e Engenharia Simultânea distribuídos e integrados para projeto e manufatura, sendo que nenhum dos sistemas relatados implementa atividades de projeto colaborativo integrado com sistemas de planejamento de processo e fabricação remota via *Web* para o domínio de peças rotacionais, constituídas por *features* concêntricas e não-concêntricas. A maioria dos sistemas relatados são voltados para peças prismáticas, como o WebCAD 2000 do sistema Cybercut, que não implementa projeto colaborativo (SMITH e WRIGHT, 2001).

HARDWICK *et al.* (1996) propuseram uma infra-estrutura que permite a colaboração entre companhias no projeto e manufatura de novos produtos. Esta arquitetura integra o WWW para compartilhamento de informações na Internet utilizando o padrão STEP para modelagem de produto. MARTINO *et al.* (1998) propõem uma abordagem para integrar as atividades de projetos com as demais atividades de manufatura baseada na abordagem de modelagem por *feature* integrada (híbrida), que suporta projeto por *features* e reconhecimento de *features*. Entretanto estes dois trabalhos são conceituais em sua essência e não apresentam uma representação bem estruturada e nem algoritmos detalhados. Por exemplo, estes trabalhos não definem como distribuir o processamento computacional necessário entre os componentes distribuídos, e como modular a comunicação entre os componentes para minimizar o *delay* da rede. Se as ações de troca de dados entre as aplicações não puderem ser disparadas apropriadamente, isto acarretará em um problema crítico para a computação distribuída. Logo, é crucial o desenvolvimento de um sistema bem integrado, centrado em rede e com arquitetura baseado em agentes para projeto e manufatura distribuída e cooperativa.

LEE *et al.* (1999) apresentaram a arquitetura de um sistema de modelagem baseada em *features* centrado em rede, em um ambiente de projeto distribuído, denominado de *NetFeature System*. Esta abordagem combina técnicas de modelagem baseada em *features* com tecnologia de comunicação e de computação distribuída para suportar atividades de modelagem de produto e projeto colaborativo em uma rede de computadores. A abordagem é implementada em uma arquitetura cliente/servidor, na qual os clientes realizam a modelagem baseada em *features* através da *Web*, o servidor cria o modelo de *features* neutras e outras aplicações se comunicação umas com as outras usando um protocolo de comunicação padrão para acessar os objetos remotos. O sistema foi concebido a fim de se ter um bom balanceamento entre as funcionalidades disponíveis no lado do cliente e largura de banda disponível na Internet.

O processamento no lado do cliente é importante quando a aplicação é baseada na *Web*. Isto significa que o servidor dá ao cliente alguma responsabilidade pelo processamento dos dados, ou seja, o cliente deve ter mais funcionalidades do que simplesmente um *front-end* desprovido de processamento local denominado como um de *thin clients*. LEE *et al.* (1999) utilizaram um protocolo de comunicação padronizado baseado em CORBA (*Common Object Request Broker Architecture*), que é mais complexo e difícil de implementar.

De forma semelhante às atividade de modelagem de produto, planejamento do processo e a manufatura assistida por computador na fase de planejamento (CAM Planejamento) e na teleoperação (CAM Execução) necessitam de um bom balanceamento entre largura de banda e funcionalidades do cliente, implementada na GUI.

5.2 Ferramentas CAD com Área de Trabalho Compartilhada

Segundo BIDARRA *et al.* (2001) ferramentas CAD com área de trabalho compartilhada (*workspace-sharing tools*) são aplicações que podem ser usadas como um programa *stand-alone*, mas que possuem facilidades para conectar-se com outras aplicações de modelagem, de maneira a estabelecer um ambiente de trabalho colaborativo. Podem também ser adicionados às aplicações existentes, para prover a funcionalidade necessária para um ambiente colaborativo. Algumas destas aplicações utilizam base de dados compartilhadas ou gerenciadores de sessões.

Sistemas colaborativos podem ser definidos como sistemas multi-usuários distribuídos sendo ao mesmo tempo concorrente e sincronizado (BIDARRA *et al.*, 2001). Concorrência envolve gerenciamento de diferentes processos que tentam simultaneamente acessar e manipular o mesmo dado. Sincronização envolve propagação, envolvendo dados entre usuários de uma aplicação distribuída, de forma a deixar seus dados consistentes.

Sistemas de modelagem cooperativa tipicamente tem uma arquitetura cliente/servidor, diferindo na distribuição de funcionalidade e de dados entre clientes e servidores. Um problema recorrente nos sistemas cliente/servidor está associado ao conflito entre a limitação da complexidade da aplicação cliente e a minimização do carregamento da rede. Em um contexto de modelagem cooperativa, a complexidade do cliente é principalmente determinada pelas facilidades de modelagem e interatividade implementadas no cliente, enquanto que o carregamento da rede é determinado principalmente pelo tipo e tamanho do modelo de dados sendo transferido de/para os clientes.

Uma solução de compromisso pode ser concebida entre os dois extremos, os chamados *thin clients* e *fat clients*. Uma arquitetura pura *thin client* tipicamente coloca toda a funcionalidade no servidor, o qual envia uma imagem de sua interface de usuário para ser mostrada no cliente.

O outro extremo, um puro *fat client* oferece total facilidade de interação e modelagem local, mantendo seu próprio modelo local. Comunicações com o servidor é então requerida quando houver necessidade de sincronizar as modificações de dados do modelo local com os outros clientes. Em ambiente colaborativo onde clientes podem concorrentemente modificar o modelo de dados remoto, a prevenção da inconsistências de dados entre diferentes clientes torna-se um problema crucial.

Em ambiente colaborativo todos os clientes devem ter o mesmo modelo geométrico. Quando um usuário altera o modelo, todos os clientes devem ser atualizados. Para que o processo tenha consistência as mensagens de atualização devem ser entregues na mesma ordem para todos os clientes. Este problema é caracterizado como uma questão de sincronização e consistência do modelo geométrico.

Um segundo problema está associado à concorrência e diversas soluções são adotadas nos sistemas descritos (BIDARRA *et al.*, 2001). O carregamento da rede, associado à largura de banda, ambiente computacional heterogêneo e problemas específicos de aplicações são também relacionados na análise feita por BIDARRA *et al.* (2001).

Sem dúvida a redução do carregamento da rede de comunicação é uma das principais metas para os desenvolvedores de sistema de modelagem distribuída e interativa, seguido da consistência e sincronização do modelo geométrico.

RODRIGUEZ e AL-ASHAAB (2005) apresentaram uma revisão de literatura descrevendo vários sistemas e arquiteturas para desenvolvimento de produto colaborativo, com enfoque na etapa de projeto conceitual. ZHANG *et al.* (2004) apresentaram uma revisão de literatura sobre informações de produtos comerciais e sistemas acadêmicos para projeto colaborativo e visualização baseados na Internet, destacando-se o uso de VRML.

Em ÁLVARES (2004 e 2003) é apresentada uma revisão de literatura de várias metodologias, ferramentas e sistemas CAD/CAPP/CAM e correlatos baseada na *Web*, destacando-se:

- CSCW-FeatureM: FeatureM (STORK, 1998) é um sistema CAD 3D que suporta modelagem sólida baseada em *features* utilizando o ACIS-*kernel* e uma linguagem interpretada denominada de MCL+, que permite ao usuário a adição de funcionalidade;

- CollIDE: O sistema *Collaborative Industrial Design Environment* (CollIDE) desenvolvido por NAM (1998) oferece uma área de trabalho 3D compartilhada para muitos usuários de uma aplicação *stand-alone* CAD;
- TOBACO: *Tool Based Co-operation* (TOBACO) foi desenvolvido por DIETRICH (1997). Fornece um componente integrado no topo de programas de projeto existentes para habilitá-los a participar na modelagem colaborativa;
- ARCADE: o objetivo dos desenvolvedores (STORK, 1997) do ARCADE (*Advanced Realism CAD Environment*) foi criar um método de discussão aprimorado para ser usado sobre a Internet. O método permite que muitos usuários trabalhem juntos sobre um projeto, interagindo um com o outro em tempo real;
- Modelagem Sólida Cooperativa (CSM): o sistema CSM (*Collaborative Solid Modelling*) é um sistema de modelagem cooperativa baseado na Web (CHAN e WONG, 1999), que usa muitos métodos de concorrência;
- NetFeature (LEE, 1999): é um sistema de modelagem baseado em *features* e na Web. Diversos *browsers* distribuídos, clientes, conectam-se ao servidor de modelagem de *features* neutro;
- CyberView (KIM, 1988): usa modelagem assíncrona combinada com visualização via *Web browser*;
- Prototipagem Rápida Baseada na Web: a URL <http://tmf.sdsc.edu> apresenta um projeto da pesquisa e desenvolvimento chamado *Tele-Manufacturing Facility* (TMF). O TMF oferece um serviço de prototipagem rápida (RP) automatizado baseado na Internet. O sistema disponibiliza um serviço para que usuários remotos tenham a possibilidade de fabricarem um protótipo de uma peça a partir do arquivo em formato “stl” (estereolitografia). Em LUO *et al.* (1999) e LAN *et al.* (2004) são descritas a arquitetura de outros sistemas de prototipagem rápida baseados na Web;
- *Manufacturing Advisory Service*: é uma ferramenta computacional para auxiliar o projetista durante o ciclo de desenvolvimento de produto, em especial na seleção de material e processo de manufatura, desenvolvida por SMITH e WRIGHT (2000) na sua tese de doutorado;
- WELD (*Web-based Electronic Design Environment*): fornece uma estrutura de plataforma independente para acoplamento de aplicações e ferramentas CAD existentes, de maneira a criar um ambiente de projeto colaborativo (CHAN e SPILLER, 1998);
- COCA (*Collaborative Objects Coordination Architecture*): é uma arquitetura de coordenação genérica que permite aos usuários utilizarem regras sociais em um sistema colaborativo (LI, 1998);
- DistView (PRAKASH e SHIM, 1994): é um *toolkit* que pode ser usado para desenvolver aplicações colaborativas para uso sobre redes de comunicação de área (WAN);

- Sistema de Telemanufatura WebOxicorte: é um sistema integrado CAD/CAPP/CAM desenvolvido para programação da máquina de oxi-corte CNC e sua teleoperação via Web (<http://weboxicorte.graco.unb.br>) em um contexto de telemanufatura (ÁLVARES *et al.*, 2002);
- Wise-ShopFloor (WANG *et al.*, 2004), iShopFloor e eShopFloor (HAO *et al.*, 2005): é uma metodologia com arquitetura aberta para monitoração em tempo real e controle remoto de máquinas CNC via Web. Segundo HAO *et al.* (2005) apenas 10% das máquinas CNC no mundo têm capacidade para conectar-se diretamente à Internet.

A seguir são descritos os dois principais sistemas que influenciaram o desenvolvimento desta tese, WebSpiff e Cybercut.

5.2.1 WebSpiff (<http://www.webspiff.org>)

WebSpiff (BIDARRA *et al.*, 2001) é baseado em uma arquitetura cliente/servidor consistindo de muitos componentes. No lado do servidor tem-se dois componentes principais:

1. Sistema de Modelagem SPIFF: fornece toda a funcionalidade para modelagem baseada em *feature*, utilizando o *kernel* de modelagem ACIS;
2. Gerenciador de Sessão: fornece funcionalidade para iniciar, associar-se, sair e fechar uma sessão de modelagem, bem como, gerência todas as comunicações entre o sistema SPIFF e os clientes.

Os componentes do portal WebSpiff fornecem o acesso inicial a uma sessão WebSpiff para um novo cliente, que inclui um servidor Web onde os dados do modelo é disponibilizado para *download* pelos clientes. Os clientes executam operações localmente, associado com a visualização, interação com o modelo em *features*, mensagens semânticas de alto nível, especificação de operações de modelagem, bem como, atualização de dados dos clientes enviados via rede.

O servidor coordena a sessão de colaboração mantendo um modelo de produto centralizado e provê todas as funcionalidades que não podem, ou não devem, ser implementados no cliente. Uma importante vantagem desta arquitetura é que existe apenas um modelo de produto centralizado, evitando-se inconsistências entre múltiplas versões do mesmo modelo.

5.2.2 WebCAD3D e CyberCut

WebCAD3D (WANG e WRIGHT, 1998) é um sistema de manufatura “rápido” que faz uso do WWW como mecanismo de integração. O sistema tenta integrar os ambientes de projeto e manufatura distribuído, através do projeto criado em um cliente Web e enviado para um servidor de manufatura sobre a Internet. No servidor de manufatura, uma aplicação envia os comandos necessários para uma fresadora CNC.

O sistema WebCAD3D é parte do sistema denominado de CyberCut (<http://cybercut.berkeley.edu>) que realiza a integração CAD/CAPP/CAM para fabricação de peças prismáticas a

partir do WebCAD3D que é um *applet*, iniciado a partir de um *browser*. Segundo WANG e WRIGHT (1998) uma comunidade de manufatura globalizada será o futuro do ambiente de manufatura. Cada membro da comunidade irá fornecer seus próprios serviços e ferramentas.

O sistema é disponível para os usuários como um Java *applet* que pode ser usado em combinação com um *Web browser*. É usado a Modelagem Sólida Destrutiva (DSM) de maneira a refletir o processo de fabricação que trabalha com operações de fresamento e furação, utilizando o conceito de *features* de usinagem/manufatura (subtrativas).

Como um meio termo entre os ambientes de manufatura e projeto, uma variação do DSM, o DSM *Solid Interchange Format* (SIF) é usado. Este contém informações adicionais da peça, como tolerâncias e informações específicas de usinagem, como raio de canto. O cliente faz uso de um conjunto pré-definidos de *features* de usinagem, armazenadas em uma biblioteca, que podem ser fabricadas na máquina disponível. Esta biblioteca de projeto está localizada no servidor.

Capítulo 6

WebMachining: Integração CAD/CAPP/CAM Baseada na Web

Este capítulo descreve a metodologia concebida para integração CAD/CAPP/CAM voltada para manufatura remota de peças rotacionais utilizando a Internet, em especial os protocolos associados à *Web* (SOA), sendo denominada de WebMachining (<http://WebMachining.AlvaresTech.com>).

A metodologia WebMachining apresenta uma arquitetura e arcabouço para implementação do sistema integrado CAD/CAPP/CAM aplicado à manufatura remota de peças rotacionais, consistindo de um conjunto de especificações, técnicas e algoritmos utilizados para a definição funcional dos módulos do sistema. A metodologia é concebida a partir do paradigma de projeto colaborativo baseada em síntese por *features* de projeto (*features* aditivas e subtrativas), permitindo a integração das atividades de projeto (CAD), planejamento do processo (CAPP) e manufatura (CAM Planejamento e CAM execução).

Esta metodologia especifica os módulos do sistema e como estes módulos interagem entre si, sendo apresentado na forma de diagramas IDEF0, IDEF1X e UML, para modelagem funcional e de informação, respectivamente (Apêndice B.2).

6.1 Metodologia WebMachining

A metodologia é concebida a partir do paradigma de modelagem baseada em *síntese por features de projeto* (*adição de features para operações de torneamento e subtração de features para operações de fresamento e furação*), a fim de permitir a integração das atividades de projeto colaborativo (CAD), planejamento do processo (CAPP) e fabricação (CAM Planejamento e CAM execução) utilizando como referência o modelo de *features* de manufatura definido por Part 224 e mais especificamente a taxonomia de *features* de forma para peças rotacionais definido por CAM-I (1986). O procedimento inicia-se na modelagem colaborativa de uma peça por *features* num contexto de manufatura remota utilizando a *Web* como meio de comunicação, num modelo computacional cliente/servidor.

Os clientes conectam-se ao Modelador de *Features* Neutro via *Web* e iniciam a instanciação de uma nova peça a ser modelada, de forma colaborativa, a partir de uma base de dados, usando a biblioteca de *features* padronizadas disponibilizada pelo sistema. Após a conclusão e validação dos

modelos de produto por *features* e geométrico, a peça criada é armazenada e disponibilizada para os demais módulos do sistema.

O sistema é concebido num ambiente distribuído de agentes de *softwares* interoperáveis denominado de Comunidade de Agentes de Manufatura, sendo sua arquitetura estratificada em três camadas: Projeto, Planejamento do Processo e Fabricação. O fluxo de informações entre as três camadas da arquitetura é realizado através do encapsulamento das informações, adotando-se o *Knowledge Query and Manipulation Language* (KQML) como padrão de linguagem de mensagens entre os agentes.

Na camada superior reside um grupo de Agentes de Projeto (agentes de interface), os quais atuam como uma ferramenta CAD colaborativa baseada em *features* e também permite aos usuários que se conectem aos níveis inferiores. Este módulo é denominado *WebCADbyFeatures*.

A camada intermediária consiste do agente de planejamento que incorpora as características de um CAPP generativo utilizando métodos de representação do conhecimento baseados em inteligência artificial, destacando-se as regras de produção, teorias de grafos e algoritmos genéticos. Este grupo de agentes interpreta as definições de projeto, modelagem por *features* e geométrica realizada pelo usuário, e determina como fabricar a peça usando o agente de fabricação (WebTurning), que reside na camada inferior. O módulo intermediário é denominado WebCAPP.

O agente de fabricação (WebTurning) permite a teleoperação da máquina de comando numérico via Internet e a monitoração do processo de usinagem através de *streaming* de vídeo e áudio. As informações sobre recursos de fabricação disponíveis (máquinas, ferramentas, informações de materiais, dispositivos de fixação, etc) estão disponíveis através de uma base de dados relacional que é acessada pelos diversos agentes do sistema. Este módulo é denominado de WebTurning.

As mensagens são trocadas usando conexões *sockets* diretas (ÁLVARES e ROMARIZ, 2002) entre os agentes. A metodologia idealizada utiliza a tecnologia Internet e de comunicação associada à SOA para oferecer um novo paradigma para o desenvolvimento dos futuros ambientes integrados CAD/CAPP/CAM. Este ambiente é globalizado, centrado em rede, colaborativo e espacialmente distribuído, tendo como *front-end* com o sistema CAD/CAPP/CAM navegadores baseados na *Web* e as linguagens HTML, Java e VRML (*Virtual Reality Modeling Language*), permitindo a independência da plataforma computacional do usuário.

A originalidade desta arquitetura está associada à contribuição metodológica para o desenvolvimento de sistemas de telemanufatura via Internet (*e-Mfg*) a partir da concepção (projeto detalhado colaborativo) de um produto (peça) utilizando a abordagem de síntese por *features* de projeto (*features* aditivas e subtrativas) até a fabricação da peça, usando a Internet como meio de conexão entre as várias tecnologias avançadas de manufatura utilizadas. Ou seja, inicia-se o processo de desenvolvimento de produto colaborativo utilizando o agente de interface CAD (*WebCADbyFeatures*) orientado por *features*, trabalhando em um ambiente distribuído e colaborativo, gerando o modelo geométrico e de *features* da peça (projeto detalhado); o planejamento de processo com alternativas e o código G (*WebCAPP*); e a seguir a teleoperação da máquina-ferramenta CNC (*WebTurning*).

A metodologia pode ser aplicada para manufatura de peças de rotacionais, bem como para manufatura de peças prismáticas. A metodologia e sua implementação irá concentrar-se com maior profundidade na modelagem colaborativa, planejamento de processo e fabricação de peças rotacionais simétricas e assimétricas.



Figura 6.1: Camadas da arquitetura multiagente MAC.

6.2 Arquitetura Multiagente

Utiliza-se de diversos tipos de agentes trabalhando colaborativamente e de forma distribuída na resolução dos diversos problemas associados à integração CAD/CAPP/CAM em um contexto de uma comunidade de agentes. Utiliza-se um agente de interface na modelagem de produto colaborativo (GUI) associado a um sistema de gerenciamento de base de dados relacional (MySQL® ou SQL®), para compartilhar as informações dos recursos disponíveis de manufatura (máquinas, ferramentas, informações de materiais, dispositivos de fixação, etc). Têm-se também agentes como os resolvedores das atividades de planejamento do processo. Os agentes são implementados utilizando-se de diversas abordagens na sua lógica de decisão: regras de produção (if/then), teoria de grafos, algoritmos genéticos e raciocínio lógico. Utiliza-se da linguagem de comunicação de agentes KQML, como linguagem que os agentes usam para se comunicar.

Este ambiente em rede de agentes de *softwares* interoperáveis é denominado de Comunidade de agentes de Manufatura (MAC). A figura 6.1 apresenta a arquitetura MAC estratificada em três camadas. Na camada superior reside um grupo de agentes de projeto, os quais atuam como ferramentas CAD e também permite aos usuários que se conectem às camadas inferiores. A camada intermediária consiste do agente de planejamento. Este agente interpreta as definições de projeto, modelagem geométrica, realizada pelo usuário, e determina como fabricar a peça usando o agente de fabricação (máquina-ferramenta) selecionado, o qual reside no nível inferior. É utilizado uma interface transparente entre o projeto, planejamento e fabricação.

A partir deste fluxo de informações entre as três camadas da arquitetura é necessário realizar o encapsulamento destas informações, sendo adotado o KQML (<http://www.cs.umbc.edu/kqml>) como padrão de linguagem de mensagens entre os agentes. O KQML contém informações sobre quais agentes enviam mensagem, onde eles estão e como interpretar a mensagem recebida pelo destinatário. As mensagens são trocadas usando conexões *sockets* diretas entre os agentes.

A arquitetura do sistema Multiagente WebMachining é apresentada na figura 6.2, sendo utilizado o ambiente de desenvolvimento de agentes JATLite baseado na linguagem KQML. A arquitetura idealizada é constituída por dez agentes agrupados em seis grupos: facilitador (1), gerenciador de base de dados (2), projeto colaborativo (3), modelagem 3D VRML (4), planejamento do processo (5)

e fabricação (6,7,8,9, 10), conforme descrição a seguir:

1. AGENTE FACILITADOR (AF): realiza o gerenciamento da comunicação entre os agentes, administrando o roteamento das mensagens entre os agentes, segurança do sistema e o registro de agentes, por exemplo. É implementado através do *Agent Message Router* da arquitetura JATLite, sendo um agente disponibilizado pelo ambiente JATLite. Deve haver pelo menos um AF definido na arquitetura. Pode ser necessário mais de um AF em função da quantidade de agentes presentes no sistema, a fim de melhorar a performance do mesmo. O maior gargalo do ambiente de desenvolvimento JATLite é o AMR, pois os agentes sempre se comunicam uns com os outros via AMR (*middleware*).
2. AGENTE GERENCIADOR DE BASE DE DADOS (ADB MySQL): este agente faz a interação com a base de dados MySQL. Qualquer agente que deseja alguma informação disponibilizada pela base de dados (linguagem SQL) faz uma solicitação para o Agente Gerenciador de Base de Dados, e este envia a resposta para o agente que solicitou a informação. O Agente Facilitador realiza o roteamento de mensagens entre estes agentes.
3. AGENTE DE INTERFACE CAD - PROJETO COLABORATIVO (AICAD COLABORATIVO - WEBCADBYFEATURES): a interface gráfica com o usuário (GUI) para projeto colaborativo por *features* é implementada através de um *applet* Java. Assim toda GUI executada por um cliente remoto, para definir o modelo de *features* e geométrico da peça bruta e da peça acabada (modelagem do produto), possui um AICAD incorporada à interface. Este agente (AICAD) irá se comunicar com a comunidade de agentes através de uma conexão ao AF, e este fará o roteamento da mensagem para o agente em questão. São enviadas mensagens para os demais módulos do sistema comunicando os dados referentes ao projeto em desenvolvimento (modelo de produto) como: usuário, nome da peça, nome do projeto, entre outros; que permitirá a identificação do modelo de produto que o cliente está elaborando. A conexão com a base de dados MySQL é realizada diretamente via *servlets*, a fim de melhorar o desempenho do sistema, não usando o ADB para esta finalidade. O cadastramento do usuário na base de dados de usuários remotos é feito via PHP. A visualização 3D do modelo de produto é gerenciada via AICAD, que se comunica com o agente de modelagem 3D, através de VRML.
4. AGENTE DE MODELAGEM 3D BASEADO EM VRML (AVRML): é responsável pela modelagem 3D utilizando VRML via *servlets*. Recebe mensagens do AICAD para construção de modelos 3D em VRML das peças modeladas por *features*. A seguir o *servlet* VRML envia o modelo da peça em VRML para o cliente, onde o mesmo é visualizado no *browser*, através de um *plug-in* VRML previamente instalado.
5. AGENTE PLANEJAMENTO DE PROCESSO (ACAPP): é responsável pelo planejamento de processo e geração do programa NC, sendo chamado pelo agente usuário remoto para gerar o plano de processo linearizado via *applet* ou *servlet*, após a conclusão da modelagem colaborativa da peça. São executadas as seguintes atividades: mapeamento de *features* de projeto para *features* de manufatura (usinagem); determinação das operações de usinagem com alternativas, associadas às *features* de usinagem: *workingsteps* de usinagem e de torneamento; determinação

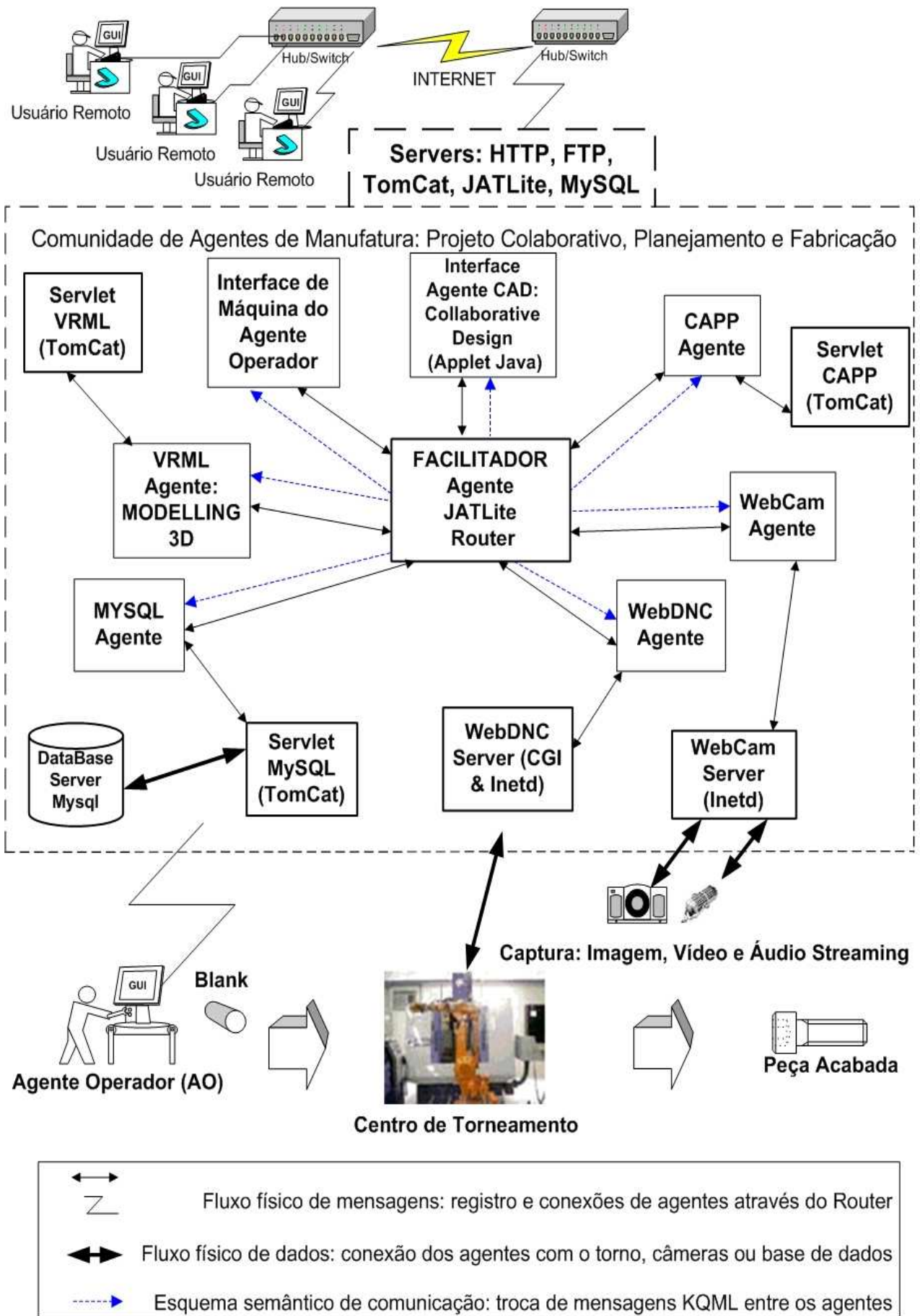


Figura 6.2: Arquitetura Multiagente WebMachining.

da seqüência de usinagem com alternativas: *workplan* não-linear; estratégias para geração de trajetórias de ferramentas; determinação das ferramentas de corte (insertos e suportes); determinação do modelo de tempos e cálculo dos tempos padrão e custo padrão para cada *workingstep*; determinação das condições tecnológicas de usinagem otimizadas, utilizando algoritmos genéticos; linearização do plano de processos não-linear, utilizando algoritmos genéticos; geração do programa NC (ISO 6983); geração de relatórios e plano de processo.

6. AGENTE DE INTERFACE CAM COM O USUÁRIO REMOTO (AICAM - WEBDNC): toda GUI associada ao CAM, executada por um cliente remoto e utilizada para fazer a teleoperação da máquina-ferramenta CNC, possui um AICAM incorporada à interface. Este agente (AICAM) irá se comunicar com a comunidade de agentes através de uma conexão ao AF, fazendo o roteamento da mensagem para o agente em questão.
7. AGENTE WEBCam (WebCam): é responsável pela captura de vídeo, imagem e áudio do processo de usinagem, enviando *streamings* de vídeo e áudio diretamente para a GUI associada ao CAM (AICAM - WEBDNC).
8. AGENTE WEBCNC (WebCNC): é responsável pelo controle remoto da máquina CNC, recebendo comandos e enviado o status da máquina para a GUI associada ao CAM, WebDNC. Recebe mensagens do AF referente à identificação do usuário, *login* e senha, nome do arquivo com o programa NC e dados de planejamento do processo (fixação, ferramentas e peça bruta), sendo responsável pela implementação do protocolo de DNC através da *Web*.
9. AGENTE INTERFACE COM O OPERADOR DA MÁQUINA (AIO): esta interface gráfica (GUI) instrui o operador no chão-de-fábrica, sendo implementada através de um *applet* Java. Este agente de interface com o operador (AIO) dá as instruções para o operador sobre fixação da peça bruta, *setup* de ferramentas, preparação da máquina, agendamento da fabricação de uma peça, entre outras.
10. AGENTE OPERADOR DA MÁQUINA (AO): este agente (AO) é o operador da máquina-ferramenta no chão-de-fábrica, que recebe instruções de fixação da peça bruta, *setup* de ferramentas, preparação da máquina, agendamento da fabricação de uma peça e outros dados associados ao planejamento de processo, e que só podem ser tratados por um operador humano.

No item seguinte é apresentado um detalhamento da arquitetura proposta através da abordagem IDEF0 (Apêndice B.2), onde são descritas as especificações funcionais de cada módulo do sistema.

6.3 Sistema WebMachining: Modelagem IDEF0

O sistema integrado CAD/CAPP/CAM denominado de WebMachining, foi modelado através da metodologia IDEF0. A figura 6.3 representa o nível 0 do modelo IDEF0 com todas as entradas, controles, mecanismos, saídas e funcionalidades que são propostas pela metodologia WebMachining.

A figura 6.4 apresenta a estratificação do sistema nas três camadas associadas aos agentes de *softwares* interoperáveis denominada de Comunidade de Agente de Manufatura, sendo as atividades:

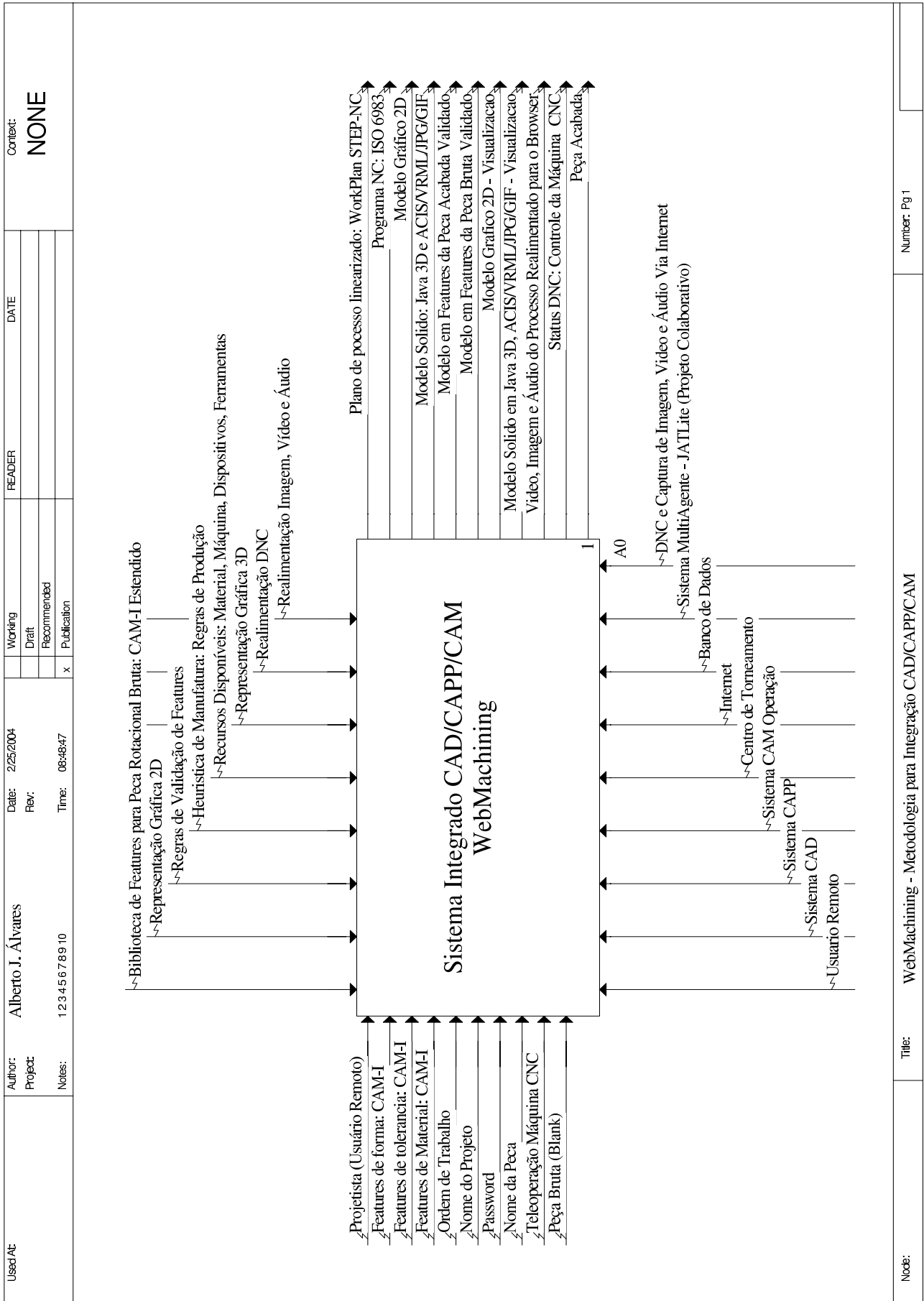


Figura 6.3: Diagrama IDEF0: WebMachining nível A0.

1. WebCADbyFeatures.
2. WebCAPP.
3. WebTurning.

A atividade WebCADbyFeatures tem como entrada de dados a modelagem por *features* e outras informações pertinentes, e como saída o modelo de *features* e geométrico da peça bruta e da peça acabada, que serve com entrada de dados para o CAPP. A atividade WebCAPP tem como saída a geração do plano de processos linearizado e o programa NC, além da lista de *workingsteps* (*features* de usinagem e de torneamento associadas às operações) e o *workplan*. A atividade WebTurning tem como entrada a peça bruta (*blank*), os dados referentes à teleoperação da máquina CNC, programa NC (DNC) e *setup* da peça, zero peça e de ferramentas; e como saída o controle da máquina, *streamings* de vídeo e áudio do processo e a peça usinada.

A seguir são apresentados os diagramas IDEF0 associados às três atividades da metodologia e sistema concebido, denominado de WebMachining¹.

6.3.1 Atividade WebCADbyFeatures

A figura 6.5 apresenta o modelo IDEF0 referente à atividade WebCADbyFeatures. É constituído por sete atividades permitindo o projeto colaborativo através da criação e manipulação do modelo de *features* para a peça bruta e acabada, armazenamento em um banco de dados, validação do modelo e visualização do modelo geométrico em 2D e 3D:

1. Projeto colaborativo: cria modelo da peça utilizando classes de *features*, disponíveis na biblioteca de *features*;
2. Validação do modelo de *features* referente instanciação da peça;
3. Banco de dados associado ao modelo da peça instanciada por *features*;
4. Modelador gráfico 2D - lado cliente;
5. Modelador gráfico 3D - lado servidor;
6. Visualização do modelo gráfico 2D na GUI do usuário - *browser* e Java;
7. Visualização do modelo gráfico 3D na GUI do usuário - *browser*, VRML e Java.

A GUI disponibiliza as funcionalidades necessárias para suportar modelagem colaborativa baseada em *features*. Esta interface CAD está associada a menus, opções de visualização (2D e 3D/Vrml), *prompts* e mensagens de erro, interações com o usuário de forma gráfica e textual, seleção de *features* (editar e apagar) e de bibliotecas de *features*, instanciação de *features* (aditivas e subtrativas), manipulação das *features*, ler e escrever (armazenar) o modelo de *features* localmente e remotamente, entre outros.

¹Na URL <http://WebMachining.AlvaresTech.com/idef0> é apresentado o IDEF0 em versão html, facilitando a navegação e entedimento do modelo funcional do sistema WebMachining.

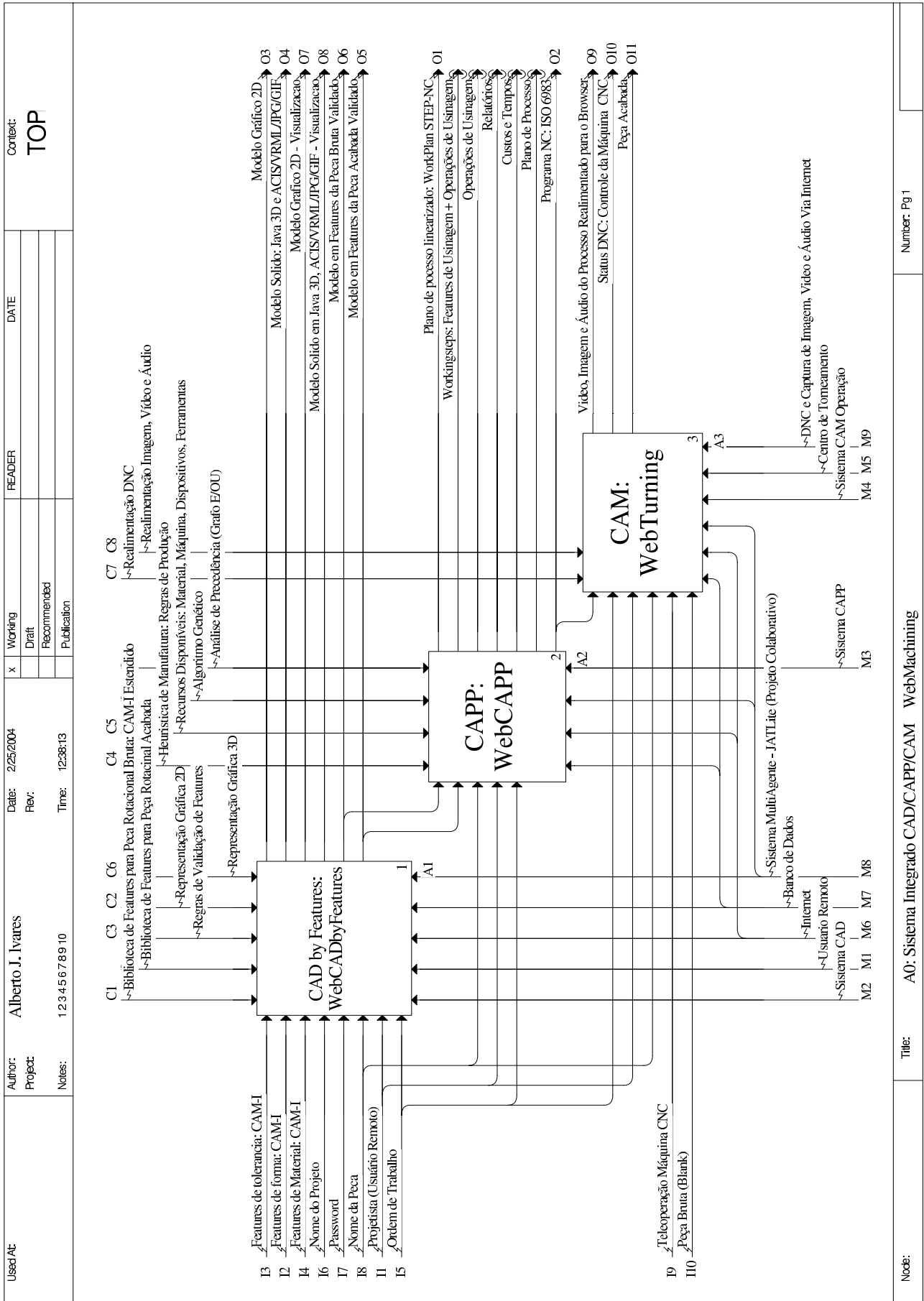


Figura 6.4: Diagrama IDEF0: principais atividades do sistema WebMachining.

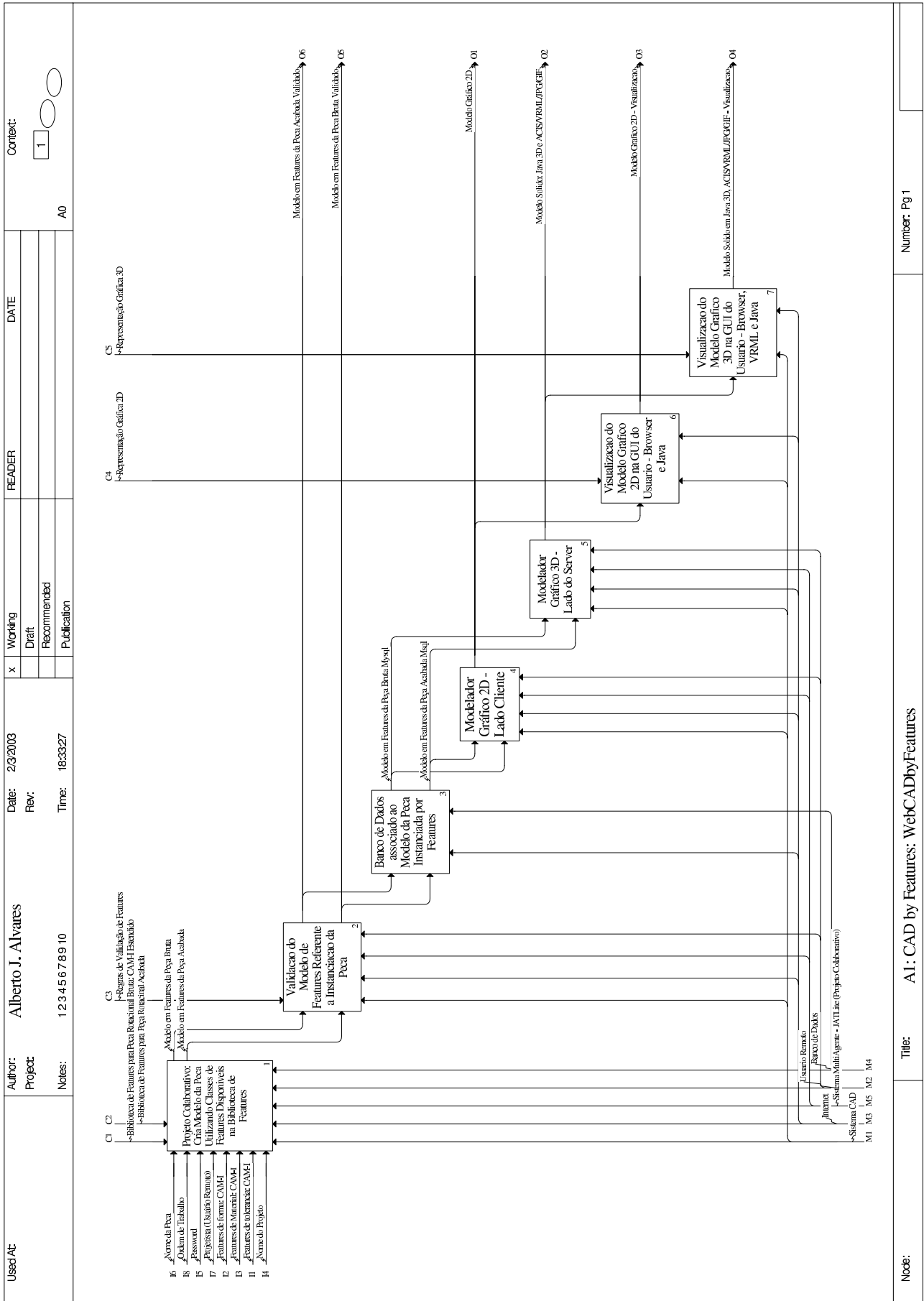


Figura 6.5: Diagrama IDEF0: Atividades WebCADbyFeatures.

A figura 6.6 apresenta os componentes da interface gráfica para modelagem colaborativa. Os principais componentes são: Interface Gráfica com o Usuário (GUI), a biblioteca de *features*, Interface Gráfica 2D, componentes de visualização 2D (primitivas gráficas) processadas localmente na própria interface, componentes de visualização 2D (animação de arquivos jpg) e 3D (*Kernel ACIS* e *VRML*) com processamento remoto, *Browser*, *plug-ins Java*, *plug-ins VRML*, *Applets Java* e *sockets TCP/IP*.

A GUI permite a modelagem colaborativa de peças rotacionais via Internet usando a abordagem de projeto por *features* integrado a um modelador de sólidos baseado em *VRML* via *servlets*. Pode-se utilizar também o *kernel ACIS* e um conversor de formato SAT para *VRML*, para modelagem de sólidos (ÁLVARES *et al.*, 2004). As informações referentes às *features* são manipuladas localmente através de um objeto serializado que armazena a biblioteca de *features*. Esta biblioteca de *features* está disponível também através de um sistema de gerenciamento de base de dados relacional, que a armazena remotamente em um servidor.

A GUI é implementada em Java e HTML, disponibilizada por *browser*, onde o usuário introduz as informações sobre a *feature* que deseja inserir na peça, e estes dados são encaminhados ao *Servlet*, que executa a função em Java para geração do modelo 3D (*VRML*) da peça; ou ao servidor em C++ para a criação do sólido no *ACIS* via *CGI (Common Gateway Interface)*. Para o usuário visualizar a peça em seu *browser*, o formato do sólido no *kernel ACIS* é transformado para o formato *VRML*, e em seguida enviado para a GUI cliente, *browser* que faz a visualização do modelo *VRML* através de um *plug-in VRML*. Rotinas em C++ e java, em função do servidor, são desenvolvidas para manter a consistência da peça que está sendo criada. Como a peça é rotacional, o usuário visualiza a peça em 2D durante a modelagem colaborativa.

As informações referentes às *features* são gerenciadas em uma base de dados *MySQL*, que armazena as informações sobre a modelagem de produto por *features*, contendo informações associadas às *features* de forma, *features* de material, *features* de tolerâncias e *features* tecnológicas (tratamento superficial, tratamento térmico e dados de produção). Estas informações conjugadas permitem o mapeamento das *features* de projeto para as *features* de usinagem e operações de usinagem associadas, que é fundamental para o planejamento do processo da peça. As informações do banco de dados são descritas utilizando-se o modelo de informações do *IDEFIX*, conforme apresentado nos Apêndices B.3 e na URL <http://WebMachining.AlvaresTech.com/idefix>.

A figura 6.7 apresenta a taxonomia de *features* de forma utilizada pelo sistema *WebMachining*, sendo baseada na proposta *CAM-I (1986)*. As *features* de forma concêntricas podem ser vistas, por exemplo, como *features* de segmento, cilíndricas, arcos, *splines*, cônicas e transição, que são utilizadas na implementação da metodologia, sendo associadas no planejamento de processo às operações de torneamento, faceamento, mandrilamento, rosqueamento, faceamento, entre outras, pela atividade *WebCAPP*. Têm-se também as *features* de forma não-concêntricas associadas à ferramenta acionada de um centro de torneamento, permitindo a execução de operações de fresamento e furação com Eixo C nas direções radial e axial.

A figura 6.8 apresenta o diagrama de classes (UML) associado à classe *Projeto*, que está relacionada com as informações geométricas, classe *Blank* (peça bruta) e classe *PeçaAcabada*; e não-geométricas, como as classes *UsuárioAgenteInterface* e *DadosdeProdução*. A classe *PeçaAcabada* está associada às *features* concêntricas e não-concêntricas (classe *Feature*). E a classe *Feature* está

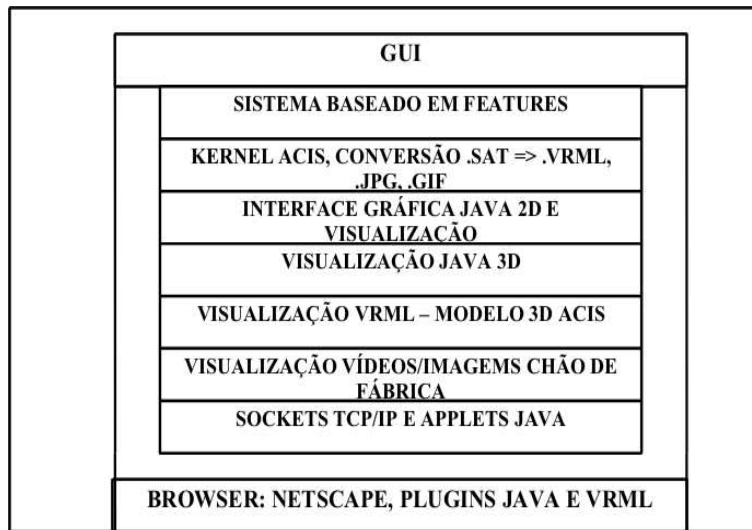


Figura 6.6: Componentes GUI.

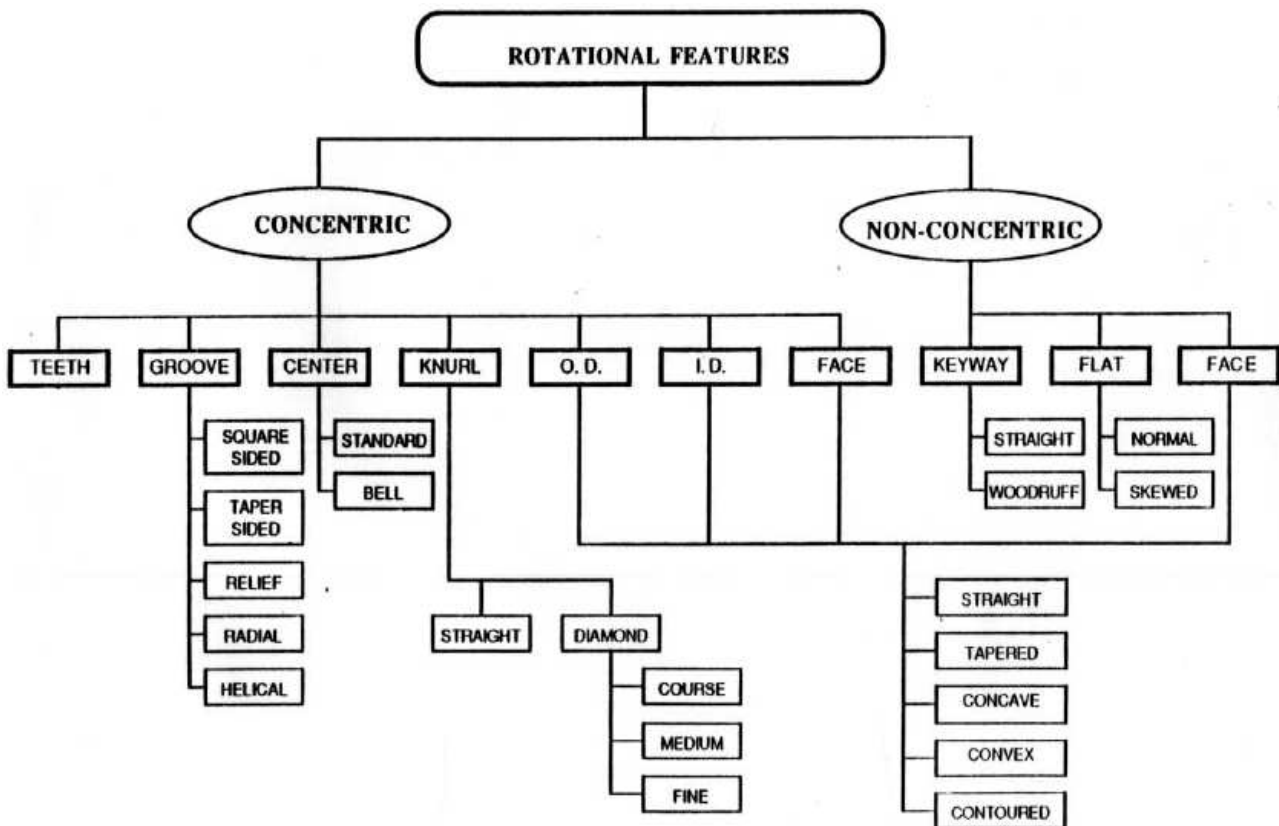


Figura 6.7: Taxonomia de *features* de forma utilizada (CAM-I, 1986).

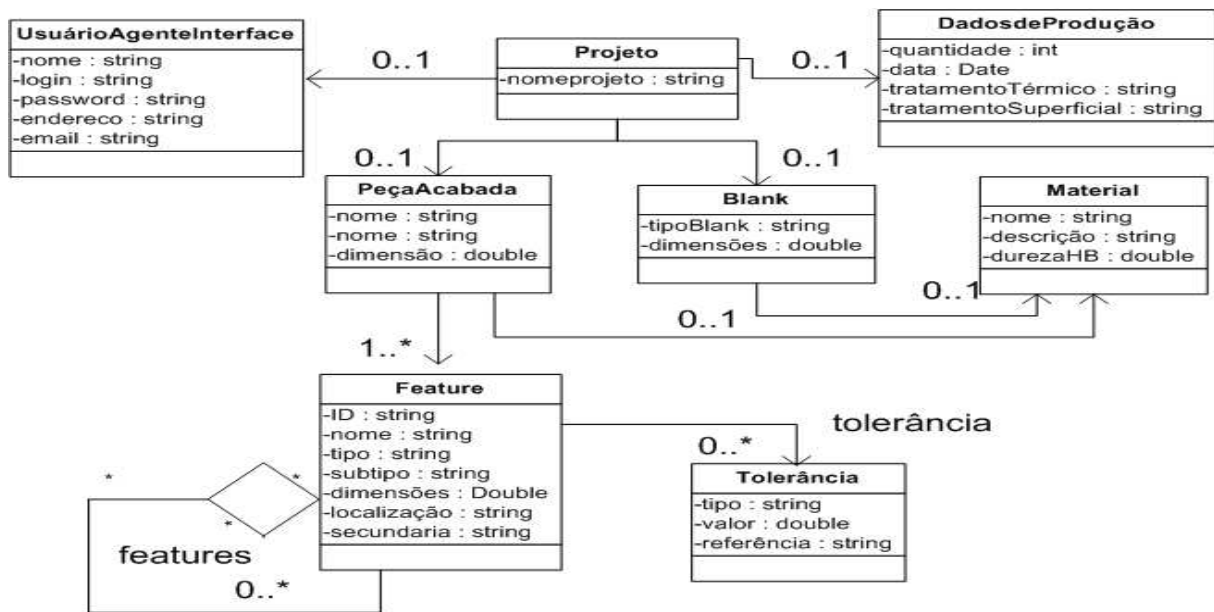


Figura 6.8: Diagrama de classes para *Projeto*.

associada à classe de *Tolerância*.

Pode-se realizar a modelagem da peça para a região externa (perfil externo, *features* concêntricas), região interna (perfil interno, *features* concêntricas) ou Eixo C (ferramenta acionada, *features* não-concêntricas).

A modelagem da peça tem início com a instanciação das *features* concêntricas externas, que sempre iniciam a partir de uma das extremidades, face esquerda ou face direita. As *features* internas são instanciadas após a conclusão da instanciação das *features* que definem o perfil externo da peça, indicando o seu ponto de início, que pode ser a partir da extremidade esquerda, região central ou da extremidade direita da peça. Assim é necessário identificar sua posição inicial através de uma coordenada cartesiana, que no caso está associada ao eixo Z, assumindo a modelagem no sistema cartesiano ZX, que é o sistema de coordenadas da máquina-ferramenta (Z,X). Já se utiliza as coordenadas ZX e não XY para evitar a necessidade de mapeamento do sistema de coordenadas de projeto para o programa NC.

São armazenadas informações sobre a ordem, precedência de *features* de forma no momento de sua instanciação (criação), mostrando o relacionamento com as *features* adjacentes. Para cada *feature* de forma instanciada pode-se incluir as *features* de tolerâncias associadas (forma, posição, orientação, localização, batimento, dimensional e rugosidade) e as *features* de referências (*datum*) de projeto.

A modelagem das *features* concêntricas pode ter início a partir do sentido da extremidade esquerda para a extremidade direita ou vice-versa, sendo adotado como padrão a opção modelagem da extremidade esquerda para direita. Assim o modelo de *features* para a peça acabada e para a peça bruta é constituído por uma lista de *features* concêntricas, sendo a lista ordenada segundo a precedência das *features* no modelo de *features* instanciado para a peça.

Após a modelagem das *features* concêntricas, inicia-se a modelagem das *features* não-concêntricas, associadas ao Eixo C (ferramenta rotativa) do centro de torneamento. Pode-se modelar *features* de Eixo C, com ferramentas rotativas radiais e axiais. O método utilizado é baseado no projeto por sín-

tese de *features* subtrativa, ou seja, definem-se os volumes de material a serem removidos a partir da seleção de ferramentas rotativas, fresamento ou furação, que irão gerar as *features* de projeto do tipo cavidade, rasgo de chaveta, furos, padrões de furação, entre outros. Roscas e escareados podem ser associados como atributos de *features* de furos não-concêntricos, gerando assim *features* secundárias. Pode-se ter também furos como atributos de *features* de cavidades, e estes furos poderão ter também atributos de escareados e rosca, por exemplo.

Outro importante atributo relacionado às *features* está associado à referência de projeto, que é utilizada como referência para o sistema de cotação, normalmente localizado nas extremidades da peça, face esquerda ou direita. Caso uma das *features*, referentes à extremidade esquerda ou direita, esteja associado ao sistema de referência de projeto (sistema de origem de cotação das dimensões de projeto), recomenda-se que o sentido de modelagem de *features* deve ser coincidente com o sistema de referência de projeto. O sistema de referência de usinagem deve coincidir com o sistema de referência de projeto, a fim de atender o princípio de coincidência de referências, evitando assim a transferência de cotas.

As *features* têm associado aos atributos dimensionais, comprimentos das *features* (eixo Z), mais um atributo relacionando esta dimensão de projeto ao sistema de referência de projeto ou à uma cota associada à *feature* adjacente predecessora.

As informações referentes às ferramentas, sistemas de fixação de peças e ferramentas, condições de usinagem, dados de máquinas, dados de produção, entre outros, são armazenadas na mesma base de dados, permitindo seu compartilhamento entre os módulos do CAPP, para tomada de decisão.

O WebCADbyFeatures fornece os dados para o sistema de planejamento de processo. Para facilitar a entrada de dados pelo projetista utiliza-se na GUI a figura de menus oferecendo uma interface amigável e confortável para a instanciação de uma *feature*, permitindo a definição de seus atributos, tolerâncias, relacionamentos e especificações, conforme descrito. Para esta finalidade utiliza-se do conceito de *features* no nível do usuário e *features* no nível do modelo. Assim o usuário projeta a peça com um simples conjunto de *features* no nível do usuário disponibilizadas pela GUI, *applet*. Estas *features* são convertidas para *features* no nível do modelo com maior detalhamento, incluindo outras informações de maneira implícita ao modelo de *features* no nível do usuário, como coordenadas inicial e final da *feature*, (Z_i, X_i) e (Z_f, X_f) obtidos indiretamente através do modelo geométrico da peça e não como um atributo em um menu, numeração lógica das *features*, amarrando a precedência das *features*, entre outras.

6.3.2 Atividade WebCAPP

O Plano de processo não-linear gerado pelo CAPP é representado utilizando uma estrutura de dados baseada em STEP-NC (ISO 14649, Part 12 e 11), que é um novo padrão para transferência de dados entre sistemas CAD/CAM e máquinas CNC (XU e HE, 2003). A arquitetura para o módulo CAD é baseada no paradigma de Projeto por *Features*. Já os módulos CAPP e CAM idealizados podem ser caracterizados segundo as quatro dimensões definidas por SHAH e MANTYLA (1995) da seguinte forma:

1. na dimensão de planejamento é voltado para geração de planos de processos não-lineares para peças rotacionais utilizando centros de torneamento, contendo definição de operações,

- fi-xação/*setup*, seqüências de operações, ferramentas, parâmetros de corte, geração do código G e estimativas de tempos e custos, ou seja, microplanejamento;
2. na dimensão de planejamento associado ao tempo pode ser aplicado no nível tático e no nível operacional de um sistema de Gestão da Produção;
 3. na dimensão método de planejamento é caracterizado com sendo um CAPP Generativo;
 4. e na dimensão profundidade de planejamento é definido como dinâmico, ou seja, planos podem ser mudados dinamicamente durante a manufatura em função das características dinâmicas do sistema de manufatura, trabalhando também de forma *on-line*.

6.3.2.1 Abordagem Para Planejamento do Processo

As duas sistemáticas de divisão das atividades de planejamento do processo (HALEVI X WANG), descritas na revisão de literatura, são bastante semelhantes sendo que um dos aspectos que as difere é a ordem, o momento, onde se executa a atividade associada à seleção da máquina-ferramenta. Na abordagem WANG & LI (1991) esta atividade é a primeira a ser realizada no estágio de projeto de operações de usinagem (macroplanejamento). Isto impõe uma forte restrição à tomada de decisão associada às demais atividades e principalmente na determinação das condições de usinagem para cada operação. A abordagem HALEVI & WEILL (1995) é menos restritiva com relação a escolha da máquina quando comparado a ordem de atividades de WANG & LI (1991), e analisando o exemplo apresentado por HALEVI E WEILL (1995) no capítulo dez do seu livro, nota-se como o método tem consistência com um abordagem bastante lógica e estruturada. As duas abordagens mantêm uma estreita correlação e muitas vezes podem ser utilizadas de forma complementar.

No caso do novo método concebido para planejamento de processo para peças rotacionais simétricas e assimétricas, e na implementação computacional desenvolvida, já se conhece uma das restrições para tomada de decisão, que diz respeito à máquina que será utilizada (Centro de Torneamento Galaxy 15M), bem como à placa e castanhas para fixação da peça. Assim a abordagem WANG & LI (1991) é mais adequada, pois não haverá máquinas alternativas para serem utilizadas.

A partir deste enfoque a abordagem de HALEVI (1999), que aplica o *Matrix Method* como método de otimização baseada em planos de processos alternativos perde o sentido, por se estar trabalhando apenas com uma máquina de comando numérico, tendo como critério de otimização a fabricação com o menor número possível de fixações da peça na máquina e *setup* de ferramentas. Com este critério pode-se penalizar determinada operação alternativa, diminuindo suas chances de ser a operação escolhida dentre as alternativas disponíveis.

O *Matrix Method* poderá ser utilizado no caso de se ter a disponibilidade de várias alternativas de máquinas-ferramenta para serem utilizadas. Neste caso deve-se utilizar como método de otimização a linearização dos diversos planos de processos gerados pelo método de CAPP proposto, definindo-se a melhor máquina para realizar a fabricação da peça, utilizando o método desenvolvido por HALEVI (1999).

O CAPP proposto busca a otimização do sistema de usinagem MFFP (*Máquina, Dispositivo de Fixação, Ferramenta e Peça*) em função da quantidade de fixações necessárias da peça, para possibilitar a execução de todo o plano de processo em uma única máquina. Trabalha-se com planos de

processos alternativos (não-linear) baseado em operações alternativas a serem processadas em uma única máquina. Estes planos de processos alternativos levam em consideração aspectos associados às alternativas de *setups* da peça, superfícies de fixações, operações de usinagem, seqüências de usinagem e seleção de ferramentas de corte.

Com a implementação em um protótipo da metodologia de integração CAD/CAPP/CAM, para geração do plano de processos, baseado na estrutura de dados STEP-NC (aderente à STEP-NC), ter-se-á também uma ferramenta de programação NC orientada ao chão-de-fábrica, gerando o programa NC baseado na ISO 6983 para determinada máquina-ferramenta.

Assim este arcabouço conceitual para integração CAD/CAPP/CAM poderá ser utilizado de forma *off-line* (externa ao CNC) ou incorporado ao próprio CNC aberto da máquina-ferramenta, sendo portanto aderente à proposta STEP-NC (SUH e CHEON, 2002).

6.3.2.2 Metodologia CAPP Para Peças Rotacionais

A figura 6.9 apresenta o diagrama IDEF0 referente ao CAPP. É constituído por dez atividades para tratamento de *features* rotacionais concêntricas (peças simétricas) e não-concêntricas (peças assimétricas, Eixo C), sendo o macroplanejamento constituído pelas atividades de 1 a 3 e o microplanejamento pelas atividades de 4 a 10:

1. mapeamento de *features* de projeto para *features* de manufatura (usinagem);
2. determinação das operações de usinagem com alternativas, associadas às *features* de usinagem: *workingsteps* de usinagem e de torneamento;
3. determinação da seqüência de usinagem com alternativas: *workplan* não-linear;
4. estratégias para geração de trajetórias de ferramentas;
5. determinação das ferramentas de corte (insertos e suportes): ferramentas para torneamento externo e interno (torneamento geral, cortar, ranhurar e rosqueamento) e ferramentas rotativas (furar, fresar, alargar, rosqueamento com macho rígido e escarear);
6. determinação do modelo de tempos e cálculo dos tempos padrão e custo padrão para cada *workingstep*;
7. determinação das condições tecnológicas de usinagem otimizadas, utilizando algoritmos genéticos;
8. linearização do plano de processos não-linear, utilizando algoritmos genéticos;
9. geração do programa NC (ISO 6983);
10. geração de relatórios e plano de processo.

A descrição das principais funcionalidades associadas às atividades são apresentadas a seguir.

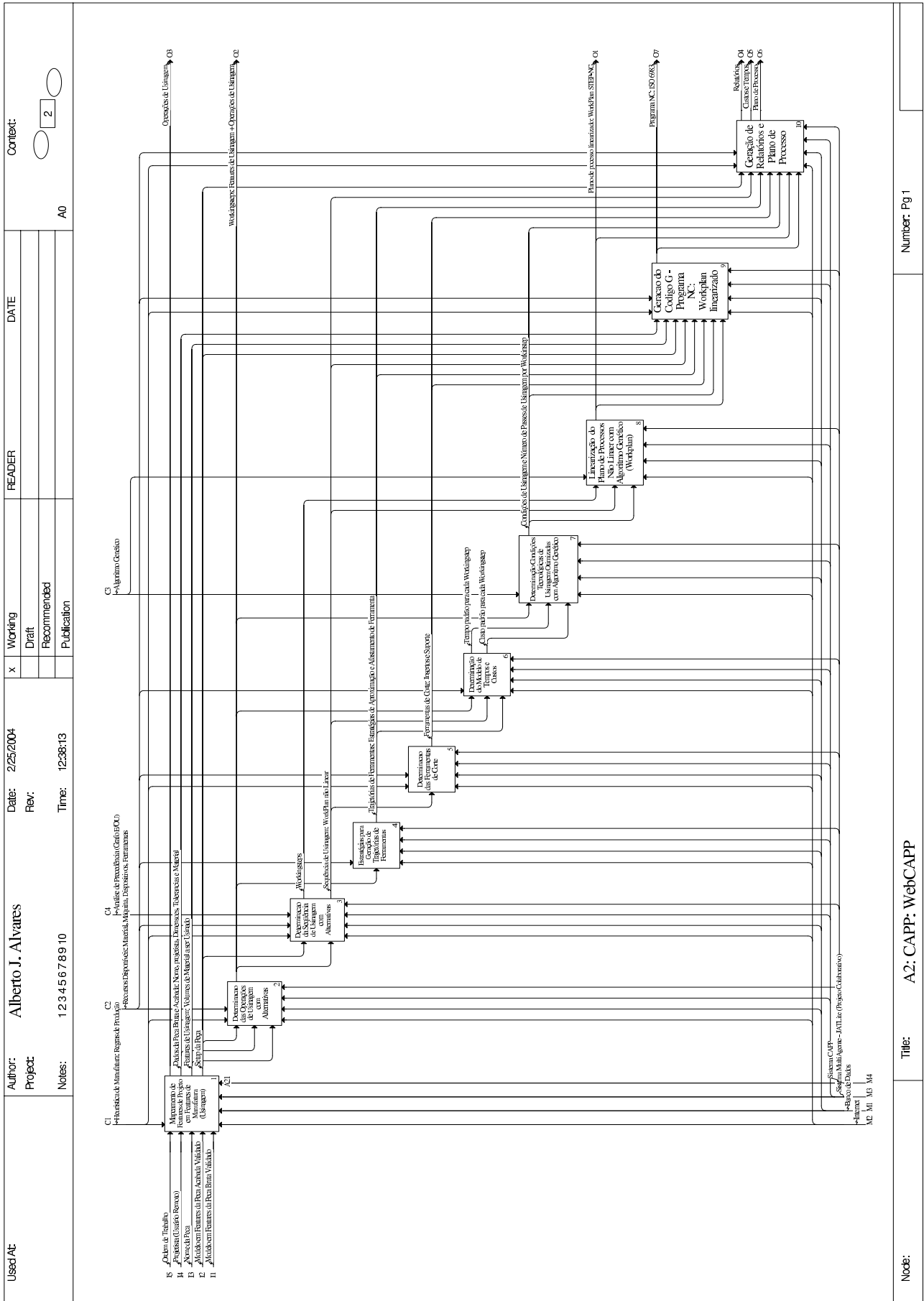


Figura 6.9: Diagrama IDEF0: Atividades CAPP.

6.3.2.3 Mapeamento de *Features* de Projeto para *Features* de Manufatura

A principal atividade do WebCAPP é o “mapeamento de *features* de projeto para *features* de manufatura” onde são determinadas as *features* de usinagem, fazendo sua associação com as alternativas de operações de usinagem para gerar a *feature* de projeto. Esta atividade está diretamente associada com o modelo de *features* utilizado. O WebCADbyFeatures fornece os dados para o sistema de planejamento de processo.

Para facilitar a entrada de dados para o projetista a GUI é orientada por menus para a instanciação de uma *feature* e a definição de seus atributos, relacionamentos, tolerâncias e especificações. Por exemplo uma *feature* “OD retilínea” (diâmetro externo, cilindro), com chanfros à esquerda e à direita, pode ser modelada como sendo uma *feature* “OD retilínea” com mais duas *features* de transição do tipo chanfro; ou se o projetista preferir, poderá ser instanciada como uma *feature* “OD cônica esquerda” + “OD retilínea” + “OD cônica direita”.

Assim o modelo de *features* gerado pela GUI e disponibilizado para o CAPP será constituído por uma *feature* “OD cônica esquerda” + “OD retilínea” + “OD cônica direita”, indiferentemente do método de modelagem utilizado pelo usuário disponível na GUI. Esta atividade é denominada de normalização da biblioteca de *features*, sendo realizada pelo CAPP.

O ambiente de projeto por *features* é baseado em *features* de forma de alto nível como chanfro, segmento cônico, furo de centro, rasgo de chaveta, etc, que irão corresponder de uma forma geral às operações de usinagem, com pouco detalhamento. O mapeamento de *features* de alto nível para *features* de baixo nível, que corresponde mais diretamente às operações de usinagem, relativas aos volumes de usinagem que serão removidos, será implementado por meio de um novo procedimento de mapeamento denominado decomposição orientada ao *setup*, à geometria e à operação de usinagem, levando em conta este três aspectos simultaneamente. A figura 6.10 apresenta o diagrama IDEF0 para esta atividade.

Por exemplo, um eixo pode ser tratado como uma seqüência de segmentos, pois o processo de torneamento pode ser adequadamente modelado por um sistema gráfico 2D, sem nenhuma ambigüidade na sua representação, diferentemente de peças prismáticas.

Uma seqüência de segmentos é uma *feature* composta que consiste de uma coleção de segmentos cônicos e cilíndricos. Os segmentos tem a mesma linha de centro. Assim as *features* de forma associadas à definição de instanciação da peça, podem ser mapeadas para *features* de seqüência de segmentos e as operações de torneamento, com seus diversos passes, podem ser determinadas para operações de desbaste, semi-acabamento e acabamento, para as *features* de usinagem, no caso de torneamento. As *features* de transição podem também ser mapeadas, determinando-se os chanfros associados as *features* de seqüência de segmentos.

Desta forma pode-se determinar as *features* de usinagem e torneamento que irão representar as operações de usinagem associadas à remoção de material em relação à peça bruta previamente definida. Uma vez que estes volumes a serem usinados foram identificados, é necessário determinar as operações de usinagem que podem produzi-los. Para capturar as características fundamentais do processo de usinagem, é necessário entender a interação peça-ferramenta, ou seja sua cinemática, para gerar os volumes a serem removidos. É necessário criar este modelo e associar restrições (geométrica, dimensional, tecnológica e econômica) e heurísticas de manufatura ao modelo do processo. Utiliza-

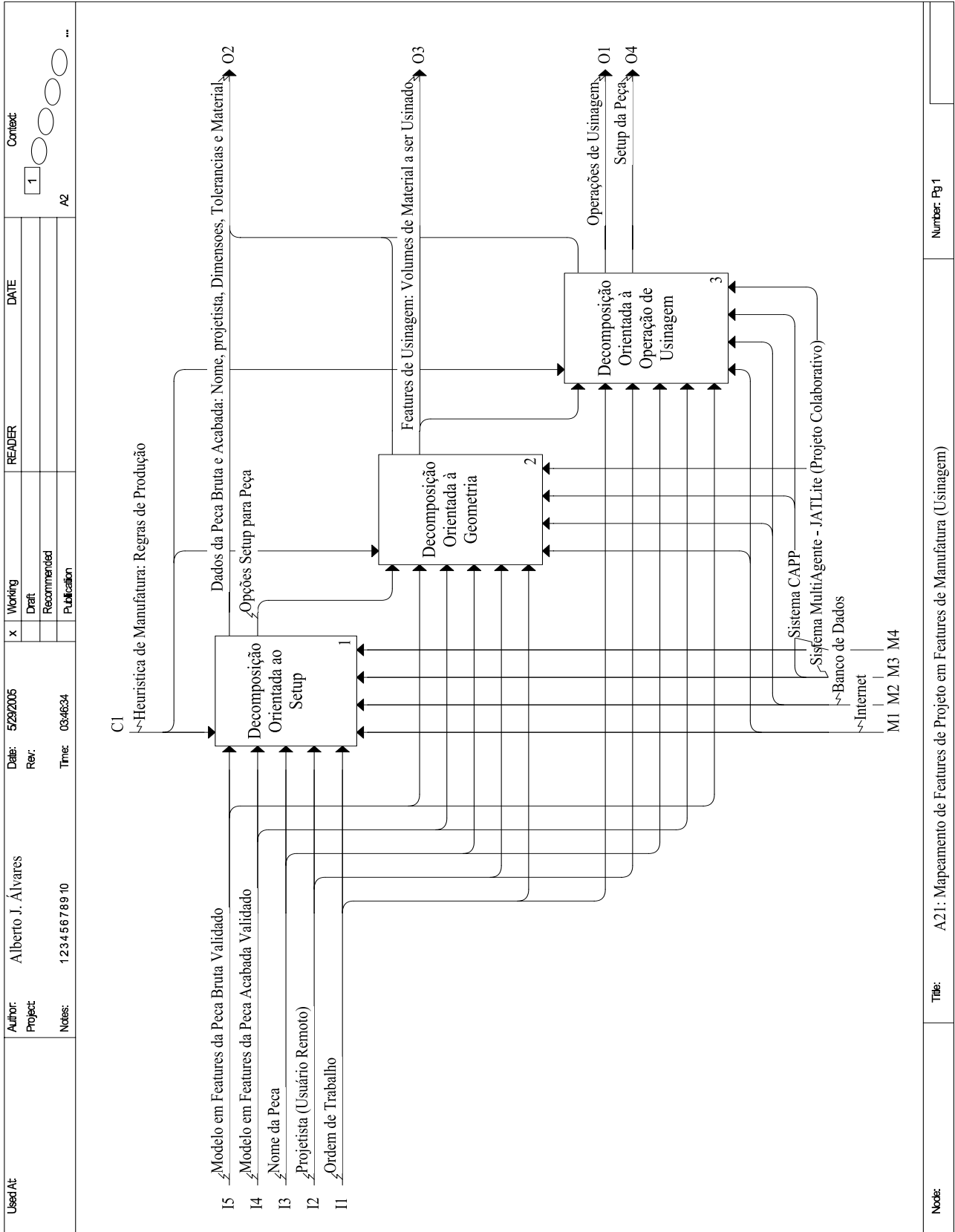


Figura 6.10: Diagrama IDEF0: sub-atividades do WebCAPP associadas ao mapeamento de *features* de projeto para *features* de manufatura.

se como referência as *features* de manufatura para torneamento definidas na ISO/DIS 14649 - Part 12 (2003) e as *features* de furação e fresamento definidas na ISO/DIS 14649 - Part 11 (2003).

A atividade de planejamento da operação de usinagem pode ser definida como a tarefa de especificar a operação mais apropriada de uma lista de operações disponíveis para criação de uma *feature*. O planejamento da operação elucida a relação entre os modelos de processo e de *features*. A representação do plano de processo linearizado é baseada na estrutura de modelo de dados da ISO 14649, denominada de STEP-NC, e o programa NC é gerado de acordo com a norma ISO 6983, códigos G/M.

Operações de torneamento podem ser modeladas em termos de tipos de arestas, na forma de linhas do contorno da peça, ou seja, de um desenho que representa o perfil 2D da peça modelada por *features* de forma, segundo o seu contorno ou através de superfícies. A remoção de material necessária é obtida em função da forma da peça bruta especificada.

As operações de torneamento estão diretamente associadas à fixação da peça, exigindo, normalmente, pelo menos duas fixações da peça. Estratégias de movimentação de ferramenta e de fixação da peça associadas ao “torneamento para a direita” e “torneamento para a esquerda” devem também ser consideradas (ÁLVARES, 2002) e (SHAH e MANTYLA, 1995).

Uma abordagem complementar no macroplanejamento ((FERREIRA, 1996) e WANG E LI (1991)), é a utilização de tabelas que já apresentam roteamentos típicos de usinagem para superfícies cilíndricas externas, internas e planas. Exemplos destes tipos de tabelas podem ser obtidos em manuais. Após a seleção dos processos finais de usinagem e roteamentos de todas as superfícies (críticas e menos críticas), faz-se uma análise considerando-se as *features* inter-relacionadas por tolerâncias. Um exemplo de roteamento de usinagem para uma determinada superfície poderia ser: torneamento (desbaste) -> torneamento (semi-acabamento) -> retificação (desbaste) -> retificação (acabamento) -> super-acabamento (Tolerância IT de 3-5 e Rugosidade de 0,008-0,16 μm).

Operações de torneamento exigem a capacidade de se tratar com o conceito de geometrias intermediárias, que são geradas durante as várias operações. Por exemplo, em função da geometria da ferramenta determinados segmentos não poderão ser executados durante uma operação de usinagem associado a uma fixação/*setup*, pois haverá interferência entre a ponta da ferramenta e a peça durante a movimentação da ferramenta, inviabilizando a operação. Devido a este fato uma nova geometria associada à cinemática peça-ferramenta será necessária, sendo denominada geometria intermediária. É necessário a usinagem desta geometria a partir de um novo *setup* da peça ou a troca de ferramenta usando-se uma ferramenta mão esquerda (avanço para direita), por exemplo.

6.3.2.4 Operações de Usinagem

As operações consideradas são torneamento cilíndrico externo e interno (mandrilamento), faceamento, furação, sangramento (interno e externo) e rosqueamento (interno e externo) em um centro de torneamento. Também são permitidas operações de fresamento e furação, radiais e axiais, através de ferramenta rotativa (Eixo C) do centro de torneamento.

A metodologia de planejamento a ser utilizada é uma combinação de dois métodos: uma árvore de decisão e um método heurístico usando regras de produção. Uma estrutura em árvore é construída através do seguinte procedimento: mapeamento de *features* (*feature* de manufatura); geração do vo-

lume a ser usinado e determinação da operação; modificação da geometria da peça.

As *features* de forma normalmente presentes em peças torneadas são: cilindro, face, rosca, cônico, chanfro, arco, rasgo e entalhe. As *features* de usinagem (concêntricas e não-concêntricas) consideradas podem ser classificadas de forma geral em cinco categorias: rosca, chanfro e rasgo, perfil externo, perfil interno e Eixo C.

A determinação do volume de material a ser removido e a operação baseia-se no tipo de *feature* reconhecido pelo sistema, conforme descrito no item mapeamento de *features* (6.3.2.3). A cada *feature* reconhecida, a área ou perfil de usinagem e a operação podem ser determinadas. As operações de acabamento para *features* concêntricas consistem de: rosqueamento, torneamento, faceamento, mandrilamento e alargamento. A operação final de acabamento constitui-se de um único passe. A profundidade de corte ainda não é conhecida e só será calculada depois que uma apropriada ferramenta de corte tenha sido selecionada.

6.3.2.5 Seqüência de Operações de Usinagem

A determinação da seqüência de operações de usinagem é baseada na análise de restrições sendo implementado através de regras de produção (CHANG *et al.*, 1998) e (ÁLVARES, 2001). As restrições que determinarão um gráfico ou tabela de precedência são de quatro tipos: restrições de operações; restrições geométricas; restrições de ferramentas; restrições de tolerâncias geométricas.

Estas restrições são analisadas de forma conjunta, no procedimento de mapeamento de *features* denominado de decomposição baseada no *setup*, geometria e operações de usinagem. Esta decomposição irá definir quantos *setups* deverão ser feitos na peça, a lista de *features* de usinagem a ser usinada em cada *setup* (levando em conta restrições de referências, tolerâncias dimensionais, tolerâncias de microgeometria (rugosidade) e de macrogeometrica (forma, localização, orientação, batimento e posição) e as operações de usinagem que serão executadas em cada *feature* de usinagem ou de torneamento.

É adotada a nomenclatura STEP NC (ISO 14649, 2003), definindo assim as *features* de usinagem e de torneamento (Fu) e os *Workingsteps* associados. Para cada alternativa de *Workingstep* relativo à uma *feature* de usinagem/torneamento (volume de material a ser removido da peça), ter-se-á as operações de usinagem e ferramental que podem ser utilizadas para remoção desta *feature* de usinagem, gerando assim um plano de processos com alternativas de operações e ferramentas que poderão ser utilizadas. Assim cada Fu terá um relacionamento de 1-> n operações, gerando “n” *Workingsteps* diferentes.

O roteamento das operações associadas ao torneamento, para perfil externo e interno, é baseado em rotas pré-estabelecidas (HUANG, 1988) associado às regras que definem boas práticas de usinagem, tendo a seguinte seqüência básica (figura 6.11): furação -> facear (desbaste e acabamento) -> torneamento longitudinal (desbaste e acabamento) -> sangramento (desbaste e acabamento) -> recartilhamento -> rosqueamento -> cortar.

Para as operações associadas a Eixo C (ferramenta rotativa, furação e fresamento (radial e axial)), tem-se o seguinte roteamento pré-determinado, para cada *setup* da peça e direção de usinagem: fresamento -> furação -> escareamento -> alargamento -> rosqueamento.

Para determinação do *Workplan* com alternativas, que determina a seqüências de execução dos

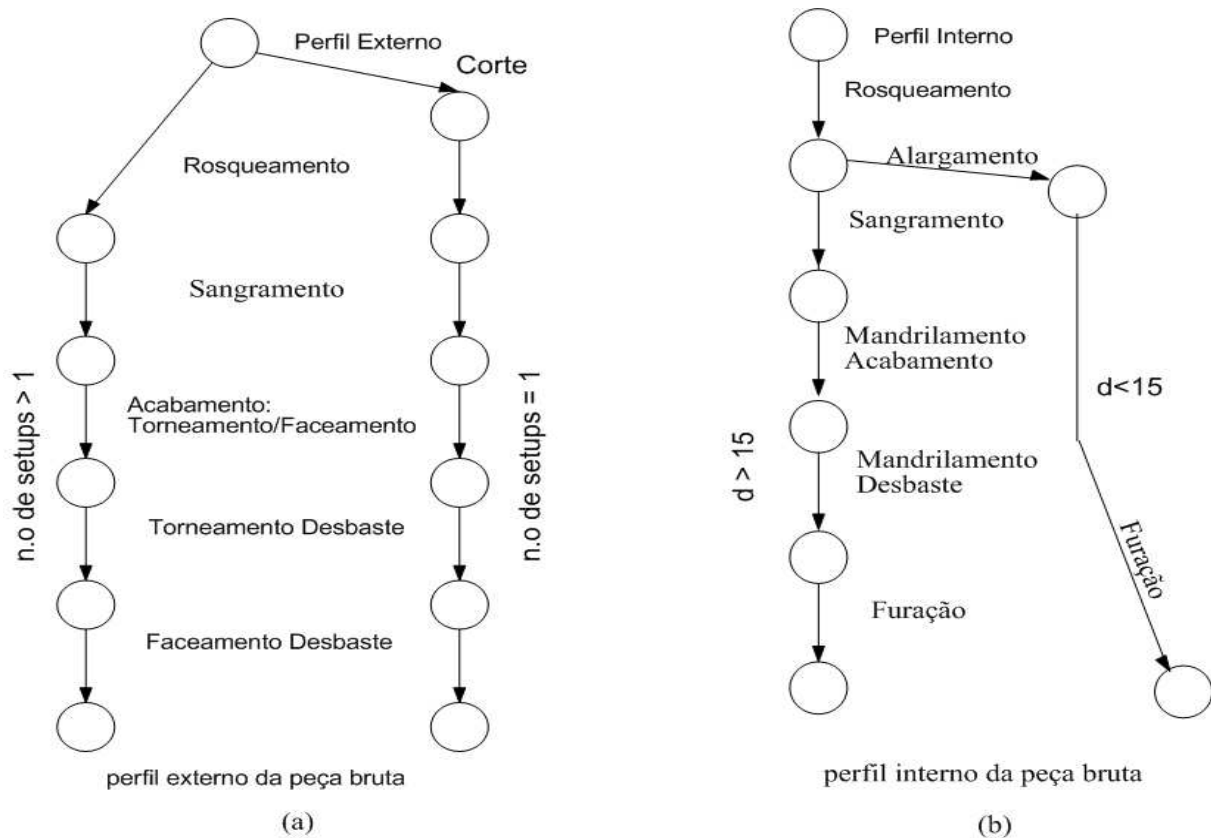


Figura 6.11: Árvores de decisão para *features* concêntricas baseada em HUANG (1988).

Workingsteps para cada *setup*, é utilizado um método baseado em SUNDARAM (1986), que permite a representação e definição do Grafo de Precedência de *Features* de Usinagem. Esta técnica estabelece um procedimento sistemático para seleção da seqüência de operações de usinagem (*Workplans*). As operações elementares de usinagem podem ser agrupadas em uma seqüência e o grupo de operações elementares pode ser processado em uma única máquina. A partir da especificação de operações de usinagem é montado uma matriz de relacionamento de precedência de operações baseada nas restrições mencionadas, que pode ser visualizado pelo Grafo de Precedência de *Features* de Usinagem.

Associado ao Grafo de Precedência de *Features* de Usinagem tem-se um grafo E/OU de *Workingsteps*, que representa para cada *feature* de usinagem/torneamento as alternativas de operações de usinagem.

6.3.2.6 Determinação das Condições de Usinagem

As condições de usinagem são obtidas através da consulta à uma base de dados que disponibiliza as informações de condições de usinagem e ferramentas baseado em dados de fabricantes de ferramentas e a utilização de algoritmo genético no procedimento de otimização. Faz-se a seleção das condições de corte para uma dada combinação de materiais de peça e ferramenta, bem como, da geometria da ferramenta WANG E LI (1991) e HUANG (1980).

As restrições associadas à velocidade de corte, avanço, profundidade de corte, força de corte e potência de corte são levadas em consideração, utilizando-se modelos de processos de usinagem para calculá-los.

6.3.2.7 Geração do Programa NC

O método desenvolvido para geração do código NC, utiliza o conceito de *Workingstep*, que tem associado uma *feature* de usinagem/torneamento e uma operação de usinagem. Assim já se conhece para as *features* de usinagem/torneamento a sua geometria associada ao perfil 2D da peça rotacional (*features* concêntricas). Para *features* não-concêntricas (Eixo C), têm-se a informação sobre o volume de material a ser removido associado à uma operação de fresamento ou furação, de forma semelhante ao que ocorre no método DSG.

Cada operação de usinagem têm associado uma ferramenta de corte, estratégia de movimentação de ferramenta, condições de usinagem, movimentação de ferramenta, plano de aproximação e um plano de segurança.

Para geração da estratégia de movimentação de ferramenta utiliza-se três abordagens:

- abordagem manual;
- abordagem automática, baseada em ciclo fixo de usinagem disponibilizada pelo CNC;
- híbrida.

Na abordagem manual, toda a movimentação de ferramenta é feita manualmente, sendo necessário o cálculo de todo o posicionamento intermediário da movimentação de ferramenta, o que apresenta um alto custo computacional para o CAPP, sendo propenso ao erro.

Na abordagem automática, baseada em ciclos fixos, define-se apenas os parâmetros necessários ao ciclo fixo, e os dados relacionados as cotas (X, Z) que definem o perfil da peça acabada, segmentos.

Na abordagem híbrida utiliza-se as duas estratégias de movimentação de ferramenta; como por exemplo em operações de sangramento de canais largos, que pode ser necessário a utilização da movimentação manual associado a um ciclo fixo.

6.3.3 WebTurning: CAM Execução

A figura 6.12 apresenta o diagrama IDEF0 referente ao CAM execução, sendo cons-tituído por três atividades:

1. GUI para teleoperação (WebDNC);
2. WebCam para captura e *streamings* de vídeo e áudio;
3. WebCNC para teleoperação da máquina CNC.

O sistema de teleoperação do torno, denominado WebTurning, é baseado em uma arquitetura Cliente-Servidor, utilizando metodologia desenvolvida por ÁLVARES e ROMARIZ (2002), constituída por dois módulos (ÁLVARES, *et al.*, 2002):

- servidores: representado pelos programas localizados na estação de trabalho (computador pessoal com Sistema Operacional Linux), conectado à máquina-ferramenta através da interface ethernet e protocolos TCP/IP e DNC;
- clientes: representado por *applet* Java e páginas HTML.

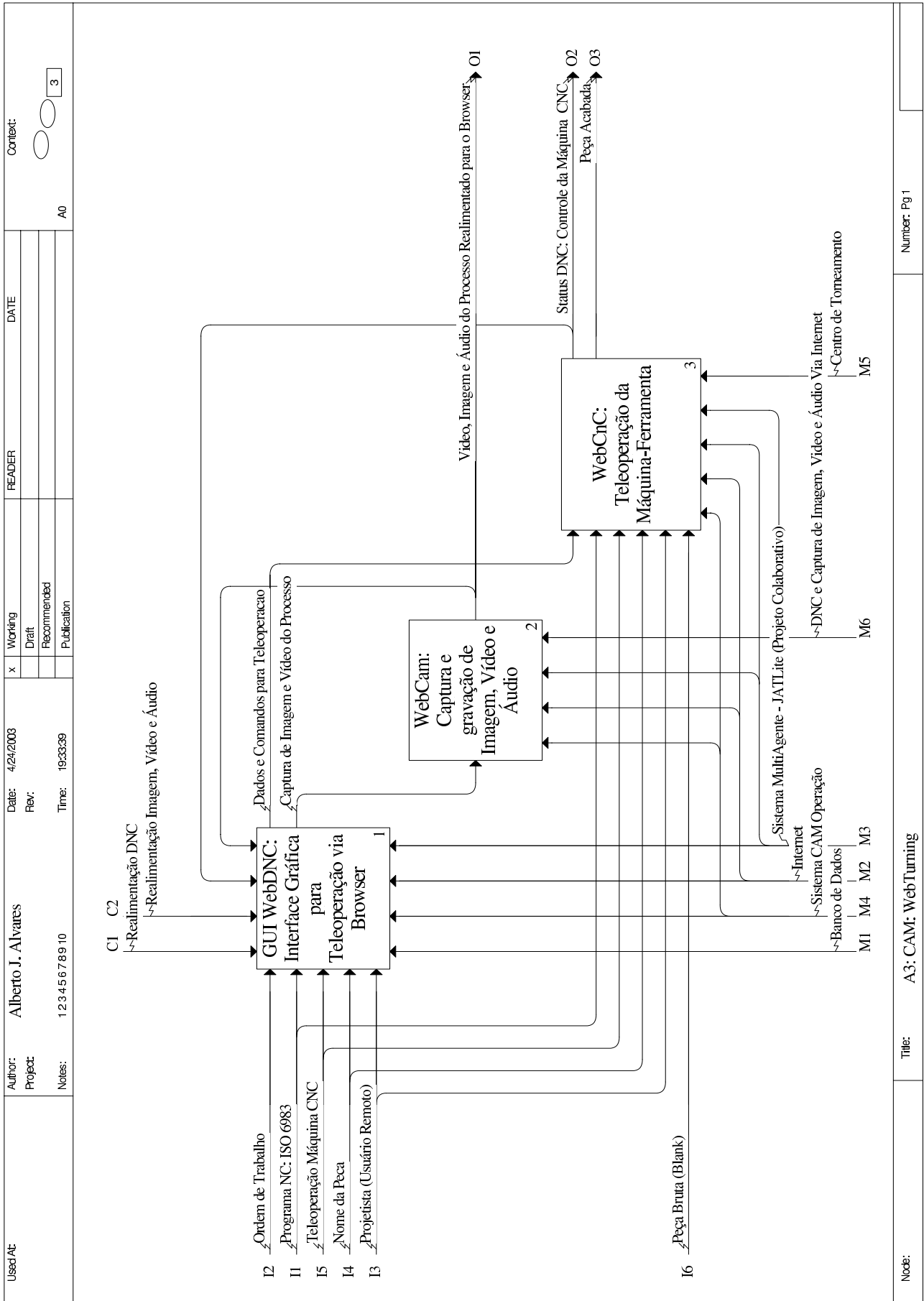


Figura 6.12: Diagrama IDEF0: Atividades CAM.

6.4 Modelagem IDEF1X

O modelo de informação que possibilita a integração CAD/CAPP/CAM é baseado na abordagem Projeto por *Features* (síntese por *features* de projeto), sendo concebido através da abordagem metodológica IDEF1X (banco de dados relacional), tendo todos os dados referentes ao modelo de informação contidos em um único diagrama IDEF1X (ÁLVARES, 2003). O modelo de informação descreve cada subsistema em termos de entidades/objetos, atributos e relacionamentos. A partir deste modelo IDEF1X é gerado o modelo físico da base de dados relacional. É utilizado o sistema de gerenciamento de base de dados MySQL. O diagrama IDEF1X é apresentado na <http://WebMachining.AlvaresTech.com/idef1x>.

O modelo de informação é dividido em domínios (CHEP *et al.*, 1998) associados às bases de dados de *features* (*features* de forma, *features* de tolerância, *features* de produção e *features* de materiais) e base de dados de tecnologia de usinagem (biblioteca de máquinas-ferramenta, biblioteca de ferramentas de corte, biblioteca de usinabilidade e biblioteca de fixações). A base de dados de *features* relaciona-se com o modelo do produto e a base de dados de tecnologia de usinagem relaciona-se com o modelo de recursos. Estas bases de dados foram implementadas fisicamente através de uma única base de dados denominada de WebMachining, sendo constituída por oitenta e duas tabelas.

O usuário remoto conecta-se ao sistema criando um novo projeto, que será constituído pelos dados referentes à modelagem por *features* de uma peça acabada e de uma peça bruta, além de informações tecnológicas, tolerâncias dimensionais e geométricas, referências e cotas de projeto, relacionamento de *features*, acabamento superficial, dados de produção e dados do usuário. Todo projeto é constituído por este conjunto de dados para peça bruta e acabada, que estão definidos nos vários domínios descritos. Um usuário pode ter vários projetos concebidos e armazenados no sistema de base de dados. Cada usuário tem uma identificação única no sistema.

Todos os relacionamentos entre as entidades são vistos de forma hierárquica a partir do domínio da peça (peça bruta ou peça acabada), seguido pela classe de *features* (concêntrica ou não-concêntrica, Eixo C), pelo tipo de *features* de forma (furo, diâmetro externo, diâmetro interno, reentrância, canal, rosca, face, rasgo, entre outros) e finalmente pela *feature* em si e seus atributos. Mecanismos de herança de *features* são aplicados nesta taxonomia de *features*, quando se aplica conceitos de orientação a objetos nos relacionamentos das *features* (relacionamento do tipo é um). A figura 6.13 mostra a modelagem da informação no contexto do projeto WebMachining.

Pode-se ter a modelagem de peças rotacionais associada à *features* concêntricas, que ocorrerá nas regiões externa e interna da peça; bem como associada às *features* não-concêntricas (Eixo C), que poderá utilizar-se de ferramenta rotativa axial ou radial.

Para cada tipo de *feature* podem ser associadas as tolerâncias dimensional, forma, localização, orientação, batimento, posição e qualidade superficial, que são tratadas como entidades relacionadas às *features* de forma, segundo a taxonomia de *features* utilizada (CAM-I, 1986). Por exemplo, uma *feature* do tipo furo pode ter uma tolerância dimensional e uma tolerância de circularidade associada. Uma *feature* pode ter também uma referência (*Datum*), sendo este *Datum* uma referência para outra *feature*, associado à uma tolerância de posição, localização, orientação ou de batimento. A figura 6.14 apresenta as *features* de tolerâncias que podem estar associadas às *features* de forma baseadas em CAM-I (1986). A modelagem em UML (diagrama de classes) associada aos recursos de manufatura

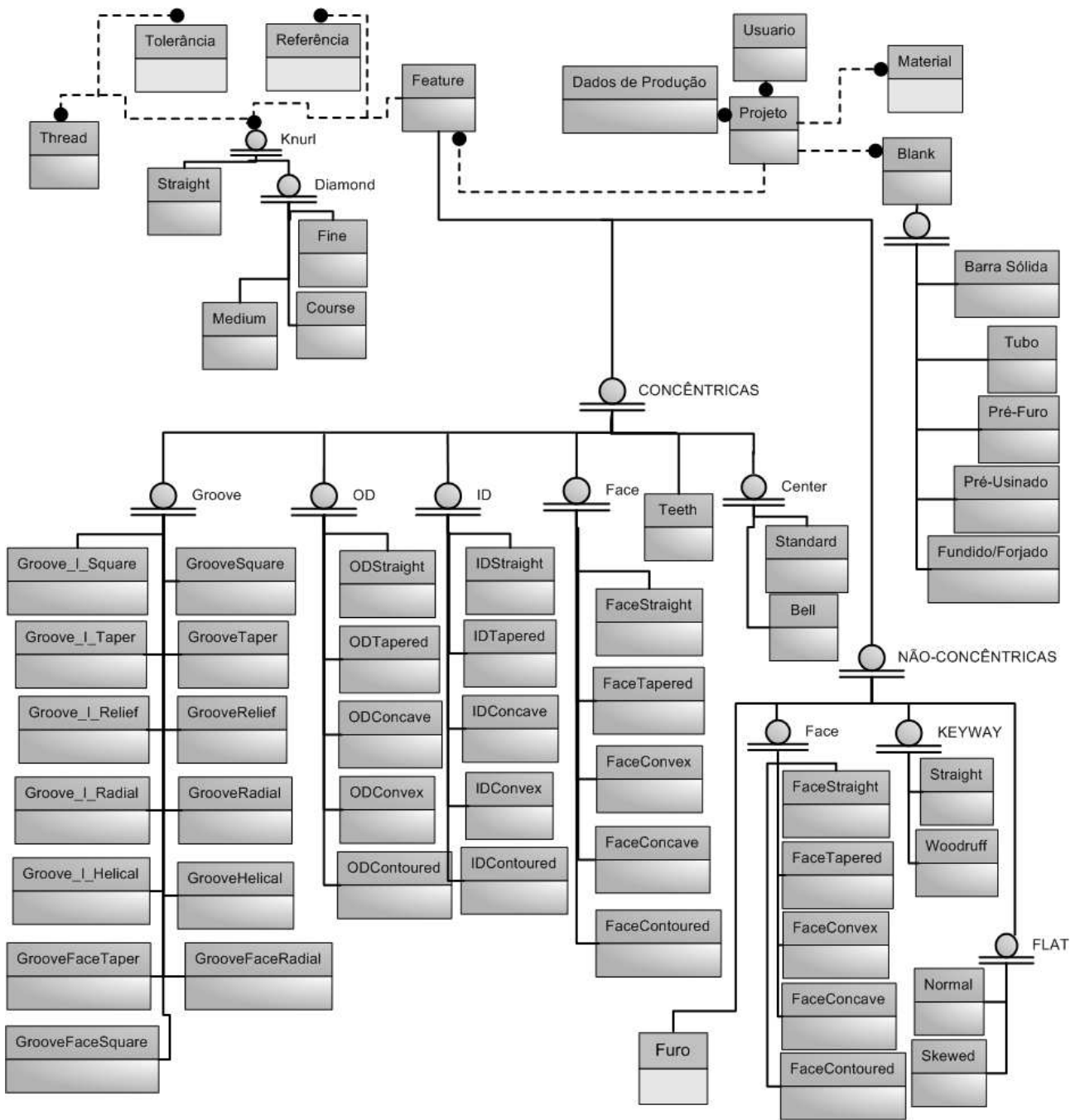


Figura 6.13: Modelagem da informação de algumas *features* de forma utilizadas através da metodologia IDEF1X.

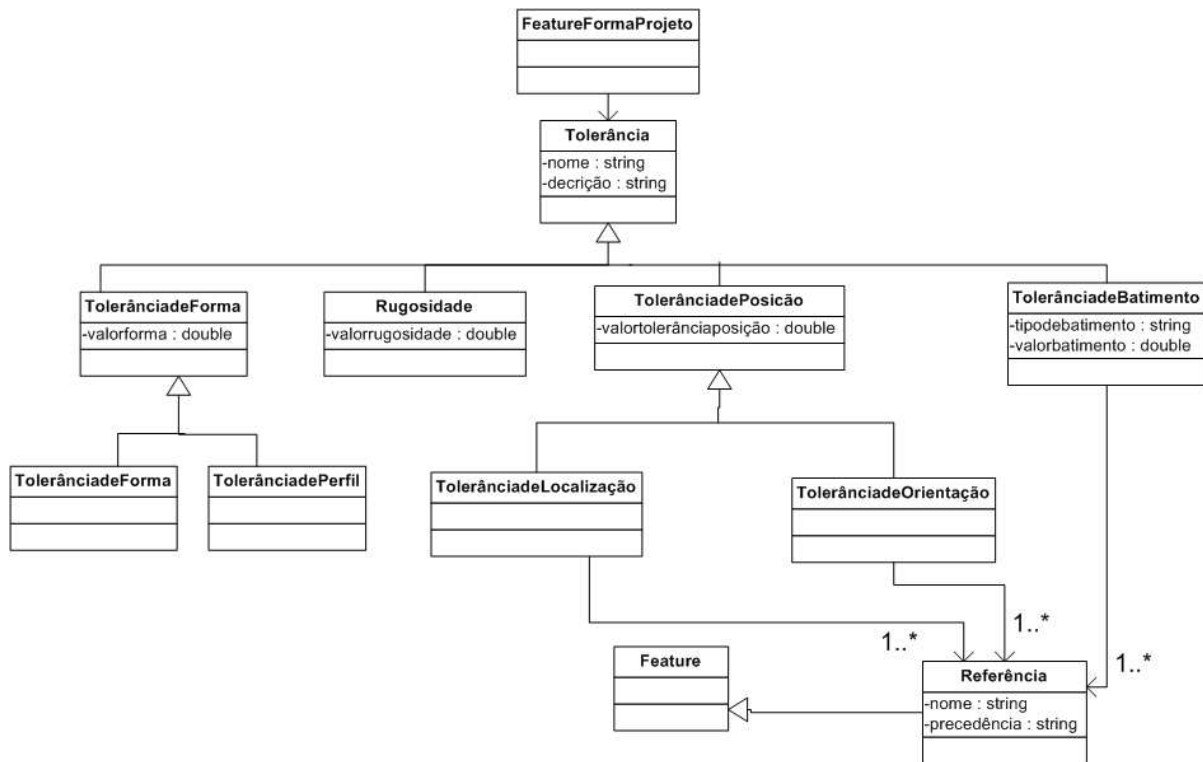


Figura 6.14: Diagrama de classes para *features* de tolerâncias.

é apresentado na figura 6.15

Roscas (*Threads*) são tratadas como atributos de *features* de Diâmetro Externo (OD) e Diâmetro Interno (ID), podendo ser identificadas também como uma *feature* de Eixo C (não-concêntrica), associada a um furo concêntrico ou não-concêntrico. Recartilhado (*Knurl*) são tratados como atributos de *features* de Diâmetro Externo (OD). Ou seja, a partir de uma *feature* básica/principal (OD ou ID) é associada uma *feature* secundária especificada na forma de um atributo, que poderá estar presente ou não na *feature* principal (relacionamento do tipo tem um). Caso esteja presente, é estabelecido um relacionamento entre as duas entidades, ou seja uma *feature* OD (principal) tem uma rosca (secundária). A utilização deste artifício é necessário pois não é possível separar uma *thread* cilíndrica de um cilindro interno ou externo, sendo mais adequado trabalhar com o conceito de atributo, conforme sugerido por CAM-I (1986).

As *features* de transição (filetes, arredondamentos e chanfros) são representadas como atributos de *features* de Diâmetro Externo (OD) e Diâmetro Interno (ID). Podem também ser tratadas como uma *feature* de forma do tipo Diâmetro Externo (OD) ou Diâmetro Interno (ID). Neste caso não se tem a presença de atributos de filetes e chanfros na instânciação da *feature*. Esta definição fica a critério do projetista, possibilitando mais flexibilidade na modelagem da peça.

Os capítulos seguintes (7 a 13) apresentam um detalhamento da metodologia WebMachining, ilustrando os métodos e algoritmos concebidos, resultados da implementação e estudos de casos. O Apêndice C apresenta a modelagem matemática para aplicação de GA e o Apêndice D apresenta a modelagem UML associada ao sistema computacional desenvolvido e disponibilizado na URL <http://WebMachining.AlvaresTech.com>.

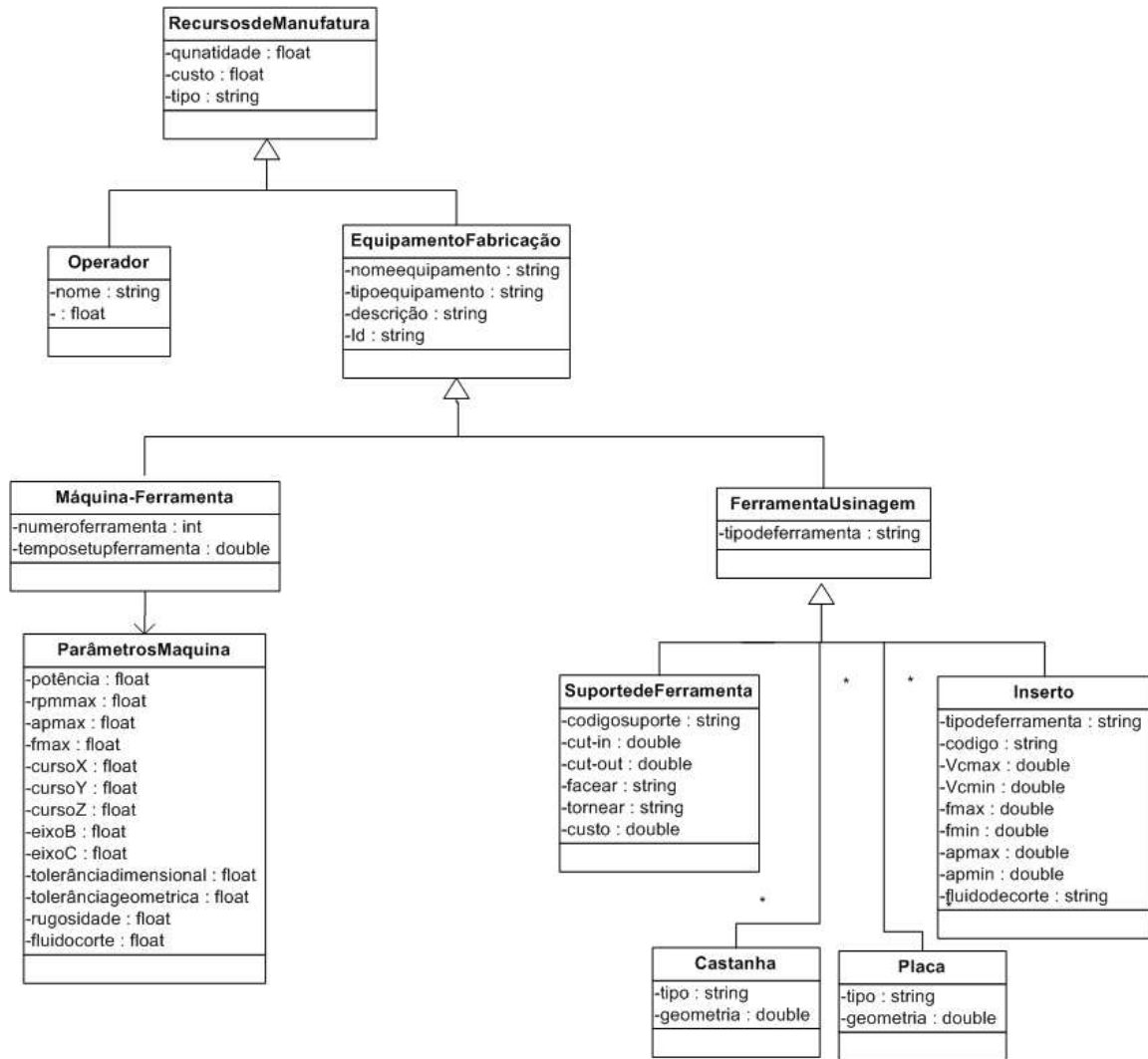


Figura 6.15: Diagrama de classes para tecnologia de usinagem (recursos de manufatura).

Capítulo 7

Interface CAD-CAPP: CAPP Baseado em Mapeamento de *Features*

Este capítulo descreve a interface CAD-CAPP concebida para peças rotacionais (concêntricas e não-concêntricas), consistindo de um conjunto de especificações, técnicas, métodos e algoritmos utilizados na definição funcional dos módulos da metodologia para interfacear as atividades de projeto detalhado colaborativo (*WebCADbyFeatures*) e o planejamento de processo generativo não-linear (*WebCAPP*), voltado para geração de planos de processo linearizados e de programas NC para um centro de torneamento.

Após o projeto colaborativo, a peça modelada é armazenada e disponibilizada para a metodologia CAPP gerar o plano de processo com alternativas, sua linearização e a representação do plano de processo linearizado baseado em STEP-NC (ISO 14649 - Part 12 e Part 11), bem como a geração do programa NC para o centro de torneamento selecionado.

O planejamento de processo tem início na atividade de mapeamento de *features*, ou seja na transformação ou mapeamento de *features* de projeto (concêntricas e não-concêntricas) em *features* de usinagem. A descrição dos métodos e técnicas utilizadas em cada uma das dez atividades são apresentadas neste capítulo.

7.1 Mapeamento de *Features*

O método de mapeamento de *features* desenvolvido depende da natureza da *feature* de forma que está sendo analisada, isoladamente e também de maneira conjunta com as demais *features* de mesmo tipo, sendo oriundo das classes de mapeamento descritas a seguir:

1. classe de *mapeamento um-para-um* (1 =>1): *features* do tipo recartilhado e rosca mapeadas em operações de recartilhamento e rosqueamento, respectivamente, gerando um `Machining_workingstep` (STEP NC - Part 10 e Part 12) para operações associadas a torneamento. Já para *features* de Eixo C (furos, rasgo de chaveta e cavidades) é necessário o uso de operações com ferramentas rotativas em um centro de torneamento, sendo estas *features* mapeadas em operações de furação, fresamento, escareamento e rosqueamento. STEP-NC Part 12 não prevê *features* de eixo C, devendo-se utilizar a norma STEP-NC Part 11;

2. classe de *mapeamento agregação discreta* ($m \Rightarrow 1$): onde “m” *features* de projeto são combinadas em uma *feature* de usinagem simples, gerando o *Turning_workingstep* (STEP NC - Part 10 e Part 12) ou um *Machining_workingstep* do tipo *groove* ou *spline*, denominados de *groove* complexo e *spline virtual*, respectivamente. A biblioteca de *features* é normalizada durante o procedimento de mapeamento de *features*, para interpretar estes tipos de *features* de projeto em *features* de usinagem, e a partir destas associar um determinado tipo de *workingstep*. Ocorre nas operações de torneamento cilíndrico, perfilamento e sangramento;
3. classe de *mapeamento especializado* ($1 \Rightarrow n$): ocorre em canais (*grooves*), *splines* e superfícies cônicas, operações de sangramento e torneamento cônico;
4. classe de *decomposição discreta* ($1 \Rightarrow m$): uma *feature* é decomposta em “m” *features*, sendo representada por uma *Composite_feature* de usinagem.

As técnicas de mapeamento a serem utilizadas são derivadas de:

1. *métodos heurísticos*: onde as *features* podem ser combinadas ou decompostas em função de um conjunto de regras de transformação de uma visão à outra, por exemplo, mapeamento $1 \Rightarrow 1$, aplicado em recartilhados e roscas;
2. técnica de *mapeamento baseado em células*: onde volumes são combinados dentro de diferentes grupos extraíndo-se as *features* de usinagem das *features* de projeto. A decomposição de célula volumétrica segue quatro fases:
 - determinar o volume a ser removido por usinagem (peça acabada menos peça bruta);
 - particionar cada volume de remoção de material em sub-volumes elementares;
 - combinar os volumes elementares em *features* de usinagem;
 - combinar as *features* de usinagem para operações de usinagem com base na classificação anterior.

As *features* de forma de usinagem são classificadas em grupos de formas similares e cada grupo é associado com processos que podem produzir a peça, determinando assim um *Machining_workingstep* ou um *Turning_workingstep*. As *features* de usinagem mapeadas são utilizadas para representação simbólica das formas. Devido à simetria rotacional o planejamento de operações de torneamento recomenda uma representação baseada em arestas/linhas ou superfícies.

A atividade de mapeamento de *features* de usinagem depende da geometria da peça bruta e da peça acabada, dos métodos de usinagem (operações e *setup*) e da seqüência de operações de manufatura. Em função destas opções pode-se ter mapeadas diferentes *features* de usinagem. Algumas *features* de projeto podem ser convertidas diretamente e outras não, devido às discrepâncias semântica e geométrica do modelo de *features* em uma determinada visão (projeto) para outra (usinagem).

A metodologia concebida propõe um sistema de modelagem por *features* com múltiplas visões, para permitir a integração das diferentes fases do ciclo de vida do produto (fases de projeto detalhado, planejamento de processo e fabricação), usando relações geométricas (geometria do produto) e

não-geométricas (tolerâncias dimensionais, forma, posição, orientação, batimento e rugosidade) para conectar as diferentes fases do ciclo de vida do produto, tendo uma grande influência no planejamento do processo e fabricação do produto.

A nova técnica proposta de mapeamento é derivada da técnica de *mapeamento baseada em células*, apresentando uma abordagem lógica para decomposição volumétrica orientada por três aspectos, sendo aplicada para *features* rotacionais concêntricas e não-concêntricas (Eixo C):

1. *Setup*¹: determina-se inicialmente a quantidade de fixações necessárias (1, 2 ou 3) para fabricar a peça, e a seguir as alternativas de fixação da peça para cada um dos *setups*, determinando-se as superfícies de fixação da peça e grupo de *features* a serem usinadas em cada *setup*. Em função do *setup*, as *features* de usinagem podem ser diferentes. Logo os volumes de usinagem dependem do *setup*.
2. Geométrico: geometria da peça acabada e bruta, gerando um esquema de representação do tipo *Poliforma Fechada*, composta por elementos geométricos 2D dos tipos linhas, arcos e *splines*, onde são conhecidos todos os vértices da poliforma, representando a fronteira de remoção de material. Esta poliforma, modelo geométrico, é gerada a partir do modelo de *features*.
3. Operações de usinagem: em função do tipo de *feature* de projeto pode-se determinar a classe de mapeamento, simplificando o problema. Por exemplo uma *feature* rosca ou um recartilhado podem ser mapeadas em um modelo de classe 1 => 1 usando o método heurístico. Já um canal, em função da sua geometria, pode ser mapeado em vários tipos de combinações de operações de usinagem podendo ser classificado como um mapeamento com alternativas, classe de *mapeamento especializado* (1 => n).

O método idealizado leva em consideração, de forma simultânea, os três aspectos que mais influenciam a determinação dos volumes a serem usinados em peças rotacionais. As *features* de usinagem são mapeadas a partir das *features* de projeto disponíveis na biblioteca de *features*, através da decomposição volumétrica baseada em critérios geométricos, *setup* da peça e nas operações de usinagem. O método é o mais geral possível tratando a maior parte das geometrias de peças rotacionais simétricas e as principais *features* não-concêntricas presentes em peças assimétricas rotacionais, que se pode conceber, tendo a seguinte abrangência:

- a peça bruta (*blank*) do tipo barra, cilindro, tubo e fundida;
- o *setup* da peça, do tipo placa, placa e contra-ponta e entre pontas;
- e operações de torneamento longitudinal, faceamento, perfilamento/cópia, mandrilamento, furação, alargamento, recartilhamento, sangramento, corte de barra, furação e fresamento.

Este novo método propõe uma abordagem de decomposição volumétrica baseada nos três aspectos fundamentais para determinação dos volumes de material a serem removidos da peça bruta. Em algumas abordagens geométricas (DONG e VIJAYAN, 1997) trabalha-se apenas com geometria externas,

¹*Setup* está associado com as atividades de fixação da peça no centro de torneamento, que no caso utiliza-se de uma placa pneumática de três castanhas, com ou sem o uso de contra-ponta. Também é necessário que a peça tenha uma *feature* do tipo ID ou OD com no mínimo 5 mm de comprimento de contato para fixação e *feature* de face retilínea para ser utilizada como superfície de apoio.

não contemplando geometrias internas. Outra abordagem baseada em geometria e operações, descrita em CHO *et al.* (1994), modela apenas linhas, polígono, descrevendo somente as superfícies externas. Segundo CHO *et al.* (1994), LEE *et al.* (1993), RICO *et al.* (2000), DONG e VIJAYAN (1997) e SHAH & MANTYLA (1995) várias abordagens fazem a decomposição baseando-se em aspectos geométricos e/ou em aspectos de operações de usinagem; e alguns em aspectos geométricos e em operações de usinagem.

Esta nova abordagem também contempla *features* concêntricas do tipo *splines* e não-concêntricas (Eixo C) presentes em centros de torneamento, que normalmente não são tratadas pela maioria dos sistemas relatados na literatura, podendo ser considerado como mais uma contribuição para o planejamento de processos de peças rotacionais simétricas e assimétricas (Eixo C).

A seguir são apresentados os três métodos que compõem a decomposição orientada ao *setup*, à geometria e à operação. Estes métodos determinarão as alternativas para *setup* da peça, volumes a serem removidos e operações de usinagem com alternativas para remoção dos volumes determinados, gerando assim um plano de processo não-linear.

7.1.1 Decomposição Orientada ao Setup

A decomposição orientada ao *setup* leva em consideração a quantidade de *setups* da peça, definida em função dos aspectos de tolerâncias geométricas, qualidade superficial, estabilidade e rigidez da peça, modelo geométrico e modelo funcional do dispositivo de fixação. Tem-se como saída deste modelo de decomposição orientada ao *setup* a definição do tipo, número de *setups* e dos grupos de *features* a serem processados em cada *setup*.

É feita uma análise dos requisitos tecnológicos para seleção do sistema de fixação da peça baseado nos mesmos procedimentos adotados por um processista na definição da fixação da peça:

- análise do desenho da peça (tolerância dimensional, geométrica e acabamento superficial);
- define quais os elementos geométricos que serão usinados no mesmo *setup* (grupos) para garantir os requisitos tecnológicos;
- define o sistema de fixação para os diversos *setups*.

As tolerâncias de uma peça podem ser categorizadas em tolerâncias locais e relativas. Uma tolerância local está relacionada com uma única *feature* e é determinada principalmente pela capacidade do processo/máquina. Tolerância relativa está relacionada com mais de uma *feature* e é influenciada pela capacidade da máquina/processo, mas também pela seleção de referências (*datum*) e *setup* no planejamento do processo. Quando uma determinada dimensão de desenho (cota) não pode ser obtida diretamente é necessário fazer seu cálculo baseado na cadeia de tolerâncias. Logo o controle de tolerância pode ser realizado pelo planejamento do *setup*, sendo que do ponto de vista da fabricação, tolerâncias são erros de fabricação oriundos de muitas fontes (ÁLVARES, 1990) resultando numa variação nas dimensões da peça, de caráter sistemática e aleatória.

A seguir é apresentada a proposta de decomposição orientada ao *setup* com as principais regras para se chegar ao planejamento de *setup* com alternativas, tendo como orientação a escolha do método de *setup* I (método erros dimensionais e geométricos). Caso não seja possível, opta-se pelo método de



Figura 7.1: Grafo e/ou *setup*: alternativas para *setup* da peça quando $\frac{L_{peca}}{D_{peca}} > 4$.

setup II (erro de localização na dimensão). O procedimento a ser adotado é sintetizado nas seguintes regras:

- definir tipo da peça bruta (cilindro, barra, tubo, fundido), denominado de *blank*.
- escolha em função da geometria (superfícies cilíndricas OD e ID da peça bruta e da peça acabada): apenas superfícies OD e ID cilíndricas são candidatas à superfície de fixação no *setup* da peça, desde que o comprimento da superfície cilíndrica e seu diâmetro interno ou externo seja compatível com o modelo geométrico da placa de fixação e da castanha do torno CNC.
- função da estabilidade e rigidez da relação comprimento da peça/diâmetro da peça (figura 7.1): quando a relação $\frac{L_{peca}}{D_{peca}} > 4$, a peça será definida como um eixo, peça longa, sendo necessário um conjunto de fixações do tipo placa e contra-ponta e/ou entre-centro, tendo as seguintes alternativas de fixação: duas fixações do tipo placa e contra-ponta; três fixações sendo duas do tipo placa e contra-ponta para operações de desbaste e a terceira entre-centros para operações de acabamento, quando o torque *driver* (T_d), torque máquina (T_m) e torque placa (T_c), forem da magnitude $T_m \gg T_d$ e $T_c \gg T_d$; ou uma fixação placa e contra-ponta, quando $T_d > T_m$ e $T_d > T_c$. Se alimentação da máquina for feita por barra pode-se ter um único *setup* do tipo placa e contra-ponta tendo como última operação o corte da peça no próprio torno, desde que uma das faces, extremidades da peça, não tenha *features* de projeto que necessitem de uma usinagem.
- função da estabilidade e rigidez da relação comprimento da peça/diâmetro da peça (figura 7.2): quando a relação $\frac{L_{peca}}{D_{peca}} < 2$, a peça será definida como uma peça curta, sendo necessário um conjunto de no máximo duas fixações tipo placa, uma para peça bruta (*Setup* 1) e outra para peça semi-acabada (*Setup* 2). Se a alimentação da máquina for feita por barra pode-se ter um único *setup* do tipo placa tendo como última operação o corte da peça no próprio torno, desde que uma das faces, extremidades da peça, não tenha *features* de projeto que necessitem de uma usinagem.
- função da estabilidade e rigidez da relação comprimento da peça/diâmetro da peça (figura 7.3): quando a relação $2 \leq \frac{L_{peca}}{D_{peca}} \leq 4$, a peça será definida como uma peça intermediária, podendo

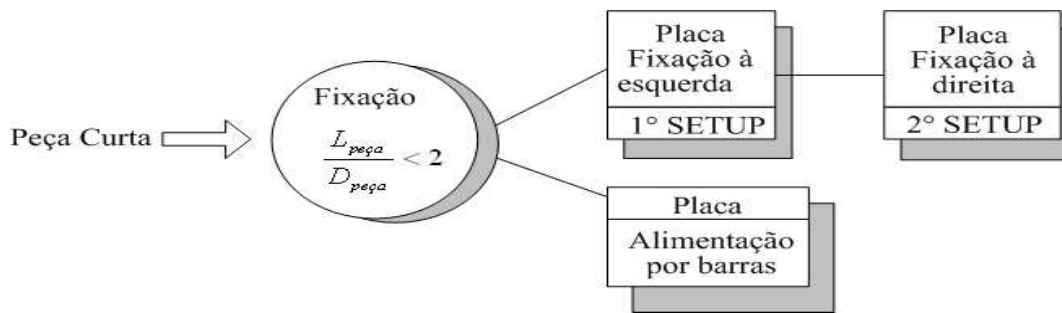


Figura 7.2: Grafo e/ou *setup*: alternativas para *setup* da peça quando $\frac{L_{peça}}{D_{peça}} < 2$.

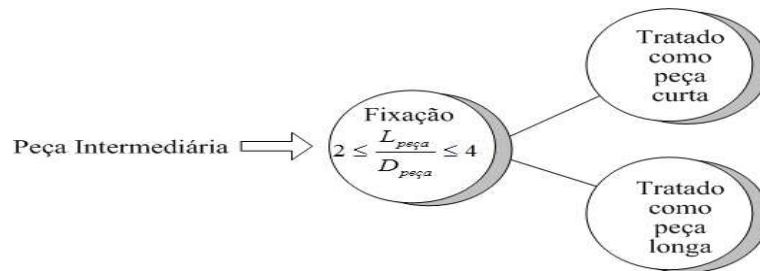


Figura 7.3: Grafo e/ou *setup*: alternativas para *setup* da peça quando $2 \leq \frac{L_{peça}}{D_{peça}} \leq 4$.

ser tratada em termos da quantidade e tipo de fixações como uma peça curta ou uma peça longa, em função da sua rigidez e estabilidade e das *features* de forma presentes.

- peças com *features* ID vazadas são fixadas com placas em um ou dois *setups*.
- qualidade superficial: eliminar as superfícies OD e ID com qualidade superficial apertada ($0,2 \mu m$) da lista de candidatas à superfícies de *setup*.
- tolerâncias dimensionais: eliminar as superfícies OD e ID com tolerâncias apertadas (IT 7) da lista de candidatas à superfícies de *setup*.
- tolerância geométrica (concentricidade, paralelismo, perpendicularismo e batimento): *features* com requisitos de tolerâncias de posição, orientação, localização ou batimento deverão ser agrupadas no mesmo *setup*.
- Definir as alternativas de superfícies de fixação para o *blank* e para a peça acabada a partir do modelo funcional, que relaciona a interação de esforços entre a castanha e a peça. De acordo com a diretriz VDI 3106 (MASSAROPPI e MASIERO, 2001) as possibilidades de contato entre castanha e peça podem ser divididas em: pelo centro, pelas extremidades ou completo, onde o raio de curvatura da superfície de fixação da castanha é maior, menor e igual ao raio da peça, respectivamente. Durante a usinagem a solicitação crítica (MASSAROPPI e MASIERO, 2001) é aplicada na extremidade da peça (Z_{ext}), no ponto mais distante do centro de gravidade da fixação (Z_{cg}). Segundo MASSAROPPI e MASIERO (2001) a fixação mais segura para a peça durante o processo de usinagem é aquela que apresentar o menor índice de fixação, dado pela

razão entre o acréscimo da densidade de carga (Δqm) provocado pela força de usinagem e a própria força de usinagem. Assim I_f (Índice de Fixação) é:

$$I_f = \frac{(F + \Delta qm)}{F} \quad (7.1)$$

$$\Delta qm = q_{max} - q_0 = 4 * \frac{M}{L^2} = 4 * F * \frac{(Z_{ext} - Z_{cg})}{L^2} \quad (7.2)$$

Logo, quanto maior for o comprimento de contato (L) e menor for a distância entre o ponto de aplicação da carga ao centro de gravidade de contato, menor será o Índice de Fixação, tendo assim uma melhor qualidade de fixação. Finalmente seleciona-se as alternativas de superfícies cilíndricas OD e ID candidatas aos *setups*: *setup 1 (blank)*, *setup 2* e *setup 3*.

- subdividir o *blank* e a peça final em grupos para cada *setup*, levando em conta as superfícies de fixação definidas.
- para cada superfície de fixação, *features* ID e OD próximas às faces esquerda e direita da peça, define-se uma Linha de Demarcação (LD) no ponto de início e no ponto final da *feature* em análise. O número de combinações será: 2 x (Número de *features* OD x Número de *features* ID).
- para cada grupo (alternativa) verifique se existe alguma definição de *features* de tolerâncias de posição, orientação, batimento ou localização. Exclua aqueles que não atendam ao requisito de agrupamento de tolerâncias geométricas em um mesmo *setup*.
- verifique requisitos de qualidade superficial agrupando *features* com a mesma qualidade.
- escolha a melhor alternativa para os *setups*.
- definir os grupos de *features* por *setup*.

7.1.2 Decomposição Orientada à Geometria

Após a determinação do número de *setups*, para a melhor alternativa, e a definição dos grupos de *features* por *setup*, é realizada a decomposição de volumes baseada na geometria do *blank* e da peça final. O método idealizado é baseado na decomposição de uma poliforma² onde o mapeamento de *features* é feito a partir da determinação do volume a ser removido por usinagem levando em conta a geometria da peça acabada menos a peça bruta.

O volume de usinagem global é dividido em *features* de usinagem usando um algoritmo de varredura de linhas horizontal e vertical. Para diâmetros externos é realizada a varredura para *features* de forma do tipo OD de maior diâmetro para *features* OD de diâmetro imediatamente inferior, determinando-se os vértices desta poliforma. A seguir repete-se o procedimento até concluir a varredura de todos os diâmetros externos das *features* OD.

²Poliforma é uma forma geométrica fechada formada por uma sucessão de linhas e arcos conectados, que descrevem em 2D a área final da peça rotacional e a área global de material a ser removido do *blank*. Na realidade a área (2D) está relacionada ao volume do sólido rotacional e ao volume global de material a ser removido do *blank*.

Para diâmetros internos repete-se o mesmo procedimento de forma inversa às *features* OD, ou seja, é feita a varredura para *features* de forma do tipo ID do menor diâmetro para *features* ID de diâmetro imediatamente superior. O procedimento é repetido até varrer o menor diâmetro das *features* ID que constituem a peça.

As *features* OD de maior diâmetro e ID de menor diâmetro de uma peça são as superfícies que definem uma possível inversão no perfil crescente ou decrescente da poliforma, devendo portanto ser examinados detalhadamente.

A seguir são apresentadas as regras e heurísticas associadas ao método de decomposição geométrica:

- O modelo de *features* de projeto é transformado em um modelo geométrico 2D, denominado de poliforma fechada, formado por segmentos de retas, arcos e *splines* conectados de maneira lógica (figura 7.4).
- Faz-se uma varredura com uma linha de referência horizontal (diâmetro= coordenada X= constante) para *features* OD e para *features* ID de cada grupo de *setups*, do maior para o menor diâmetro e do menor para o maior diâmetro, respectivamente, definindo-se os volumes de material a serem removidos para cada grupo de *setup*. A seguir identificam-se os volumes de material a serem removidos por operações de torneamento, mandrilamento e perfilamento.
- Faz-se uma varredura com uma linha de referência vertical (comprimento longitudinal = coordenada Z = constante) para *features* de faces nas extremidades da peça para os grupos de *setups*, do maior para o menor comprimento longitudinal, definindo-se os volumes de material a serem removidos para cada grupo de *setup*. A seguir identificam-se os volumes de material a serem removidos por operações de faceamento, face esquerda e face direita.
- Roscas e superfícies recartilhadas são atributos de *feature* ID ou OD e OD, respectivamente. Os volumes de material a serem removidos são identificados, gerando o modelo geométrico associado às *features*.
- Canais e furos são identificados separadamente em função do tipo de *features* representado no modelo de *features*. A seguir os volumes de material a serem removidos são identificados para cada *feature*.
- As *features* de Eixo C (fresamento e furação) são determinadas independentemente, pois já estão identificadas na biblioteca de *features*, e seus volumes já são conhecidos previamente, devido ao método de modelamento utilizado, que é baseado no projeto por síntese de *features* subtrativas, muito similar ao método DSG. Em função da localização da *feature* na peça a usinagem poderá ocorrer no primeiro ou no segundo *setup* da peça.

7.1.3 Decomposição Orientada à Operação

Após a identificação do grupo de *features* por *setup*, dos volumes de material a serem removidos (levando em consideração a geometria do *blank* e da peça acabada para cada grupo de *features* por

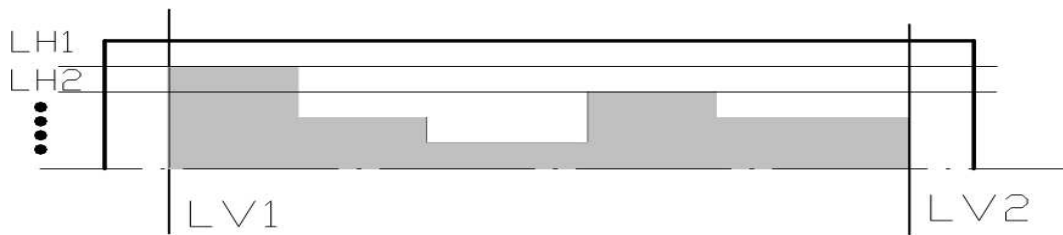


Figura 7.4: Poliforma fechada representando o perfil 2D da peça bruta e peça acabada.

Tabela 7.1: Operações de usinagem modeladas na decomposição orientada à operação.

Operações	Tipo de <i>feature</i> de usinagem associada e direção
Torneamento	Volume associado ao perfil da peça na direção horizontal
Faceamento	Volume associado ao perfil da peça na direção vertical
Perfilamento	Volume associado ao contorno da peça sem direção preferencial
Furação	Volume associado a um furo
Canal/Reentrância	Volume associado a um canal ou reentrância na peça pode ser curto ou largo
Recartilhamento	Atributo da <i>feature</i> de usinagem pai (superfície primária e secundária)
Fresamento	Volume associado a uma cavidade, rasgo de chaveta, canal cilíndrico, etc
Alargamento	Volume associado a um furo
Escareamento	Atributo da <i>feature</i> de usinagem pai (superfície primária e secundária)
Rosqueamento	Atributo da <i>feature</i> de usinagem pai (superfície primária e secundária)

setup) é necessária a decomposição dos volumes globais de material a serem removidos, previamente identificados, em *features* de usinagem, associando um sub-volume de um volume maior à uma operação de usinagem e um determinado tipo de ferramenta.

Em algumas situações as *features* de usinagem decompostas pelo método geométrico não podem ser usinadas em apenas uma operação (figura 7.5 c). Assim as *features* de usinagem devem ser decompostas em outras *features* de usinagem (figura 7.5). Por exemplo, canal (largo ou estreito), faceamento, torneamento e *feature* de perfil. Logo pode-se ter muitas alternativas de operações de usinagem para uma determinada *feature* de usinagem, no caso de torneamento.

Deve-se levar em consideração os tipos de operações de usinagem a serem utilizadas para remoção dos volumes identificados, gerando alternativas de operações para uma mesma *feature* de usinagem. As operações de usinagem modeladas são apresentadas na tabela 7.1.

Assim roscas e superfícies recartilhadas (superfícies secundárias) são mapeadas através da classe de mapeamento 1:1, sendo atributos de uma *feature* primária, do modelo de *features*, e usinadas por operações de rosqueamento e recartilhamento. É necessário definir apenas o sobrematerial necessário a ser deixado pela operação de torneamento anterior. Já Furos são mapeados através da classe de mapeamento 1:1 sendo uma *feature* primária.

As demais *features* concêntricas de projeto são mapeadas através da classe de mapeamento 1 : muitos (1 => n) e muitos : 1 (m => 1), levando em consideração as alternativas de operações e ferramentas a serem utilizadas na remoção dos volumes associados às *features* de usinagem identificadas.

Um determinado volume de material a ser removido, associado ao perfil da peça cilíndrica, pode ser decomposto em operações de torneamento ou faceamento (fig. 7.6). Uma operação de torneamento tem preferência sobre uma operação de faceamento, quando a dimensão Z de usinagem (di-

reção longitudinal) é maior que a dimensão X de usinagem (direção radial, associado ao diâmetro da peça). Por outro lado, uma operação de faceamento tem preferência sobre uma operação de torneamento, quando a dimensão X de usinagem (direção radial, associado ao diâmetro da peça) é maior que dimensão Z de usinagem (direção longitudinal). Utilizando esta heurística obtém-se um bom resultado para o deslocamento de ferramenta em termos de tempo, não sendo necessário analisar todas as alternativas de sub-volumes de material (*feature* de torneamento) que poderia gerar um roteamento alternativo (fig. 7.6).

Uma ranhura estreita ou larga, ou mesmo uma reentrância, que pode ser considerada uma ranhura larga, podem ser usinadas tendo as seguintes alternativas, conforme apresentado na figura 7.5 (c), considerando um único *setup*:

- reentrância/canal usinado apenas com a utilização de uma ferramenta de sangramento (bedame);
- reentrância usinada com uma ferramenta de sangramento e uma segunda para torneamento (uma ferramenta esquerda para direita, neutra ou uma ferramenta direita para esquerda com ângulo de abordagem compatível com a geometria cônica);
- reentrância com usinagem zig-zag, onde é feita a usinagem com ferramenta direita para esquerda, deixando uma superfície intermediária e depois a usinagem é completada com uma ferramenta esquerda para direita, a fim de remover a superfície intermediária. Outra alternativa é utilizar uma ferramenta neutra com ângulo de abordagem compatível com a geometria cônica da saliência.

7.2 Determinação das Operações de Usinagem Associadas às *Features: Workingsteps*

A atividade concebida de mapeamento de *features* determina as *features* de usinagem associando-as às operações de usinagem com alternativas, considerando a geometria da peça bruta e da peça acabada, *setup* e operações de usinagem. Para representar todas estas alternativas utiliza-se um grafo E/OU que relaciona as várias possibilidades de operações de usinagem, geometria das ferramentas e *setup*, para gerar uma determinada *feature* de usinagem.

Uma operação de usinagem define o processo de usinagem para uma área limitada da peça denominada de *feature* de usinagem. Define-se assim o conteúdo de um *workingstep* de usinagem (*Machining_workingstep*³) ou de torneamento (*Turning_workingstep*⁴) associando a operação de usi-

³Segundo STEP-NC Part 10, um *Machining_workingstep* associa uma operação de usinagem a uma única *feature* de usinagem, incluindo também um plano de segurança.

⁴Segundo STEP-NC Part 12, um *Turning_workingstep* associa uma operação de usinagem a uma lista de *features* concêntricas de usinagem (duas ou mais), incluindo também um plano de segurança. Esta definição viola a definição de Part 10. Pela Part 10 não pode haver ambiguidade de relações de "n" *features* de usinagem e 1 operação (n:1), não permitindo uma lista de *features* de usinagem, conforme definido em Part 12. *Workingstep* está sempre associado com 0 ou 1 *feature* de usinagem. Uma *feature* de usinagem pode estar associada a "n" operações de usinagem (cardinalidade 1 => n). Já *Machining_workingstep* pode estar associada a uma única *feature* de usinagem. Esta contradição contínua pendente até as próximas revisões do *draft* da norma atual. Devido as características peculiares do torneamento, é necessário esta adequação na norma, permitindo este relacionamento.

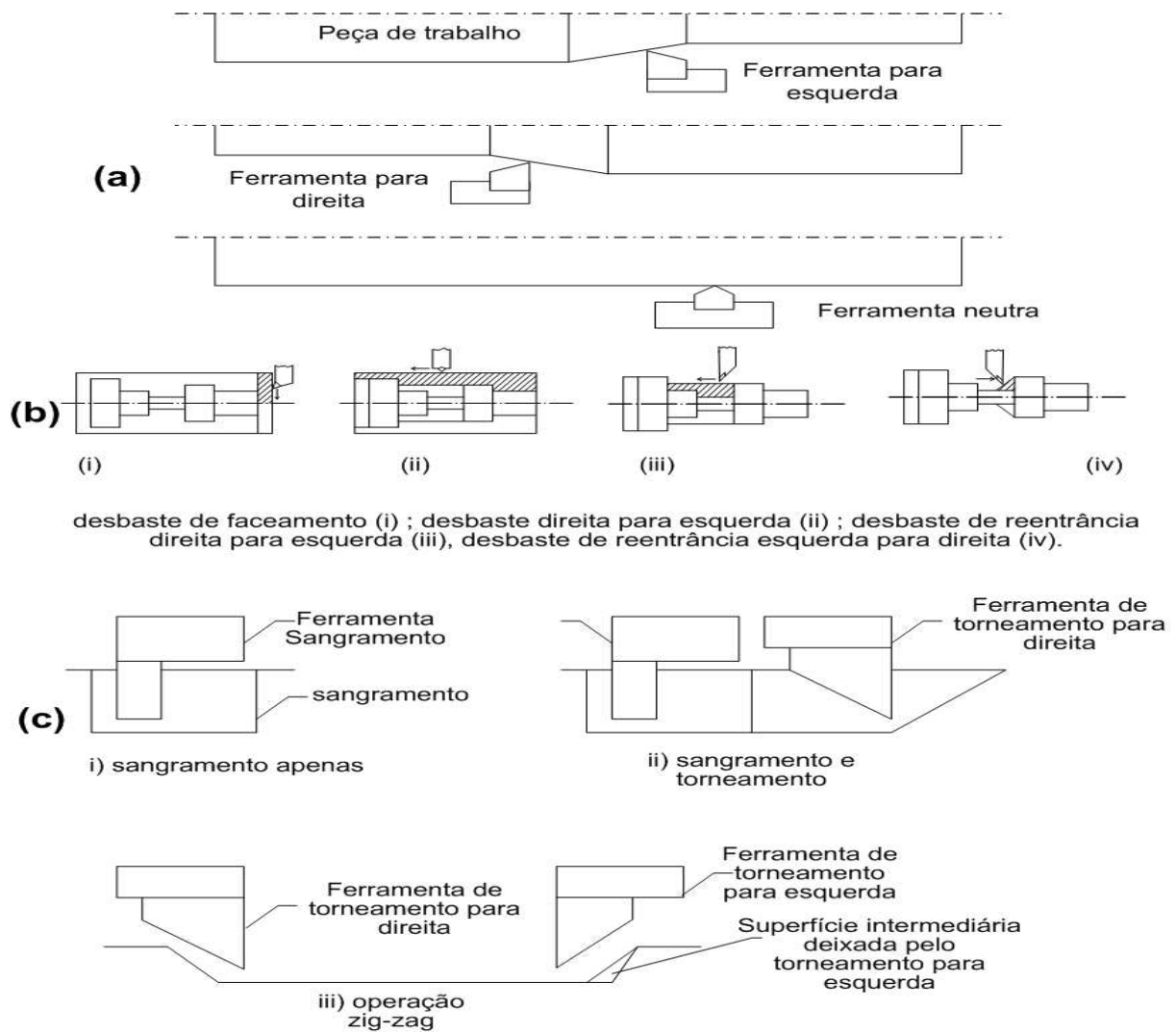


Figura 7.5: (a) Ferramentas: direita para esquerda, esquerda para direita e neutra (CHANG *et al.*, 1998) (b) Operações mais usuais em peças de revolução (MAZIERO e NASCIMENTO, 2003) (c) métodos de usinagem de canal/reentrância (YANG e LEE, 1998).

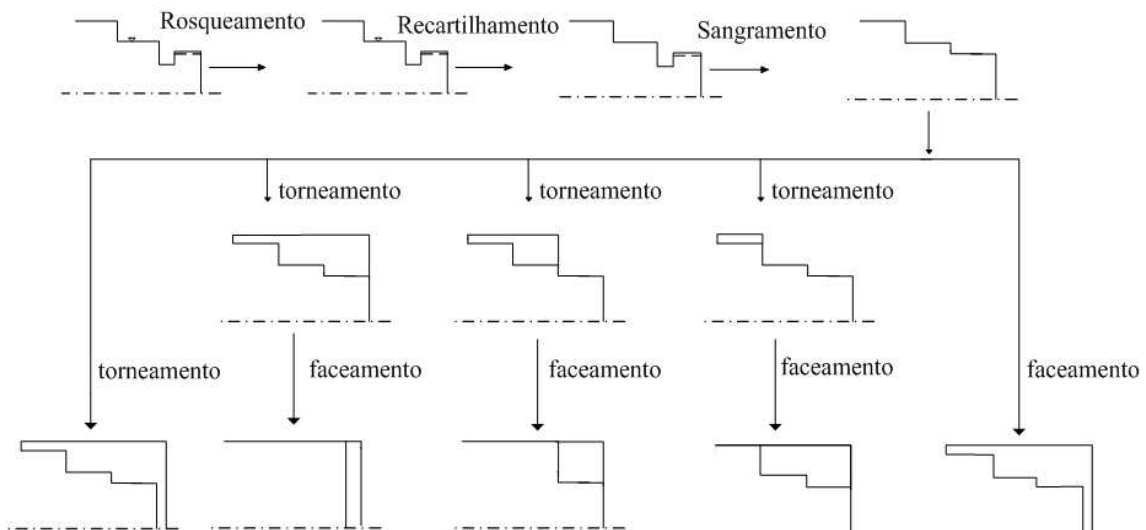


Figura 7.6: Um exemplo do procedimento associado a uma árvores de decisão (HUANG, 1988), apresentando alternativas de volumes e operações de usinagem para *features* concêntricas.

Tabela 7.2: Classificação das *features* reconhecidas pelo método.

<i>Features</i> Concêntricas	<i>Features</i> Eixo C
rosca e recartilhado	cavidade e rasgo
canal, corte, alívio e reentrância	furo
arredondamento, filete e chanfro	escareado
face	alargamento
perfil externo (ODs)	rosca
perfil interno (IDs)	
<i>spline</i>	
furo	

nagem à uma *feature* de usinagem, definindo no mínimo a ferramenta e os parâmetros tecnológicos de usinagem segundo STEP-NC.

Define ainda, em uma operação de usinagem, o movimento de abordagem/aproximação da ferramenta em relação à peça, definindo o ponto de início de usinagem, bem como o movimento de afastamento, especificando o plano de afastamento (plano de retração) onde a ferramenta se posicionará após o passe de usinagem. Estes planos são associados à uma estratégia de movimentação de ferramenta. Define-se também um plano de segurança onde a ferramenta poderá ser posicionada evitando o choque com a peça.

Utiliza-se uma árvore de decisão no planejamento da seqüência de operações para usinagem das regiões da peça associadas ao perfil externo da peça (figura 6.11 (a)), e uma árvore para o perfil interno da peça (figura 6.11(b)), levando em consideração a quantidade de *setups* e o diâmetro interno, associados às operações de furação e mandrilamento.

A árvore é associada ao procedimento de mapeamento de *features* gerando um plano de processo com alternativas de operações de usinagem, *setup* e ferramentas (figuras 6.11 e 7.6).

As *features* concêntricas presentes em peças torneadas são: diâmetro, face, rosca, cônico, chanfro, arco, canal, reentrância, alívio, recartilhado, filete e chanfro. As *features* Eixo C, que exigem usinagem com ferramenta rotativa, presentes em peças rotacionais assimétricas são: furo⁵, cavidade, rasgo de chaveta, face, rosca, escareado, canal, rosca, padrão de furos, padrão de rasgos, entre outros. As *features* consideradas são classificadas e apresentadas na tabela 7.2.

À cada *feature* mapeada, a área ou perfil de usinagem associado à uma operação de desbaste é determinada. É previsto um sobrematerial para as operações de semi-acabamento e/ou acabamento. As operações de acabamento consistem de: perfilamento/cópia, rosqueamento, torneamento, faceamento, sangramento, mandrilamento⁶, fresamento, furação, escareamento e alargamento. A operação final de acabamento constitui-se de um único passe. As operações de desbaste consistem de faceamento, torneamento, sangramento, mandrilamento, fresamento e furação. A profundidade de corte

⁵As operações de furação axial podem ser executadas com ferramenta acionada ou fixa. Para ferramenta acionada pode-se furar diâmetros menores que 16 mm, que é o máximo diâmetro compatível com a ferramenta acionada. Para ferramenta não-acionada tem-se disponível um porta-ferramenta para furação de 20 mm. Assim pode-se trabalhar no desbaste com furos axiais a partir de 20 mm.

⁶O menor diâmetro que se pode usinar com as barras de mandrilar disponíveis é da ordem de 22 mm. Assim o torneamento interno é viável a partir de furos de 22 mm. Se a peça bruta não for vazada, faz-se o furo passante com ferramenta fixa para um diâmetro de no mínimo 22 mm. A seguir utiliza-se uma barra de mandrilar.

ainda não é conhecida e só será calculada depois que uma apropriada ferramenta de corte tenha sido selecionada e as condições de usinagem tenham sido otimizadas.

As *features* de projeto são mapeadas em *features* de usinagem, mais especificamente em *features* de torneamento (*features* concêntricas) e *features* de usinagem (*features* não-concêntricas, fresamento e furação, Eixo C) de acordo com a especificação das *features* de torneamento e de *features* de usinagem definidas na ISO 14649 - Part 12 e Part 11 (2003), sendo incluídas as seguintes *features* de torneamento⁷ definidas na proposta Part 12 e as *features* de usinagem incluídas na Part 10 e Part 11:

- Part 12: *outer_round* (*outer_diameter*, *outer_diameter_to_shoulder*), *revolved_feature* (*groove*, *revolved_flat*, *revolved_round*, *general_revolution*), *knurl* (*straight_knurl*, *diagonal_knurl*, *diamond_knurl*, *catalogue_knurl*);
- O Anexo C da proposta de norma ISO 14649 - Part 12 inclui alguns tipos de *features* de torneamento que são normalmente utilizadas no chão-de-fábrica como: *cut_in* e *circular_face*, *bottom_transition*, *bottom_transition_slope*, *bottom_transition_round*, *face_radiused*;
- Part 10 e Part 11: *planar_face*, *pocket* (*closed_pocket*, *open_pocket*), *slot*, *step*, *round_hole*, *profile_feature* (*general_outside_profile*, *shape_profile*), *toolpath*, *boss*, *rounded_end*, *compound*, *replicate* (*rectangular_pattern*, *circular_pattern*, *general_pattern*), *thread*, *transition* (*chamfer*, *edge_round* (*fillet*)), *profile* (*closed_profile*, *open_profile*);
- além destas são incluídas as *features* associadas às operações de torneamento internas (mandrilamento) semelhantes às *features* de torneamento externa e furação, que não foram incluídas na Part 12, e estão descritas na ISO 10303 AP224 e ISO 14649 - Part 10.

No item seguinte são elaborados grafos de precedência de *features*, que relacionam a precedência de uma *feature* em relação a outra, restringindo as possibilidades de combinações de seqüência de usinagem e minimizando o problema de otimização da determinação da seqüência de usinagem.

A descrição das alternativas de fabricação para cada uma das *features* de usinagem é apresentada a seguir, podendo subdividir as operações de usinagem em operações de desbaste, semi-acabamento e acabamento em função dos requisitos de qualidade definidos para a *feature* de projeto⁸.

Utilizou-se para as *features* de torneamento concêntricas uma nomenclatura utilizada comumente na indústria e na academia, para fugir da terminologia em língua inglesa adotada no STEP-NC, que ainda não possui uma tradução para língua portuguesa. Já se incluiu na descrição da operação os tipos de ferramentas que poderão ser utilizadas, apresentando assim as alternativas disponíveis para fabricação da *feature* de usinagem, descrevendo então um *workingstep* de usinagem, associando uma *feature* de usinagem com alternativas de operações de usinagem (cardinalidade 1 *feature* : n operações). Mais a frente são incluídas as estratégias de movimentação de ferramentas na descrição das operações de usinagem.

⁷ *Feature* de torneamento é uma classe base abstrata para todas as *features* de usinagem usadas em torneamento. As *features* de torneamento definidas são classificadas como formas geométricas que podem ser obtidas pelo torneamento de peças cilíndricas com operações de dois eixos (X e Z) ou operações em 3 eixos (Z, X e C). As *features* que podem ser obtidas por operações em três eixos não são tratadas pela norma STEP-NC Part 12.

⁸ Considere a descrição da geometria da peça da direita para a esquerda, associada à fixação da peça em um torno CNC.



Figura 7.7: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo OD (Diâmetro Externo) cilíndrica/retilínea.

Utiliza-se um mapeamento 1:1 para as *features* de usinagem não-concêntricas (Eixo C), tendo-se alternativas apenas com relação ao tipo de ferramenta utilizada e a estratégia de usinagem associada. Nesta tese não se considerou alternativas de operações para Eixo C, já amarrando o processo de usinagem à *feature*. Foge do escopo deste trabalho a proposição de novos algoritmos para se usinar uma cavidade, por exemplo. São utilizadas estratégias de usinagem pré-definidas e lineares, sem avaliação de alternativas.

7.2.1 Grafos E/Ou Para Operações de Desbaste

A seguir são apresentados os grafos E/OU para as *features* concêntricas associadas às operações de desbaste. A partir destes grafos E/OU pode-se definir as alternativas de operações de desbaste para cada *feature* de torneamento mapeada.

1. OD cilíndrico/retilíneo: torneamento com ferramenta para direita, para esquerda ou neutra (figura 7.7).
2. OD cônica para dentro da peça (*cut-in*): torneamento com ferramenta para direita⁹; ou ferramenta para esquerda¹⁰ com ângulo de abordagem da ferramenta compatível com a inclinação do cone. Outra alternativa é utilizar uma ferramenta neutra (figura 7.8).
3. OD cônica para fora da peça (*cut-out*): torneamento com ferramenta para esquerda; ou ferramenta para direita com ângulo de abordagem da ferramenta compatível com a inclinação do cone. Outra alternativa é utilizar uma ferramenta neutra (figura 7.9).
4. Canal estreito (canal): sangramento da peça com ferramenta compatível com a dimensão do canal.
5. Canal largo (reentrância) com as duas faces perpendiculares: sangramento da peça com ferramenta compatível com a dimensão do canal, sendo necessário vários passes ao longo da direção X a partir das faces perpendiculares para o centro; ou torneamento com ferramenta neutra

⁹Na terminologia adotada ferramenta para direita está associada com a direção do movimento da ferramenta (avanço) da esquerda para a direita, ou seja, ferramenta lado esquerdo (*left-hand tool*).

¹⁰Na terminologia adotada ferramenta para esquerda está associada com a direção do movimento da ferramenta (avanço) da direita para a esquerda, ou seja, ferramenta lado direito (*right-hand tool*).

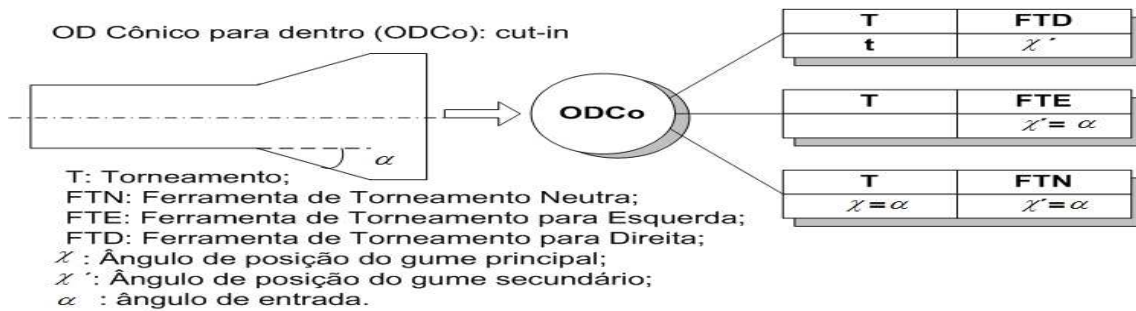


Figura 7.8: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo OD (Diâmetro Externo) cônica para dentro (*cut-in*).

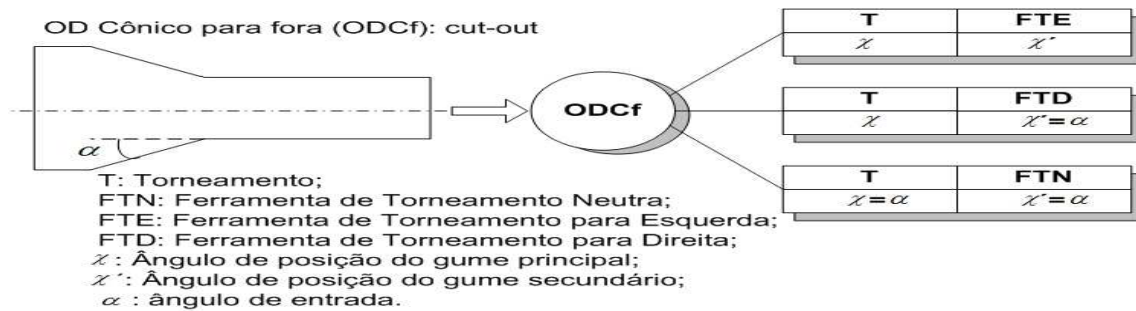


Figura 7.9: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo OD (Diâmetro Externo) cônica para fora (*cut-out*).

com ângulo de abordagem da ferramenta adequado + sangramento; ou torneamento com ferramenta para esquerda com ângulo de abordagem da ferramenta adequado + torneamento com ferramenta para direita com ângulo de abordagem da ferramenta adequada (usinagem zig-zag) (figura 7.10).

- Canal largo (reentrância) com uma face perpendicular à esquerda e uma superfície cônica à direita, para dentro da peça (*cut-in*): sangramento com ferramenta compatível + torneamento longitudinal com ferramenta de sangramento; ou sangramento + torneamento com ferramenta para direita; ou sangramento + torneamento com ferramenta para esquerda com ângulo de abor-

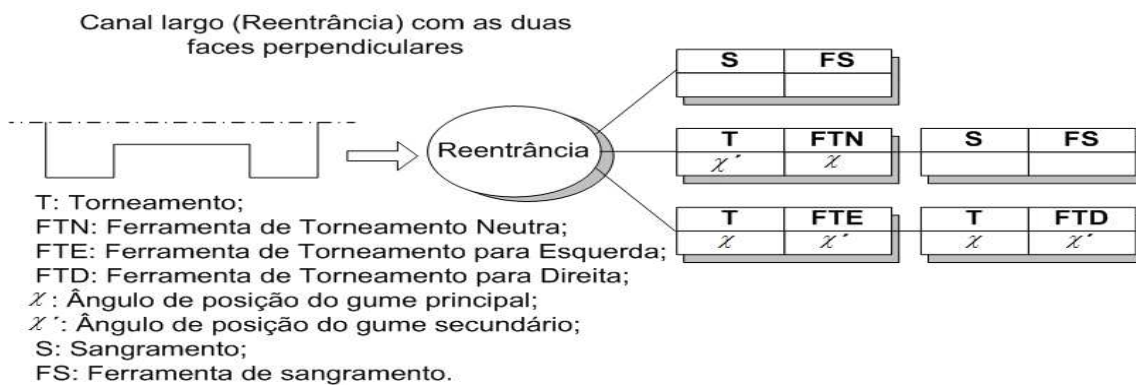


Figura 7.10: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo canal estreito.

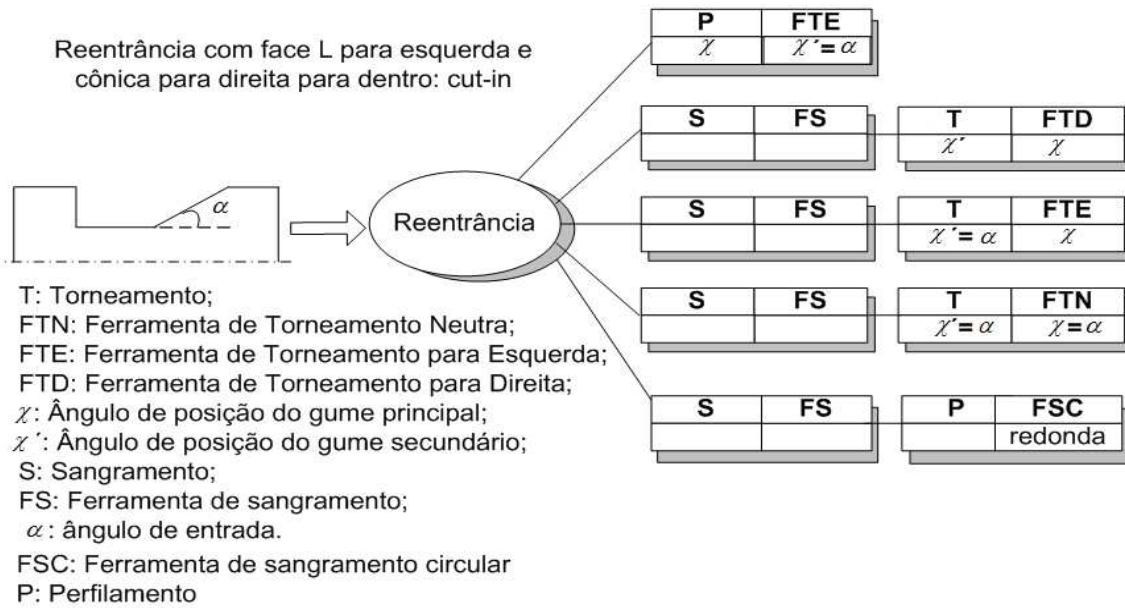


Figura 7.11: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo canal largo, reentrância, com as duas faces perpendiculares.

dagem da ferramenta compatível com a inclinação do cone; ou sangramento + torneamento com ferramenta neutra com ângulo de abordagem da ferramenta compatível com a inclinação do cone; ou sangramento + sangramento com ferramenta circular compatível (figura 7.11).

7. Canal largo (reentrância) com uma face perpendicular à direita e um superfície cônica à esquerda, para fora da peça (*cut-out*): sangramento com ferramenta compatível + torneamento longitudinal com ferramenta de sangramento; ou sangramento + torneamento com ferramenta para esquerda, ou sangramento + torneamento com ferramenta para direita com ângulo de abordagem da ferramenta compatível com a inclinação do cone; ou sangramento + ferramenta neutra com ângulo de abordagem da ferramenta compatível com a inclinação do cone; ou sangramento + sangramento com ferramenta circular compatível (figura 7.12).
8. Canal largo (reentrância) com superfície cônica à esquerda (para fora da peça, *cut-out*) e superfície cônica à direita (para dentro da peça, *cut-in*): sangramento com ferramenta compatível + torneamento longitudinal com ferramenta de sangramento; ou torneamento utilizando ferramenta para direita + torneamento utilizando ferramenta para esquerda (usinagem zig-zag); ou torneamento utilizando ferramenta para esquerda com ângulo de abordagem da ferramenta compatível com a inclinação do cone à direita; ou torneamento com ferramenta neutra com ângulo de abordagem de ferramenta compatível com a inclinação dos cones à direita e à esquerda; ou sangramento com ferramenta circular compatível (figura 7.13).
9. Face: faceamento normalmente realizado na face esquerda e/ou direita da peça (onde a dimensão radial/diametral X é maior que a dimensão longitudinal Z); ou ferramenta de sangramento (7.14).
10. Chanfros: são *features* modificadores, transição, sendo tratadas como se fossem *features* do tipo OD cônica, com operação de torneamento bi-axial (movimento simultâneo nas coordenadas X

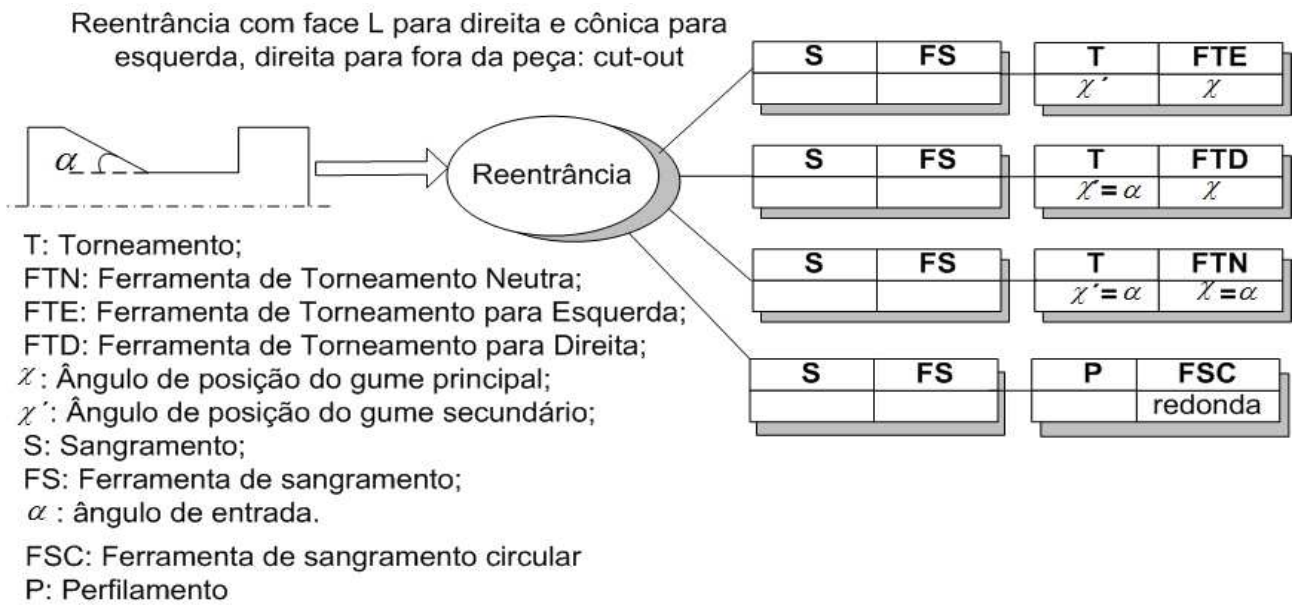


Figura 7.12: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo canal largo, reentrância, com uma face perpendicular à esquerda e uma superfície cônica à direita, para dentro da peça (*cut-in*).

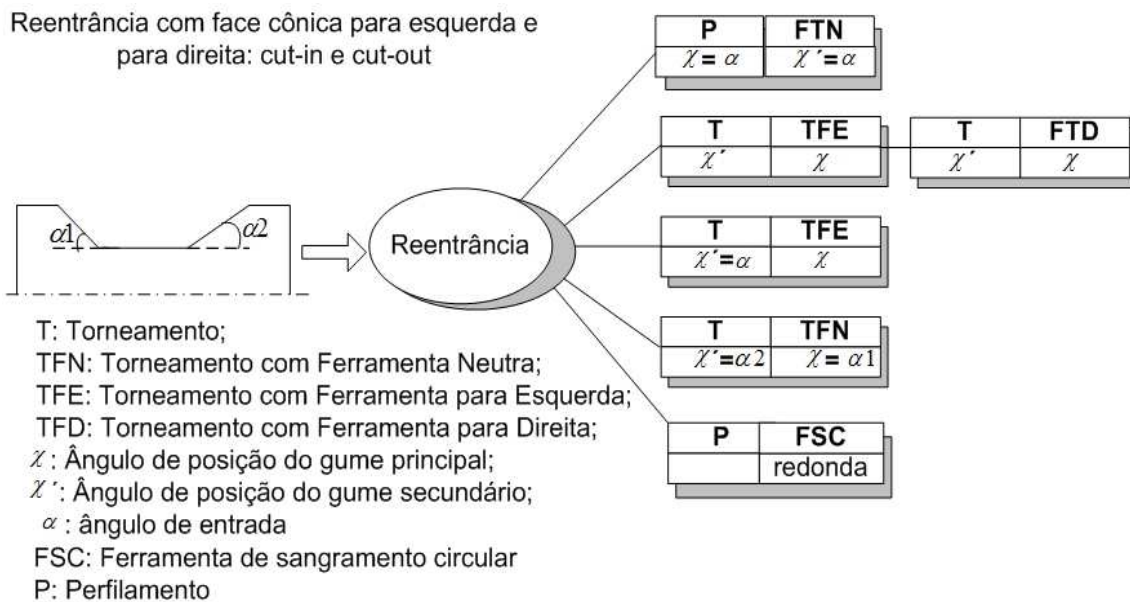


Figura 7.13: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo canal largo, reentrância, com uma face perpendicular à direita e uma superfície cônica à esquerda, para fora da peça (*cut-out*).

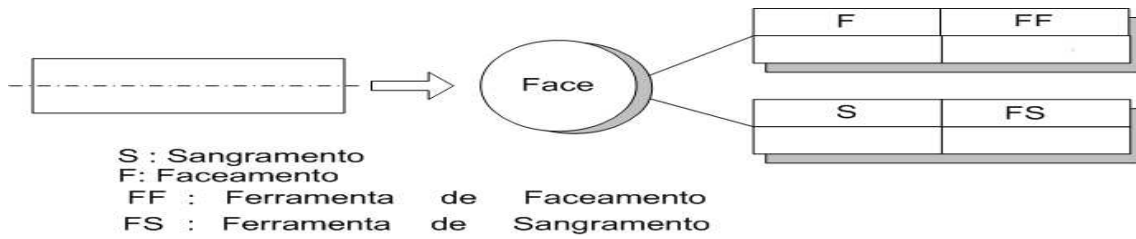


Figura 7.14: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo face.

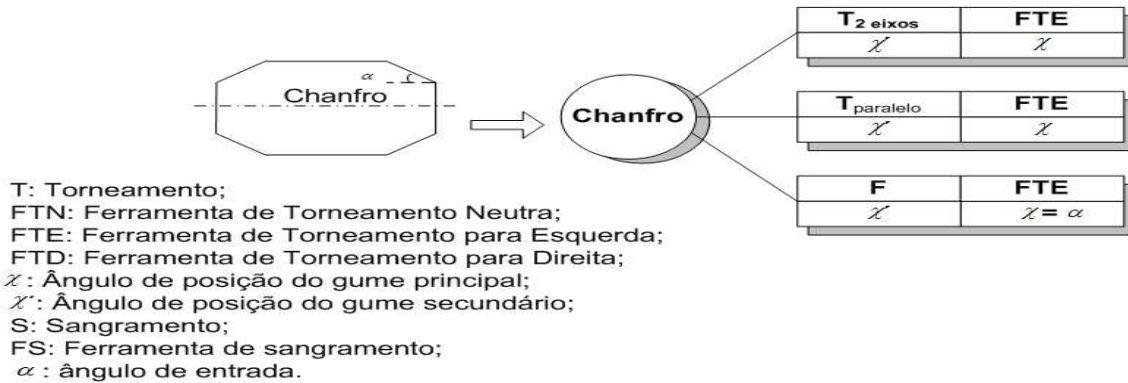


Figura 7.15: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo chanfro.

e Z paralelo ao cone); ou operação de torneamento paralela (coordenada X constante para cada passe) acompanhando o perfil da peça; ou operação de faceamento com ferramenta com ângulo de abordagem compatível com a inclinação do cone (figura 7.15).

11. Filetes e arredondamentos (superfícies convexas e côncavas): são *features* modificadores, transição, sendo tratadas como se fossem *features* do tipo OD convexas ou côncavas, com operação de torneamento bi-axial (movimento simultâneo nas coordenadas X e Z paralelo ao arco); ou operação de torneamento paralela (coordenada X constante para cada passe) acompanhando o perfil da peça; ou operação de perfilamento com ferramenta circular (figura 7.16).
12. Furos axial passante ou cego: furação com broca helicoidal, utilizando ferramenta acionada ou não-acionada.



Figura 7.16: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo filete ou arredondamento.

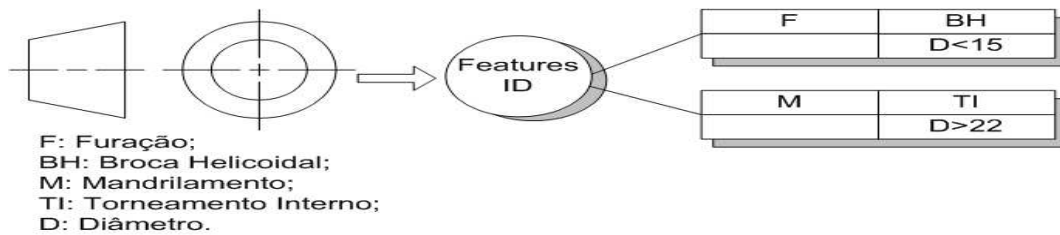


Figura 7.17: Grafo E/OU apresentando alternativas de usinagem para *feature* ID.

13. Furos de centro: furação com brocas de centros para fixação com contra-ponta ou entre-centros, utilizando ferramenta acionada ou não-acionada.
14. *Features* concêntricas internas com diâmetro > 22 mm: representadas pelas *features* do tipo ID sendo tratadas da mesma maneira que as *features* do tipo OD, bem como as *features* de canal, reentrâncias, chanfros, filetes e arredondamentos. As operações de torneamento externo são associadas às operações de mandrilamento (torneamento interno) e as demais operações são tratadas de forma semelhante, tendo como diferencial restrições de deslocamento da ferramenta para acessar a parte interna da peça a fim de evitar interferência e choque do sistema de fixação da ferramenta com a peça. O diâmetro interno deve ser maior do que 22 mm para que estas operações possam ocorrer. As *features* de usinagem internas são semelhantes as *features* de usinagem externas (figura 7.17).
15. *Features* concêntricas internas com diâmetro ≤ 15 mm: apenas furação, utilizando ferramenta acionada ou não-acionada (figura 7.17).

7.2.2 Grafos E/Ou Para Operações de Acabamento e Semi-acabamento

A seguir são apresentados os grafos E/OU para as *features* concêntricas associadas às operações de acabamento e semi-acabamento. A partir destes grafos E/OU pode-se definir as alternativas de operações de acabamento e semi-acabamento para cada *feature* de torneamento mapeada.

1. Face: faceamento de acabamento.
2. Perfil para *features* OD e ID: mandrilamento ou torneamento com ferramenta de torneamento e mandrilamento esquerda, a partir do sobrematerial deixado nas operações de desbaste de *features* OD e ID cilíndricas; ou com ferramenta de torneamento/mandrilamento neutra (figura 7.18).
3. Perfil para fora (*cut-out*) e para dentro (*cut-in*) externo com reentrância: perfilamento/cópia com ferramenta de torneamento para esquerda e movimento para fora, a partir do sobrematerial deixado nas operações de desbaste de *features* de reentrâncias + perfilamento com ferramenta de torneamento para direita; ou perfilamento com ferramenta neutra; ou perfilamento com ferramenta de torneamento para esquerda com ângulo de abordagem compatível com a superfície para dentro; ou perfilamento com ferramenta circular de torneamento ou sangramento com diâmetro da ferramenta compatível (figura 7.19).

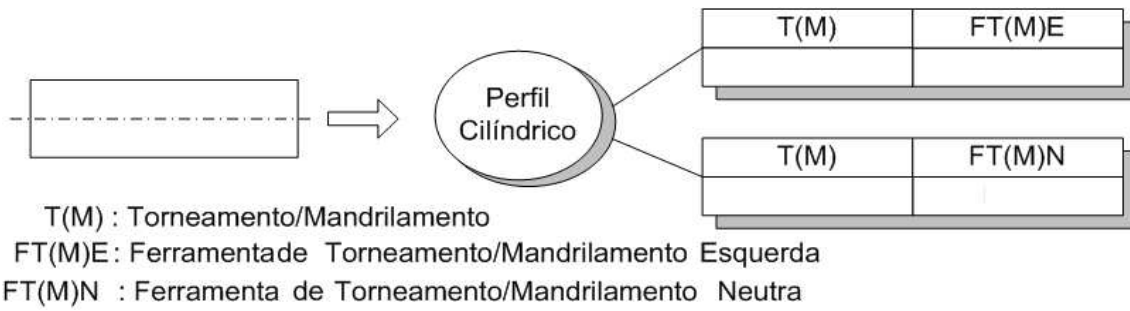


Figura 7.18: Grafo E/OU apresentando alternativas de usinagem para perfil com *features* OD e ID cilíndricas.

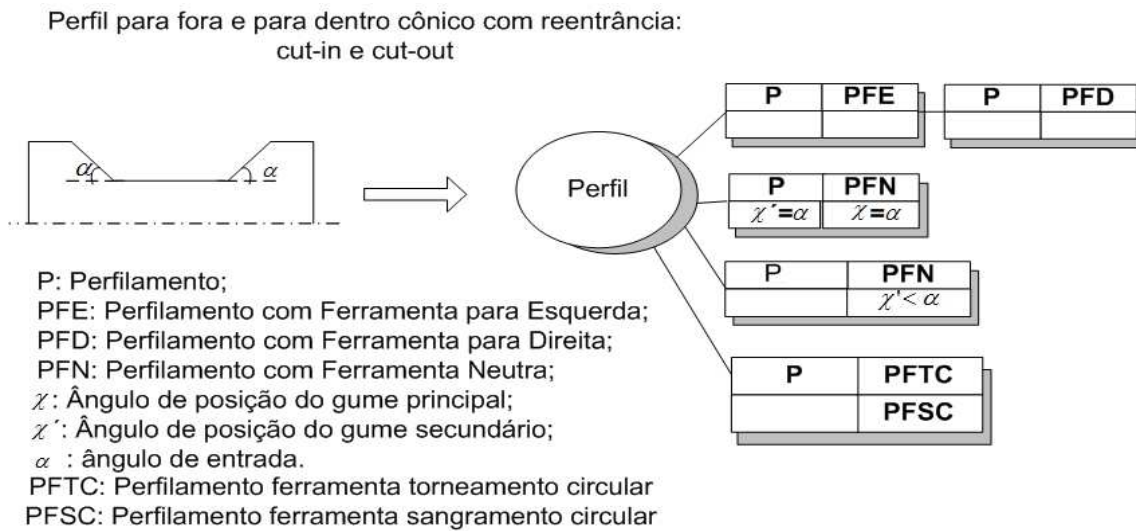


Figura 7.19: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo Perfil Externo para fora e para dentro com reentrância (*cut-in*).

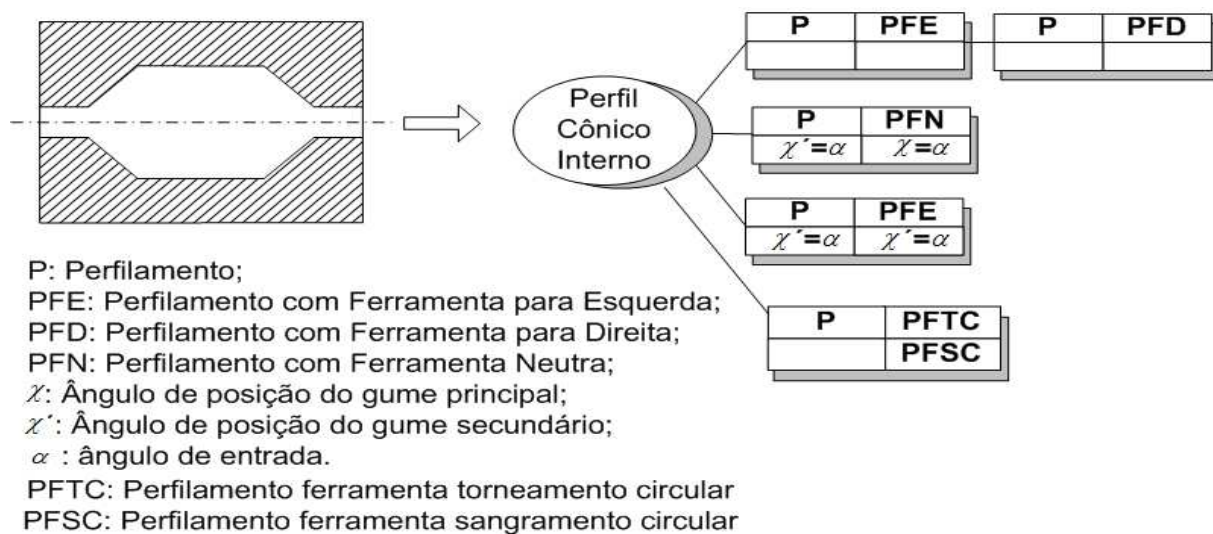


Figura 7.20: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo Perfil Interno para fora e para dentro com reentrância.

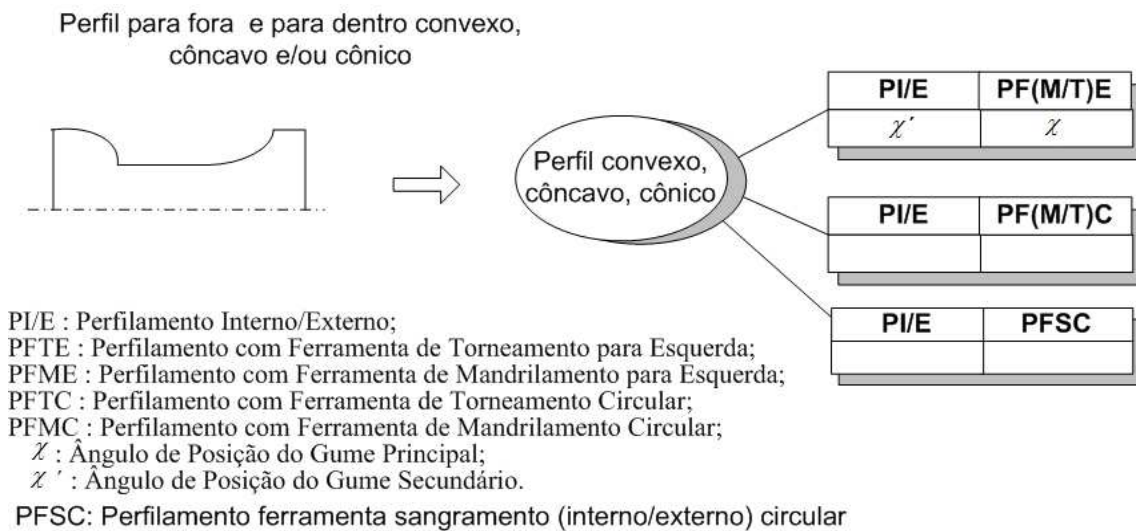


Figura 7.21: Grafo E/OU apresentando alternativas de usinagem para *feature* do tipo perfil para fora (*cut-out*) e para dentro (*cut-in*) com filetes para superfícies internas e externas.

4. Perfil para fora (*cut-out*) e para dentro (*cut-in*) interno com reentrância: perfilamento/cópia com ferramenta de mandrilamento para esquerda e movimento para fora, a partir do sobrematerial deixado nas operações de desbaste de *features* de reentrâncias internas + perfilamento com ferramenta de mandrilamento para direita; ou perfilamento com ferramenta neutra; ou perfilamento com ferramenta de mandrilamento; ou perfilamento com ferramenta circular de torneamento ou sangramento com diâmetro da ferramenta compatível (figura 7.20).
5. Perfil para fora e para dentro com filetes para superfícies internas e externas: perfilamento/cópia com ferramenta de torneamento/mandrilamento para esquerda com ângulo de abordagem adequado seguindo o perfil; ou ferramenta de torneamento/mandrilamento com geometria circular, para fora ou para dentro, a partir do sobrematerial deixado nas operações de desbaste de *features* de reentrâncias com filetes; ou perfilamento com ferramenta circular de torneamento ou sangramento com diâmetro da ferramenta compatível (figura 7.21).
6. Canal estreito: sangramento de acabamento.
7. Rosca: rosqueamento em uma superfície OD ou ID cilíndrica.
8. Recartilhado: recartilhamento em uma superfície OD.
9. Furo: alargamento de furos com qualquer diâmetro.

7.3 Representação do Plano de Operações Não-Linear por Features de Usinagem: *Workingsteps*

Faz-se a representação do plano de processos não-linear através de *Workingsteps*, incluindo as entidades *Non_sequential* e *Selective*, para cada *setup*, segundo ISO 14649 - Part 10 (STEP NC), re-

presentando assim as operações de usinagem com alternativas (grafo E/OU) associadas às *features* de usinagem, adotando como referência o roteamento padrão¹¹ estabelecido anteriormente.

A figura 7.22 apresenta um exemplo para uma peça, destacando a geometria da peça, *features* de usinagem e *setup* (a), a lista de *workingstep* (b), o grafo E/OU da seqüência de processo (c) e a estrutura de dados das operações de usinagem associadas às *features*, descrita através de uma lista (d).

Até o momento foi realizado apenas a representação do plano de processo não-linear através da estrutura de dados orientada a objetos definida pelo STEP-NC. A seguir será determinado a ordem na qual as *features* de usinagem deverão ser removidas da peça, ou seja, a ordem que os *workingsteps* deverão ser executados, que até o momento não foi definido, sem ainda linearizar o plano de processos não-linear.

7.4 Determinação da Seqüência de Operações: *Workplan*

Tendo-se determinado as *features* de usinagem (FU) associadas às operações de usinagem com alternativas, que representam na realidade uma entidade *workingstep* (STEP-NC), parte-se para a terceira atividade de planejamento de processo, que tem como objetivo a determinação da seqüência de operações de usinagem para fabricação da peça, gerando a seqüência na qual as “n” *features* serão usinadas, ou seja a ordem de execução dos *workingsteps*. Assim definimos o conceito de *workplan* (ISO 14649 - PART 1, 2003), que é uma coleção de *workingsteps* com uma seqüência de execução, ou seja, uma lista ordenada de executáveis, para um determinado *setup*.

Para cada *setup* têm-se os *workingstep*, associando *features* de usinagem (FU) e operações de usinagem, sendo representados por grafo de FU por *setup* (lista de *features*), tabela de relacionamento por *setup* e a matriz adjacente por *setup*, conforme apresentado na figura 7.23. Após a determinação da seqüência na qual as *features* de usinagem, associadas aos *workingsteps*, serão executados para cada *setup*, gera-se um *workplan* por *setup* da peça. Por exemplo, se o plano de processo tem dois *setups*, serão gerados três *workplans*:

- *Workplan* definindo os dois *setups*: identifica cada *setup* e qual será o primeiro *setup* a ser feito sobre a peça;
- *Workplan* para o primeiro *setup*: identifica a lista ordenada de *workingsteps* que serão executados na peça associados ao primeiro *setup* (*Setup1*);
- *Workplan* para o segundo *setup*: identifica a lista ordenada de *workingsteps* que serão executados na peça associados ao segundo *setup* (*Setup2*).

Utiliza-se do conceito de roteamento pré-definidos WANG e LI (1991) e HUANG (1986), onde tabelas ou árvores de decisão (figura 6.11) apresentam roteamentos típicos de usinagem para superfícies cilíndricas externas, internas e planas (Eixo C), baseados em “boas práticas de fabricação”.

¹¹Roteamento padrão define uma seqüência de operações de usinagem genéricas que poderão ser executadas sobre determinada *feature* de usinagem. Em função do tipo de *feature* de usinagem pode-se ter várias alternativas de operações de usinagem (operação, ferramenta, ciclo fixo e tecnologia de usinagem, por exemplo).

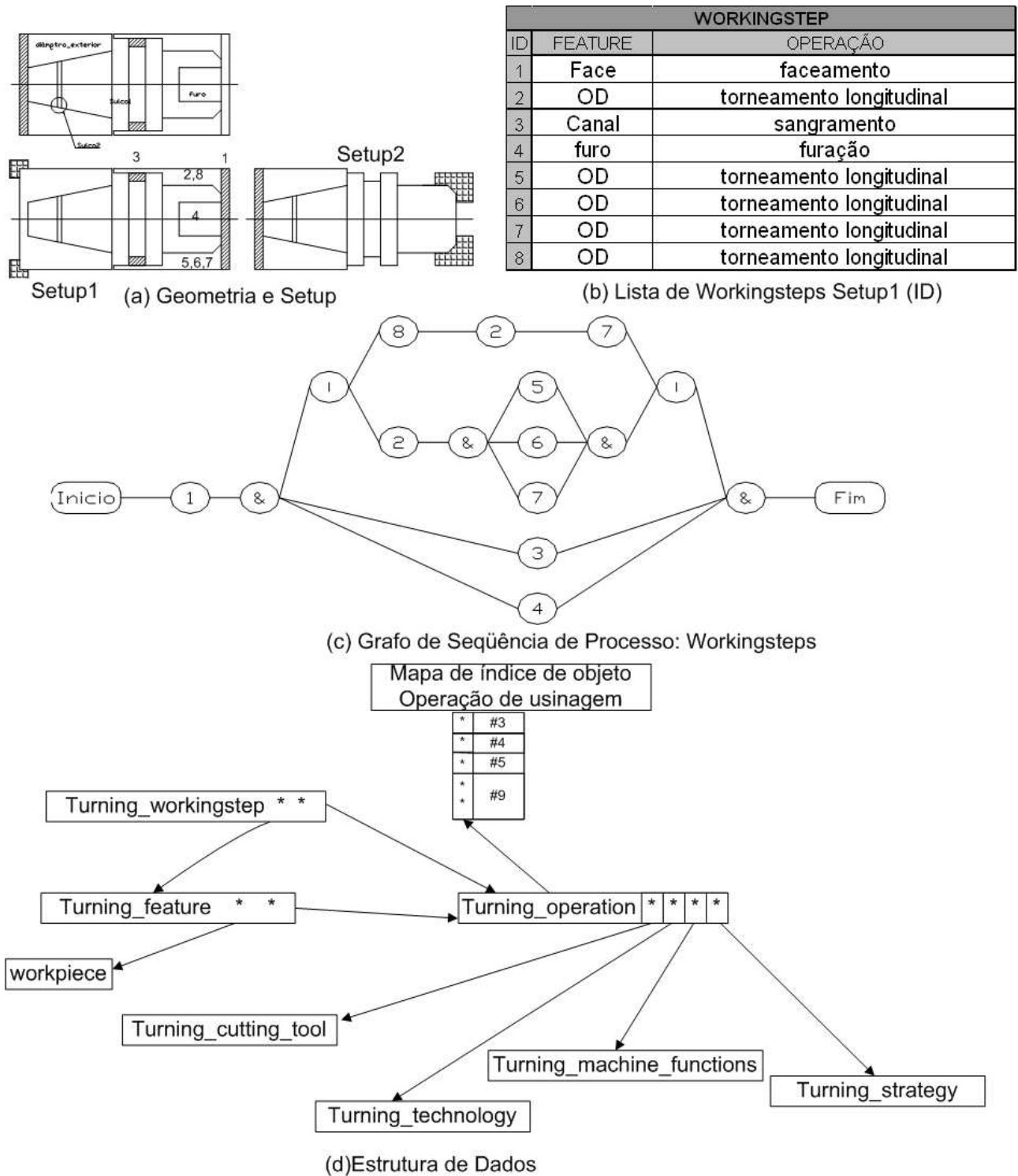
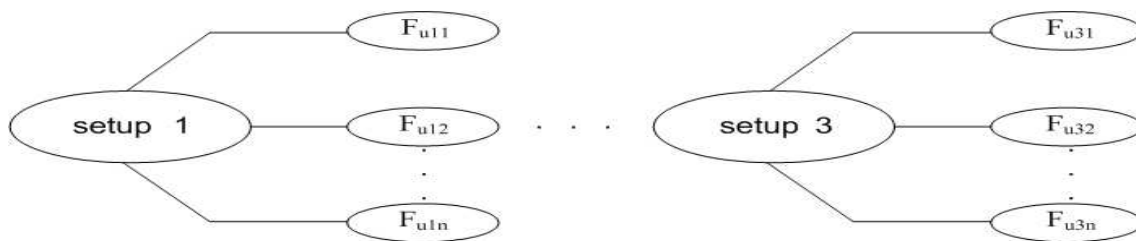


Figura 7.22: Exemplo de peça STEP-NC (2003): (a) Geometria da peça (b) lista de *workingstep*, (c) grafo E/OU, (d) lista da estrutura de dados para operações de usinagem.



Relação de Features de usinagem por setup para operações de desbaste, semi-acabamento e acabamento.

	Fu11	Fu12	...	FUn
	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
Fu11				
Fu12	*		*	*
Fu13	*			*
FUn		*		

Tabela de relacionamentos por setup

0	1	1	1
1	0	1	1
1	0	0	1
0	1	0	0

Matriz Adjacente por Setup

Figura 7.23: (a) Relação de *features* de usinagem por *setup* para operações de desbaste, semi-acabamento e acabamento (b) Tabela de relacionamentos por *setup* mostrando a precedência de *features* de usinagem (c) Matriz Adjacente por *setup*.

A determinação da seqüência de usinagem das *features* associadas às operações de usinagem, *workplan*, é baseada na análise de restrições que pode ser implementado através de regras de produção (CHANG *et al.*, 1998), (SUNDARAM, 1986) e (ÁLVARES, 2001), sem realizar ainda a otimização/linearização das alternativas de operações, que podem ser utilizadas na remoção da *feature* de usinagem.

Uma otimização eficiente do plano de processo não-linear não é possível neste estágio, pois não se tem ainda um plano de processo detalhado com as condições de usinagem, tempos e custos para cada operação. Até o momento pode-se utilizar como critério de otimização o número de *setups*, a continuidade de movimento e a perda de precedência. Entretanto não é possível utilizar critérios econômicos para definir uma função objetivo que otimize o plano de processo não-linear. Esta análise será feita mais a frente em um procedimento de otimização da seleção das operações alternativas, denominado de linearização do plano de processos com alternativas. As restrições que determinarão um gráfico ou tabela de precedência de *features* de usinagem são de quatro tipos (figura 7.23):

1. restrições de operações;
2. restrições geométricas (já analisadas);
3. restrições de ferramentas (já analisadas);
4. restrições de tolerâncias geométricas (já analisadas).

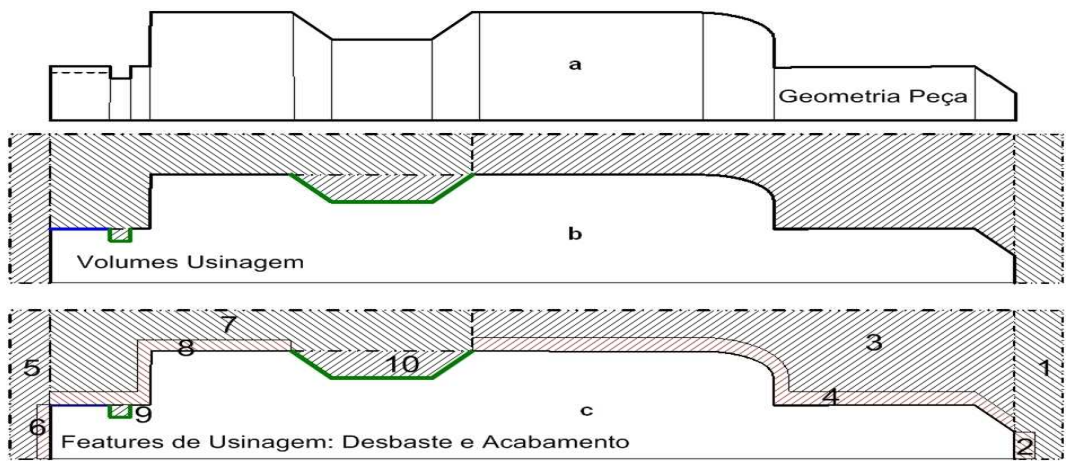
As restrições associadas às operações serão analisadas a seguir. Do ponto de vista de fabricação, boas práticas, existem várias regras que devem ser seguidas quando se vai determinar o seqüenciamento das operações, por exemplo:

- operações de desbaste devem ser executadas em primeiro lugar;
- uma *feature* primária deve ser executada antes que uma *feature* secundária associada à primária. Por exemplo, um chanfro deve ser usinado antes que uma rosca (secundária) em uma *feature* OD ou ID (primária) ou uma cavidade (fresamento) deve ser realizada antes que um furo em operações com Eixo C;
- determinar os elementos de localização, isto é, referências de apóio e fabricação. Por exemplo, as duas faces externas e furos de centro devem ser usinados em primeiro lugar;
- operações de furação concêntricas nas faces esquerda ou direita, se possível, devem ser executadas antes do faceamento externo;
- identificar as *features* que tem requisitos de tolerâncias de posicionamento, orientação, batimento ou orientação. Por exemplo as *features* de tolerâncias de concentricidade e perpendicularismo.

As regras para determinação da seqüência de usinagem associadas às rotas pré-estabelecidas de operações são apresentadas a seguir, e sintetizadas na figura 6.11, apresentada anteriormente:

1. se tem perfil interno remova a maior quantidade de material possível através de operações de furação¹²;
2. desbaste do perfil externo através de operações de faceamento, sangramento ou torneamento longitudinal;
3. execute as operações de sangramento externo;
4. execute operação de furação nas *features* que não podem ser usinadas por operações de mandrilamento ($D \leq 22$);
5. execute mandrilamento de desbaste no perfil interno;
6. execute as operações de sangramento interno;
7. acabamento dos elementos externos;
8. rosqueamento externo;
9. alargamento das *features* que não podem ser usinadas por operações de mandrilamento;
10. acabamento do perfil interno;
11. rosqueamento interno;

¹²Ferramenta não-acionada ($D \geq 22$) e Ferramenta acionada ($D \leq 15$).



$ID_{Workingstep}$	FU	Operação	Precedência	Tempo	Custo	Sequência Final
1	1	1	0			1 (início)
2	1	2	0			2
3	2	3	1			4
4	3	4	2			3
⋮	⋮	⋮	⋮			⋮
n	m	n				k (final)

Figura 7.24: Opções de *Workingsteps*: *features* de usinagem, operações de usinagem representados no formato de Matriz Operações Teóricas: operação, tempo e custo teórico da operação (TP), precedência (prioridade) e seqüência final de execução de *workingstep*.

- para eixo Eixo C deve-se seguir a seguinte seqüências de operações (ferramenta rotativa): fresamento -> furação -> escareamento -> alargamento -> rosqueamento.

A tabela de relacionamento de precedência apresenta o relacionamento de precedência de operações associadas às *features* de usinagem (FU) determinadas anteriormente (*workingstep*), levando em consideração as quatro restrições apresentadas (figura 7.23). As restrições entre *features* são determinadas e este relacionamento de precedência é representado por um Grafo Direcionado, através de uma estrutura de dados baseado em uma matriz adjacente X.

Para resolver o problema de sequenciamento de operações de usinagem é utilizado o algoritmo para determinação da matriz de acessibilidade R (DEO, 1974 & SUNDARAM, 1986).

Optou-se pela abordagem proposta por SUNDARAM (1986) por ser um algoritmo com um ótimo desempenho, quando comparado com a abordagem de algoritmos genéticos, que apresentaria um desempenho inferior e ainda seria mais complexo em termos de implementação (USHER e BOWDEN, 1996).

Este método estabelece um procedimento sistemático para seleção da seqüência de operações de usinagem, ou melhor da seqüência (ordem) de remoção de material associadas as *features* de usinagem. As operações elementares de usinagem podem ser agrupadas em uma seqüência por *setup* e o grupo de operações elementares deste *setup* (*workingstep*) pode ser processado em uma única fixação na máquina escolhida (*workplan*).

Pode-se representar o resultado desta atividade através de uma matriz de “n” *workingsteps*, relacionando “ m” *features* de usinagens e “n” operações de usinagem (alternativas, cardinalidade 1 => n)

para fabricar a *feature* de usinagem (planos alternativos), através de uma matriz semelhante à matriz das operações teóricas de Halevi (HALEVI, 1999), sendo “n” \geq “m”.

A figura 7.24 apresenta um exemplo de *workingsteps* representado através do formato de matriz das operações teóricas, sem levar em contas as restrições dos recursos de fabricação. As *features* de usinagem (fig. 7.24 (c)) são arranjadas em uma ordem de precedência, determinada anteriormente. A coluna prioridade/precedência indica a restrição na seqüência de *features* de usinagem, associadas aos *workingsteps*. Prioridade/precedência 0 (zero) significa que a operação pode ser a primeira ou pode ser executada em qualquer ordem. TP indica o tempo padrão teórico ou custo teórico para a operação, não calculados ainda. Restrições tecnológicas como tolerâncias geométricas, impõem a necessidade de que algumas operações sejam executadas sem remoção da peça da máquina ou da fixação, ou seja, mesmo *setup*, definindo a precedência. A coluna seqüência final representa a ordem de execução dos *workingsteps* após a linearização do grafo E/OU de precedências de *features*. A tabela representada na figura 7.24 é inspirada na abordagem da Matriz de Halevi (HALEVI, 1999).

Tem-se como resultado final a representação de todas as alternativas de operações para cada *feature* de usinagem. Ou melhor, uma relação de precedência de *workingsteps* na forma de um grafo de precedências de *features* linearizado, que pode ser mapeado em um grafo E/OU de *workingsteps*, indicando as alternativas de operações para geração de cada *feature* de usinagem, a ser representado por um *workplan* com alternativas.

7.5 Estratégias para Geração de Trajetórias de Ferramentas

Neste item serão apresentadas as estratégias de usinagem previstas pelo arcabouço conceitual de geração da movimentação de ferramenta para cada operação considerada. São utilizados três métodos para geração de movimentação de ferramenta:

- No primeiro método a movimentação de ferramenta para o *workingstep* é associada a ciclos fixos disponíveis no CNC Fanuc 18i-ta do centro de torneamento Galaxy 15M da Romi, utilizado na implementação do sistema. Sempre que possível faz-se a opção por movimentação de ferramenta baseada em ciclos fixos;
- No segundo método a movimentação de ferramenta para o *workingstep* considerado é gerada sem utilização de ciclos fixos;
- No terceiro método (híbrido) a movimentação de ferramenta é gerada com a combinação de ciclos fixos e programação convencional sem o uso de ciclos fixos.

A determinação da estratégia usada na geração de movimentação de ferramentas (percursos de ferramentas) para cada operação é baseada no método de raciocínio geométrico (SHAH e MANTYLA, 1995) sobre o perfil bidimensional da peça a ser usinada, e nas estratégias especificadas na ISO 14649 - Part 12 (2003). São utilizadas algumas definições associadas a ISO 14649 (STEP-NC), detalhadas a seguir:

- *Workingstep* (*Machining_workingstep* e *Turning_workingstep*): são informações de usinagem para uma ferramenta de corte atuar sobre uma *feature* de usinagem, contendo uma operação de usinagem;

- Operação de usinagem: dados tecnológicos para um *Workingstep* que detalha a operação, sendo composto de dados da ferramenta de corte, estratégias para geração de trajetórias de ferramentas para operações de torneamento, funções da máquina (fluido de corte, remoção de cavaco, etc), profundidade de corte, tolerância de acabamento, velocidade de corte, avanço, plano de retração, plano de segurança, estratégia de aproximação/abordagem e estratégia de retração.

Assim para o detalhamento de uma operação de usinagem é necessário a especificação da ferramenta e uma estratégia usada para criação da trajetória da ferramenta de usinagem. Segundo STEP-NC Parts 10 e 12 esta entidade é definida como estratégia de usinagem (*Machining_strategy*), sendo esta estratégia associada ao próprio ciclo fixo do fabricante, ou à uma estratégia particular, sem uso de ciclo fixo, ou de forma híbrida, para o *workingstep* considerado.

Uma estratégia de movimentação de ferramenta está associada basicamente à definição de quatro trajetórias da ferramenta que são:

- trajetória de aproximação/abordagem: trajetória desenvolvida pela ferramenta a partir do ponto de aproximação (plano de aproximação), para abordagem da peça a fim de iniciar o corte;
- trajetória de usinagem: trajetória desenvolvida pela ferramenta para usinagem da peça, após a execução da trajetória de aproximação para início de usinagem, onde há remoção de material;
- trajetória de afastamento/retração: trajetória desenvolvida pela ferramenta ao final de um passe de usinagem, para se afastar da peça posicionando-se no ponto de retração (plano de retração), que poderá ser o mesmo que o plano de aproximação;
- trajetória de transição da ferramenta: qualquer trajetória desenvolvida pela ferramenta para se posicionar no ponto de aproximação. Por exemplo, posicionamento no ponto de aproximação a partir do plano de segurança definido para a direção Z, posicionamento a partir do ponto de afastamento (plano de retração) até o próximo ponto de aproximação da ferramenta.

A figura 7.25 apresenta a natureza da lógica para geração do caminho de ferramenta para torneamento de desbaste em peças rotacionais simétricas (*features* concêntricas). Normalmente, mais de uma fixação da peça é necessária, consistindo de torneamento com ferramenta para direita (operações de corte de 1 a 3) e ferramenta para esquerda (operações 4 e 5). É necessário trabalhar com o conceito de geometrias intermediárias geradas durante as várias operações de usinagem, conforme já descrito. Operações de mergulho (*cut-in*) através de sangramento é uma boa alternativa para usinagem de reentrâncias que gerem superfícies intermediárias (fig. 7.25 operações 3 e 5).

Inicialmente é necessário analisar o modelo da peça bruta e da peça acabada representados na forma de segmentos cilíndricos e cônicos (figura 7.23) através da modelagem por *features*, conforme descrito nos métodos de decomposição orientado ao *setup*, geometria e operação, que identificaram as *features* de usinagem e torneamento e associaram operações de usinagem, definindo os *workingsteps*. O modelo de *features* concêntricas, não incluindo as *features* de eixo C, é convertido para um modelo baseado em arestas (figura 7.23b), segmentos de retas e arcos. Divide-se os volumes para operações de desbaste e acabamento. A geometria da operação e movimentação da ferramenta associada ao volume a ser removido, *features* de torneamento, é determinada da seguinte forma para o exemplo em questão (figura 7.23):

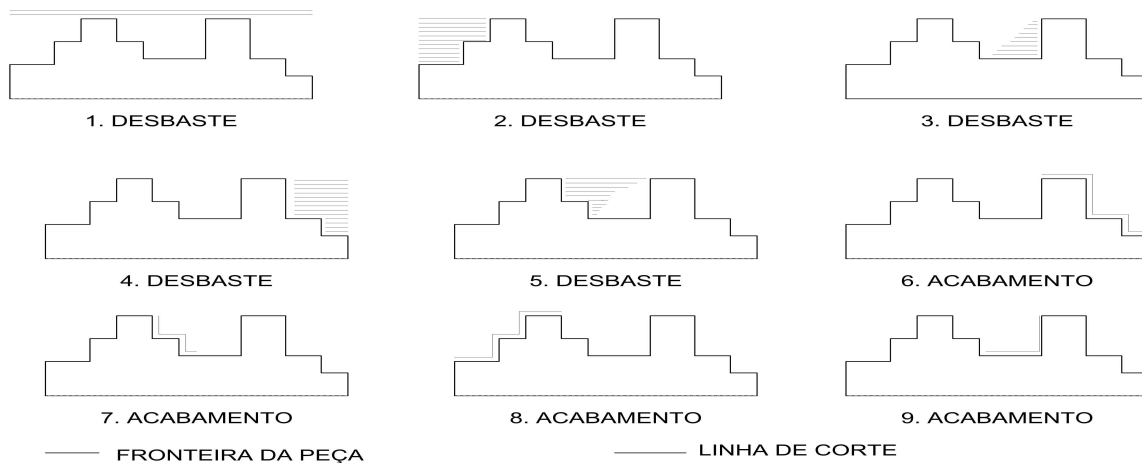


Figura 7.25: Geração lógica do caminho de ferramenta para torneamento (SHAH e MANTYLA, 1995) destacando estratégias para movimentação de ferramenta da esquerda para direita (1, 2, 3, 8 e 9) e da direita para esquerda (4 e 5) nas operações de desbaste e de acabamento (perfilamento).

1. Determinação do ponto de referência (Linha Delimitadora entre *setups*): a determinação do ponto de referência depende da estratégia de fixação da peça adotada e da forma geométrica da peça, que já foi determinado na atividade de mapeamento de *features*, tendo-se identificado as *features* de usinagem para cada *setup* (*workplan*). Como princípio deve-se evitar colisões com os elementos de fixação, deixando as áreas que podem ser usinadas de uma forma mais adequada, para as próximas operações;
2. Gerar a geometria do perfil desejado da direita para esquerda até o ponto de referência (*setup* 1): a figura 7.23(c) apresenta a primeira fixação da peça e a área a ser removida (volumes 1, 2, 3 e 4).
3. Copie as demais regiões que não foram usinadas, para o perfil da peça bruta à esquerda (*setup* 2): os volumes 5 a 10 (fig. 7.23(c)) representam a segunda fixação da peça (peça invertida);
4. Varrer a depressão, reentrância, do perfil meta de acordo com a capacidade do movimento da ferramenta (ângulo mergulho/abordagem, *cut-in*): o ângulo da linha (segmento de reta) da geometria intermediária é determinado pela geometria da ferramenta (ângulo de posição (*cut-out*) e ângulo de abordagem (*cut-in*)).

As estratégias de geração da trajetória da ferramenta de corte para cada operação de usinagem, desbaste e acabamento, associadas às *features* de usinagem previamente determinadas são baseadas na ISO 14649 - PART 12 (2003), sintetizadas no capítulo 4.

A seguir são apresentadas as estratégias previstas para cada tipo de operação considerada para remoção das *features* de usinagem, incluindo as estratégias definidas na ISO 14649 - PART 12 (2003). Algumas destas estratégias são implementadas através de ciclos fixos, parametrizados, o que facilita a geração do código; sendo utilizado, principalmente, os seguintes ciclos fixo do CNC Fanuc 18i-ta (ROMI, 2003):

1. G71 (tipos I e II) e G70 são utilizados para torneamento longitudinal automático de um perfil, sendo associado à uma lista de *features* de projeto, que são mapeadas como uma *Turn-*

- ing_Feature*. Assim G71 e G70 são associados a um *Turning_Workingstep* de desbaste e outro de acabamento, respectivamente (mapeamento $n \Rightarrow m$). O ciclo fixo G71 tipo II permite operações de usinagem em reentrâncias (*cut-in*) com um adequado ângulo de abordagem do porta-ferramenta. Já o tipo I não permite operações de mergulho, na direção da linha de centro da peça. Na usinagem de *splines* utiliza-se o ciclo fixo G71 tipo II;
2. G72 e G70 permitem o torneamento transversal automático de um perfil, sendo associado a um *groove complexo*, a partir de uma lista de *features* de projeto, e mapeados como uma *Composite_Feature*. Assim G72 e G70 são associados a um *Machining_Workingstep* de desbaste e outro de acabamento, respectivamente (mapeamento $n \Rightarrow 1$);
 3. G73 e G70 são utilizados para torneamento longitudinal automático de um perfil de uma peça fundida ou pré-usinada, com um determinado sobrematerial para permitir uma operação de perfilamento do *blank*, tendo uma estratégia de movimentação de ferramenta paralela ao perfil da peça, sendo associado à uma lista de *features* de projeto, que são mapeadas como uma *Turning_Feature*. Assim G73 e G70 são associados a um *Turning_Workingstep* de desbaste e outro de acabamento, respectivamente (mapeamento $n \Rightarrow m$);
 4. G75 permite a usinagem de canais, sendo associados as *features* de projeto do tipo *groove simples*¹³. Também é utilizado para usinagem de *features* de faces retilíneas e de recartilhados (*feature secundária*). Nos três casos, a *feature* de projeto (principal ou secundária) é mapeada como um *Machining_Workingstep* (mapeamento $1 \Rightarrow 1$);
 5. G74 é utilizado como ciclo de furação de ferramenta acionada ($D \leq 15$ mm) associado à uma *feature* de projeto de região interna ou de Eixo C mapeada como um *Machining_Workingstep* (mapeamento $1 \Rightarrow 1$);
 6. G83 é utilizado como ciclo de furação de ferramenta não-acionada ($D > 20$ mm) associado à uma *feature* de projeto de região interna mapeada como um *Machining_Workingstep* (mapeamento $1 \Rightarrow 1$);
 7. G76 (ciclo de rosqueamento automático) é utilizado em uma *feature secundária*, sendo mapeado como um *Machining_Workingstep* (mapeamento $1 \Rightarrow 1$).

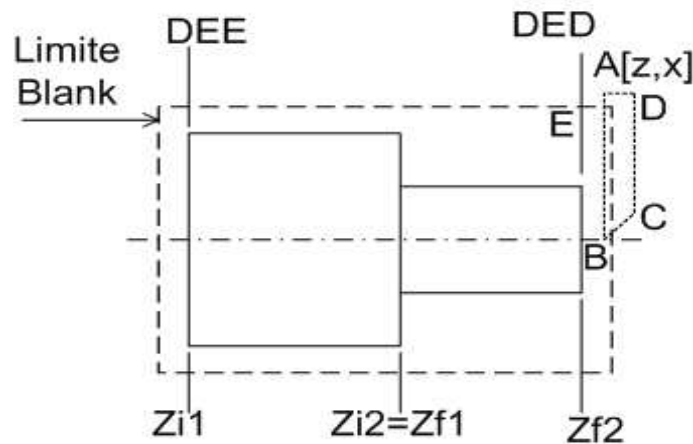
7.5.1 Faceamento: Desbaste

Esta operação de usinagem ocorre principalmente nas extremidades da peça (esquerda ou direita) em função do tipo de fixação da peça. O início da usinagem tem início no ponto A (Z,X), onde é posicionada a ponta da ferramenta de faceamento, conforme apresentado na figura associada ao algoritmo (1). O algoritmo (1) apresenta a estratégia classificada como unidirecional diagonal ou perpendicular. No caso da ilustração a estratégia é diagonal, pois a trajetória de afastamento apresenta um ângulo diferente de 90° , no caso 45° . Pode-se utilizar o ciclo fixo G75 do CNC Fanuc 18i-ta, que implementa esta operação de faceamento.

¹³*Grooves* simples são modelados diretamente da GUI e identificados na biblioteca de *features* com um *Groove* (mapeamento $1 \Rightarrow 1$).

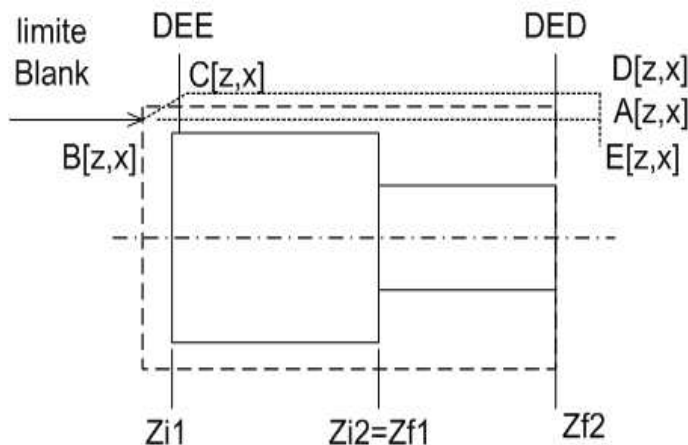
Algoritmo 1 Caminho de ferramenta para faceamento (torneamento transversal).

<p>Ponto A (Z,X): Início da usinagem $A(Z)$ = coordenada Z2 + sobrematerial para acabamento -ap $A(X)$ = coordenada X do blank (raio blank) + valor do plano de segurança de afastamento em X</p>
<p>Operação de Faceamento do ponto A para B Ferramenta desloca-se verticalmente para o ponto B, centro da peça, realizando o primeiro passe de faceamento</p>
<p>Deslocamento para o plano de segurança Z, do ponto B para C, Ferramenta afasta-se do centro da peça para o Ponto D situado um plano segurança X</p>
<p>Deslocamento no plano de segurança X, do ponto C para D Ferramenta afasta-se da extremidade esquerda da peça para o ponto D situado no plano de segurança Z $D(Z) = A(Z)$ e $D(X) = C(X)$</p>
<p>Deslocamento do Ponto D para o Ponto E (Z,X): Novo passe de usinagem $E(Z) = A(Z) + ap$ e $E(X) = B(X)$ = coordenada X do blank (raio) + valor do plano de segurança de afastamento em X do blank</p>
<p>Operação de faceamento do ponto E para F(Z)=B(Z). Ferramenta desloca-se verticalmente para o ponto B, centro da peça, realizando o segundo passe de faceamento</p>
<p>A seqüência usinagem é continuada até atingir a superfície da peça ou ter deixado um sobremetal para a operação de acabamento</p>



Algoritmo 2 Caminho de ferramenta para desbaste torneamento longitudinal da direita para esquerda.

<p>Ponto A (Z,X): Início da usinagem, deslocamento horizontal > deslocamento vertical (X=cte) $A(Z) = \text{coordenada } Z2 + \text{sobrematerial para acabamento} + \text{valor de segurança no plano } Z$ $A(X) = \text{coordenada } X \text{ do blank (raio blank)} - ap$</p>
<p>Operação de Torneamento do ponto A para B Ferramenta desloca-se horizontalmente para o ponto B, extremidade esquerda, realizando o primeiro passe de torneamento</p>
<p>Deslocamento para o plano de segurança Z, do ponto B para C, Ferramenta afasta-se da peça com deslocamento para a direita e para cima em 45°, atingindo o ponto C Ponto C está situado um plano X de segurança do blank sendo $C(X) = 2 * (\text{Diâmetro do blank}) - A(X)$</p>
<p>Deslocamento no plano de segurança X, do ponto C para D Ferramenta afasta-se da extremidade esquerda da peça para o ponto D situado no plano de segurança Z $D(Z) = A(Z)$ e $D(X) = C(X)$</p>
<p>Deslocamento do Ponto D para o Ponto E (Z,X): Novo passe de usinagem $E(X) = A(X) + ap$ $E(Z) = A(Z)$</p>
<p>A seqüência de usinagem é continuada até atingir o último passe de usinagem da peça, deixando um sobremetal para a operação de acabamento</p>
<p>Faz-se o mesmo para o próximo volume de usinagem a ser removido por torneamento com um novo comprimento de outra feature OD ou ID cilíndrica</p>



7.5.2 Torneamento Externo e Interno: Desbaste da Direita para Esquerda (Ferramenta para Esquerda)

A usinagem tem início da extremidade direita para esquerda, sendo que a remoção de material para um determinado passe(s) de usinagem leva em consideração o diâmetro nominal da peça mais um sobremetal de acabamento. O algoritmo (2) apresenta a estratégia classificada como unidirecional diagonal ou perpendicular. No caso da ilustração associada ao algoritmo (2) é diagonal, pois a trajetória de afastamento apresenta um ângulo de 45°. Pode-se utilizar o ciclo fixo G71 tipo I do CNC Fanuc 18i-ta, que implementa esta operação de torneamento longitudinal.

7.5.3 Torneamento Externo e Interno de Reentrância com Superfície Intermediária à Direita: Desbaste da Direita para Esquerda

No torneamento externo e interno de reentrância (canal largo) com superfície intermediária à direita, usinagem com sentido pra dentro da peça (ferramenta esquerda), a usinagem tem início da extremidade direita para esquerda, sendo que a remoção de material para um determinado passe(s) de usinagem leva em consideração o diâmetro nominal das *features* cilíndricas adjacentes à direita e à esquerda que definem o canal largo. O diâmetro externo da *feature* à esquerda é menor que o da *feature* à direita. Os passes de usinagem são executados de forma semelhante ao caso anterior, acrescentando-se também o sobremetal de acabamento. A diferença reside na consideração que se faz com relação ao ângulo de abordagem da ferramenta, que gera um triângulo para dentro da peça (superfície intermediária), devido à trajetória oblíqua da ferramenta (ângulo de abordagem). Para remover esta superfície triangular é necessário o uso de uma ferramenta para direita ou a usinagem da superfície, com a mesma ferramenta em outro *setup*, invertendo o lado de fixação da peça. Caso a *feature* direita seja um cone, deve-se escolher uma ferramenta com ângulo de abordagem igual ao ângulo do cone. O algoritmo (3) apresenta esta estratégia de movimentação de ferramenta e a ilustração associada exemplifica o caso.

A superfície intermediária é determinada a partir da equação da reta em função do ângulo de abordagem da ferramenta ($\alpha = \chi$), plano de segurança associado à coordenada X (ponto P(X)) e vértice da *feature* de forma à direita, no caso o ponto A (Z,X). O deslocamento de P para A, primeiro passe, que inclui o sobremetal para a operação de acabamento, é dado por:

$$\tan(\alpha) = \frac{\Delta X}{\Delta Z} = \frac{a_p - \text{sobremetal}}{Z_{maior} - Z_{menor}}$$

$$Z_{menor} = Z_{maior} - \frac{a_p - \text{sobremetal}}{\tan(\alpha)} \quad (7.3)$$

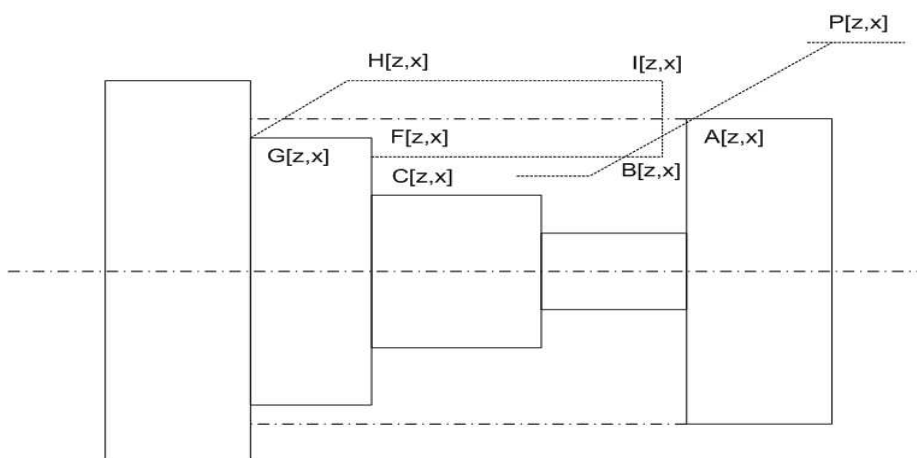
ou

$$A(Z) = P(Z) - \frac{(a_p - \text{sobremetal})}{\tan \alpha} \quad (7.4)$$

O deslocamento da ferramenta não deve ser tangente ao ponto A e sim paralelo à uma reta de inclinação α , que é obtida adicionando um valor de sobremetal na direção Z, no movimento de aproximação da ferramenta. A seguir calcula-se para cada profundidade de corte (a_p) o ponto de início de usinagem, substituindo a coordenada Z na equação da reta determinada. O deslocamento da ferramenta a partir do ponto P se dá através de um posicionamento bi-axial da ferramenta no plano ZX, iniciando o movimento de usinagem. A seguir executa-se o movimento de retração e de transição, para reposicionar a ferramenta para o movimento de aproximação e iniciar o novo passe de usinagem. Pode-se utilizar o ciclo fixo G71 tipo II do CNC Fanuc 18i-ta, para implementar este tipo de estratégia.

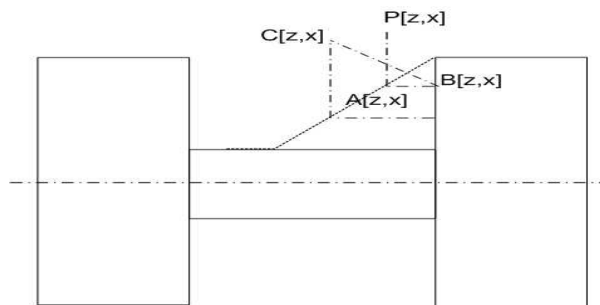
Algoritmo 3 Algoritmo para determinação do caminho de ferramenta para desbaste de reentrância em torneamento longitudinal da direita para esquerda.

<p>Analisa o ângulo de entrada/abordagem da ferramenta em relação ao ângulo da reentrância à direita: ferramenta apropriada</p> <p>Ponto A (Z,X) – Início da usinagem – deslocamento horizontal > deslocamento vertical (X=cte)</p> <p>$A(Z) = \text{coordenada } Z2 + \text{sobrematerial } ap$</p>
<p>Operação de Torneamento do ponto P para A</p> <p>Ferramenta desloca-se obliquamente para o ponto A, superfície intermediária, iniciando o primeiro passe de torneamento até a extremidade G da peça</p>
<p>Deslocamento para plano de segurança Z, do ponto G para H</p> <p>Ferramenta afasta-se da peça com deslocamento para a direita e para cima em 45°, atingindo o ponto H</p> <p>Ponto H está situado no plano X de segurança do blank sendo</p> <p>$H(X) = 2 * (\text{Diâmetro do blank}) - A(X)$</p>
<p>Deslocamento no plano de segurança X, do ponto H para I</p> <p>Ferramenta afasta-se da extremidade esquerda da peça para o ponto I situado no plano de segurança X</p> <p>$I(Z) = A(Z)$ $I(X) = H(X)$</p>
<p>Deslocamento do Ponto I para o Ponto B (Z,X), reta oblíqua: Novo passe de usinagem até C</p> <p>$B(X) = A(X) - ap$ $B(Z) = A(Z)$</p>
<p>A seqüência de usinagem é continuada até atingir o último passe de usinagem da peça, deixando um sobre-metal para a operação de acabamento</p>
<p>Faz-se o mesmo para o próximo volume de usinagem a ser removido por torneamento com um novo comprimento de outra feature OD ou ID cilíndrica</p>



Algoritmo 4 Algoritmo para determinação do caminho de ferramenta no torneamento, desbaste da esquerda para direita em uma superfície intermediária.

<p>Analisa o ângulo de entrada/abordagem (cut-in) da ferramenta em relação ao ângulo da reentrância à direita: ferramenta apropriada</p> <p>Ponto P (Z,X) = plano de segurança inicial = Diâmetro + afastamento de segurança da placa</p> <p>P(Z) = coordenada Z do diâmetro externo do</p>
<p>Deslocamento do ponto P (Z,X) ao ponto B (Z,X)</p> <p>Ferramenta desloca-se verticalmente para o ponto B, que intercepta a reta oblíqua</p>
<p>Operação de torneamento da esquerda para a direita, movimentação em Z</p> <p>Ferramenta aproxima-se da peça com deslocamento para a esquerda, até atingir a cota nominal Z do eixo à esquerda + sobrematerial para operação de acabamento (ap)</p>
<p>Deslocamento para o ponto C(Z,X) afastando-se da peça, em trajetória com direção de 45° da reta oblíqua</p>
<p>A seqüência de usinagem é continuada até atingir o último passe de usinagem da peça, removendo a superfície intermediária, em forma triangular, deixando um sobremetal para a operação de acabamento</p>



7.5.4 Torneamento Externo e Interno de Reentrância (Canal Largo) com Superfície Intermediária Triangular à Direita: Desbaste Esquerda para Direita

A usinagem tem início da esquerda para direita (ferramenta para direita) após a operação sobre a reentrância da direita para esquerda ou de uma *feature* cônica para dentro (*cut-in*) da peça usinada por uma ferramenta com ângulo de abordagem não compatível com a inclinação do cone. Deve-se remover esta superfície intermediária na forma triangular (2D), sendo que a remoção de material para um determinado passe(s) de usinagem deve levar em consideração o diâmetro nominal da peça mais o sobremetal de acabamento.

Pode-se utilizar o mesmo procedimento, na geração da trajetória de ferramenta, para usinar esta superfície intermediária em um segundo *setup*, invertendo o lado de fixação da peça, e utilizando uma ferramenta para esquerda, conforme descrito no item anterior.

Faz-se o torneamento a partir do ponto P (plano de aproximação inicial) fazendo com que a fer-

ramenta atinja o ponto correspondente sobre a reta gerada pela operação intermediária (movimento de aproximação) e a seguir inicia-se o torneamento até o ponto B, deixando o sobremetal para a operação de acabamento. A seguir desloca-se a ferramenta para o ponto C, afastando da peça, movimento de retrocesso, e faz-se o posicionamento sobre o ponto seguinte sobre o reta inclinada, movimento de aproximação. Esta movimentação é feita até a remoção completa da superfície intermediária, removendo-se a região triangular (algoritmo 4). Pode-se utilizar o ciclo fixo G71 tipo I do CNC Fanuc 18i-ta.

7.5.5 Torneamento Externo e Interno de Reentrância (Canal Largo) com Superfície Cônica à Esquerda: Desbaste Direita para Esquerda

Pode-se utilizar duas abordagens de movimentação de ferramenta: cortes paralelo e bi-axial. A usinagem tem início da direita para esquerda, sendo que a remoção de material para um determinado passe(s) de usinagem leva em consideração o perfil cônico nominal da peça mais um sobremetal de acabamento.

No corte paralelo a ferramenta se desloca para cada passe de usinagem, em uma trajetória horizontal com uma cota X constante. A superfície gerada após os “n” passes será uma superfície oblíqua com “n” degraus, necessitando portanto de uma operação final de perfilamento com um passe de acabamento e/ou de semi-acabamento. A geometria do corte é apresentada na figura 7.26 (esquerda). Os pontos de intersecção à reta inclinada (Z,X) para uma determinada cota X (associada a profundidade de corte utilizada) são determinados através da igualdade das equações das retas consideradas, por exemplo:

$$\begin{aligned} X &= X_i \text{ (profundidade de corte } a_p=d) \\ X &= X_0 - (Z - Z_0) \tan(\alpha) \\ Z &= \frac{(X_0 + Z_0 \tan \alpha) - X_i}{\tan(\alpha)} \text{ (ponto de intersecção)} \end{aligned}$$

Assim ficam determinadas as cotas (Z,X) associadas ao ponto de intersecção com a reta inclinada.

No corte bi-axial a ferramenta se desloca de forma oblíqua com deslocamento linear simultâneo em X e Z seguindo uma trajetória paralela à superfície cônica da reentrância. Determina-se os pontos inicial e final da trajetória, através das várias equações das retas paralelas. Inicialmente determina-se o número de passes de usinagem (n) necessários em função da geometria da reta inclinada, dado por (7.5).

$$n = \frac{(h1 - h2)}{a_p} \cos(\alpha) \quad (7.5)$$

Os pontos inicial e final, já incluindo o valor de sobrematerial para a operação de acabamento nas cotas de X e Z da reta oblíqua, são determinados para cada passe por:

$$\begin{aligned} X_{passe\ final} &= X_0 \\ X_{passe\ inicial} &= X_{passe\ intermediario} = X_0 - Passe_{numero} * a_p \end{aligned}$$

onde $Passe_{numero}$ está associado à seqüência de passes de 1 a N, na ordem decrescente, sendo o primeiro passe igual a “n” e o enésimo passe igual a 1.

$$\begin{aligned} Z_{passe\ inicial} &= Z_0 \\ Z_{passe\ final} &= Z_{passe\ intermediario} = Z_0 - (Passe_{numero} * a_p * \tan(\alpha)) \end{aligned}$$

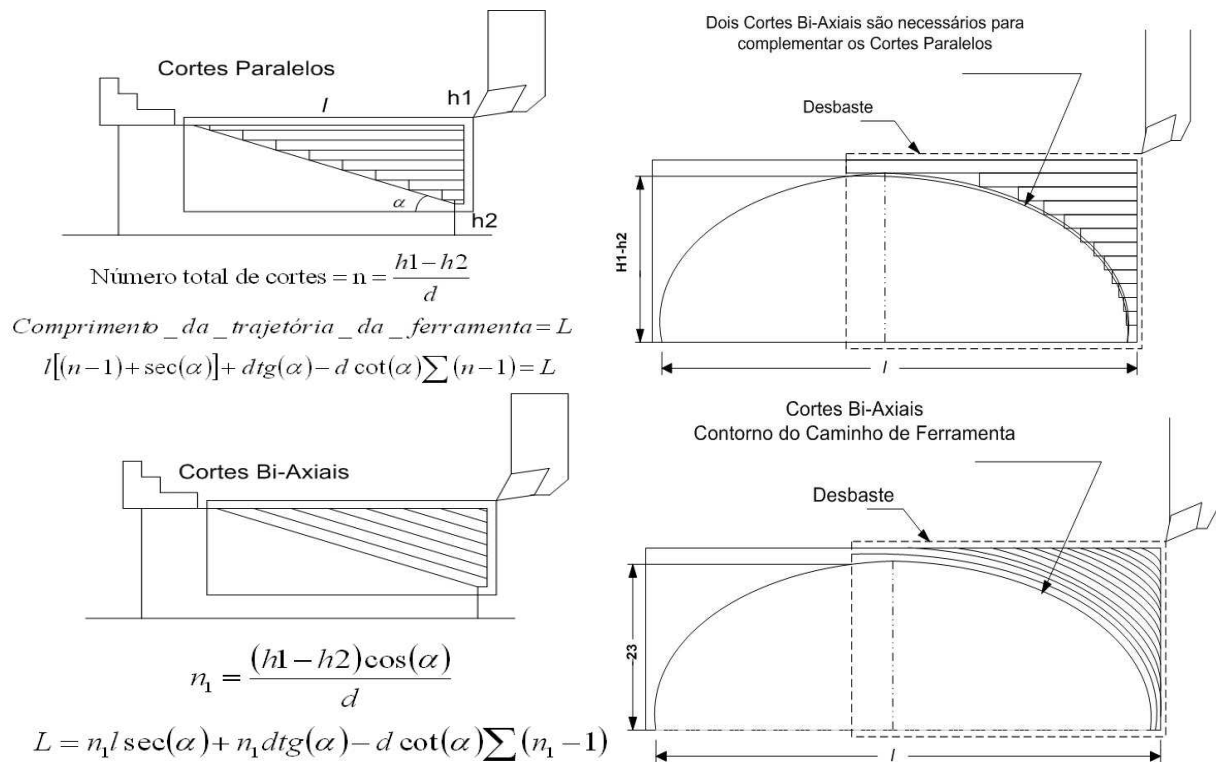


Figura 7.26: Trajetória de ferramenta para perfil cônico e convexo: corte paralelo x corte bi-axial.

Assim os pontos inicial e final são determinados para cada passe de usinagem, permitindo a interpolação linear nos dois eixos do CNC do ponto inicial ao ponto final. Pode-se utilizar o ciclo fixo G71 tipo II do CNC Fanuc 18i-ta.

7.5.6 Torneamento Externo e Interno de Reentrância (Canal Largo) com Superfície Circular (Côncava ou Convexa) à Esquerda: Desbaste Direita para Esquerda

Pode-se utilizar duas abordagens de movimentação de ferramenta: cortes paralelo e bi-axial (fig. 7.26 direita). A usinagem tem início da direita para esquerda, sendo que a remoção de material para um determinado passe(s) de usinagem leva em consideração o perfil circular nominal da peça mais um sobremetal de acabamento. O caso é semelhante ao anterior.

No corte paralelo a ferramenta se desloca para cada passe de usinagem em uma trajetória horizontal com uma cota X constante. A superfície gerada após os “n” passes será uma superfície “circular” com “n” degraus, necessitando portanto de uma operação final de perfilamento com um passe de semi-acabamento e/ou de acabamento. Os pontos de intersecção (Z,X) ao círculo, para uma determinada cota X (associada à profundidade de corte utilizada) são determinados através da igualdade das equações da reta e do círculo consideradas:

$$X = X_1 \text{ (a seqüência de “n” profundidades de corte determinando cada equação de X)}$$

$$(X - X_0)^2 + (Z - Z_0)^2 = R^2$$

Ponto de intersecção para $X = X_1$:

$$(X_1 - X_0)^2 + (Z - Z_0)^2 = R^2$$

$$Z^2 - 2 * Z_0 Z + (X_1 - X_0)^2 + Z_0^2 - R^2 = 0 \quad (7.6)$$

Assim resolvendo esta equação do segundo grau determina-se as duas raízes, sendo utilizada a raiz compatível com o quadrante que se está trabalhando. Neste caso, trata-se do primeiro quadrante onde o valor da cota Z está associada. Desta forma ficam determinados as cotas (Z,X) associadas ao ponto de intersecção com o arco.

No corte bi-axial a ferramenta se desloca de forma circular com deslocamento linear simultâneo em X e Z seguindo uma trajetória paralela à superfície circular da reentrância. Determina-se os pontos inicial e final da trajetória, através das várias equações de círculos concêntricos. Inicialmente determina-se o número de passes de usinagem (n) necessários, em função da geometria dos círculos concêntricos, sendo do menor raio “ R ”, até o maior raio que ocorre no vértice esquerdo do volume a ser removido, “ R_{total} ”.

$$R_{total} = \sqrt{Z_0^2 + X_0^2} \quad (7.7)$$

$$n = \frac{R_{total} - R}{a_{radial}} \quad (7.8)$$

Os pontos inicial e final, já incluindo nas cotas (Z,X) e no raio o valor de sobrematerial para a operação de acabamento, serão determinados para cada passe por:

$$X_{passe\ final} = X_0$$

$$X_{passe\ inicial} = X_{passe\ intermediario} = X_0 - X_{R_{passe\ numero}},$$

onde $X_{R_{passe\ numero}}$ está associado a cota X para o raio considerado e que intercepta a reta associada a cota $X_{R_{passe\ numero}}$ determinada pela equação do círculo, conhecendo-se o raio e a cota Z_0 (material a ser removido, já levando em conta o sobrematerial para acabamento), na seqüência de passes de 1 a “ N ”, na ordem decrescente.

$$Z_{passe\ inicial} = Z_0$$

$$Z_{passe\ final} = Z_{passe\ intermediario} = Z_0 - Z_{R_{passe\ numero}},$$

onde $Z_{R_{passe\ numero}}$ está associado à cota Z para o raio considerado e que intercepta a reta associada a cota $Z_{R_{passe\ numero}}$ determinada pela equação do círculo, conhecendo-se o raio e a cota X_0 (material a ser removido, já levando em conta o sobrematerial para acabamento), na seqüência de passes de 1 a “ n ”, na ordem decrescente.

Assim os pontos inicial e final são determinados para cada passe de usinagem permitindo a interpolação circular em sentido horário ou anti-horário (função do quadrante que o arco se encontra) nos dois eixos do CNC, do ponto inicial ao ponto final. Pode-se utilizar o ciclo fixo G71 tipo II do CNC Fanuc 18i-ta.

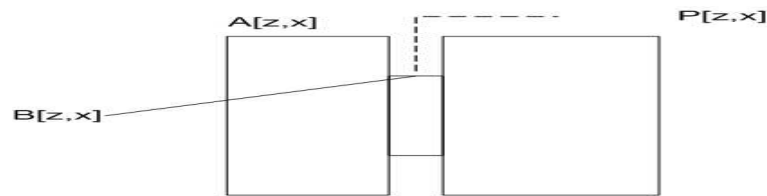


Figura 7.27: Caminho de ferramenta para desbaste de ranhura.

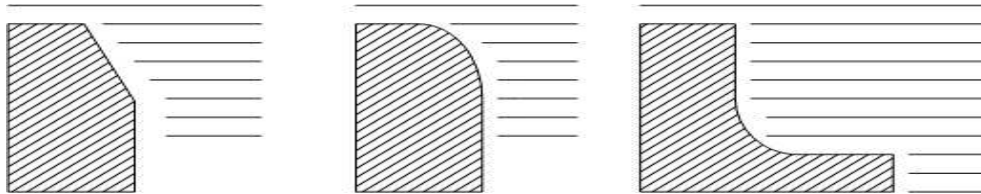


Figura 7.28: Caminho da ferramenta para desbaste de *features* de transição: chanfro, arredondamento e filete, respectivamente.

7.5.7 Torneamento Externo e Interno de Ranhuras: Desbaste Canal Estreito

A usinagem ocorre em função da geometria da ranhura, sendo que o movimento da ferramenta tem direção radial, perpendicular ao eixo rotacional da peça. A operação utiliza uma ferramenta de sangramento, podendo ocorrer um ou vários passes na direção longitudinal, em função da largura do canal. Deve-se prever um sobrematerial para acabamento.

A figura 7.27 apresenta a trajetória da ferramenta que inicia no ponto $P(Z,X)$ até o ponto $A(Z,X)$ em avanço rápido, movimentação de transição, e à uma distância de segurança da peça (cota X_0 , plano de retração). A ferramenta é posicionada em $A(Z)$ de forma adequada para geração do canal. A seguir a ferramenta move-se até o ponto $B(X)$ em avanço programado, movimento de aproximação seguido de movimento de usinagem, e depois retorna para o ponto $A(X)$, em avanço rápido (movimento de retração). Se for necessário um novo passe de usinagem na direção longitudinal, repete-se a mesma movimentação da ferramenta em avanço rápido e depois em avanço programado, gerando mais uma seção do canal, até completá-lo na sua largura. Pode-se utilizar o ciclo fixo G75, G72 ou G71 tipo I do CNC Fanuc 18i-ta.

7.5.8 Torneamento Externo e Interno de *Features* de Transição: Desbaste Chanfros, Arredondamentos e Filetes: Usinagem Direita para Esquerda

A usinagem ocorre em função da geometria da *features* de transição, podendo ser do tipo chanfro (superfície cônica) ou arredondamento e filetes (superfícies convexas e côncavas). Utilizam-se abordagens semelhantes à usinagem de reentrâncias com movimentação da ferramenta da direita para esquerda, conforme descrito anteriormente.

A figura 7.28 apresenta a trajetória da ferramenta da direita para esquerda. Pode-se ter a mesma movimentação com ferramenta da esquerda para direita, utilizando-se o mesmo procedimento. Deve-se prever um sobrematerial para a operação de acabamento.

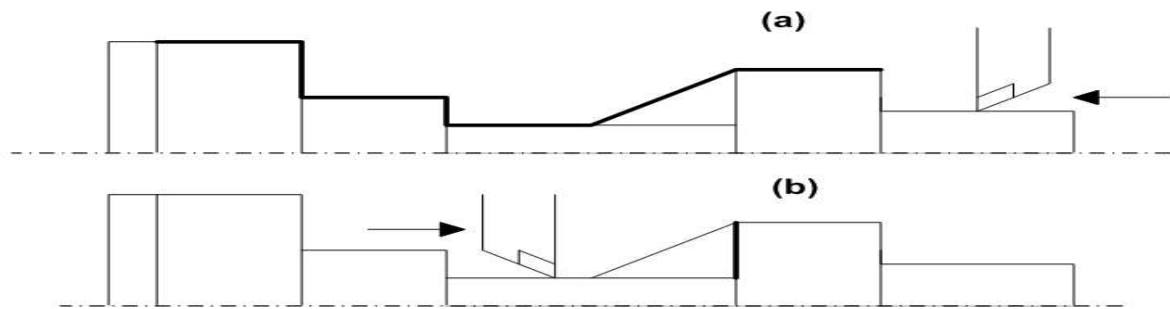


Figura 7.29: Caminho da ferramenta para acabamento: operação de perfilamento (movimento bi-axial) com ferramenta para esquerda e torneamento com ferramenta para direita (direção do avanço).

7.5.9 Operações de Acabamento: Perfilamento Externo e Interno de *Features* e Outras Operações

O perfilamento/cópia consiste em operações de acabamento e/ou semi-acabamento ao longo do perfil da peça, para atingir a qualidade dimensional e geométrica especificada em projeto. As demais operações de acabamento consistem no passe final sobre uma *feature* de usinagem que sofreu uma operação preliminar de desbaste.

As operações de acabamento da geometria do perfil da peça estão associadas ao sobrematerial deixado pela operação de desbaste. A diferença geométrica das operações de acabamento com relação ao desbaste concentra-se no fato da usinagem ocorrer ao longo do perfil, em um passe de usinagem (acabamento) ou dois passes (semi-acabamento e acabamento). São modeladas as seguintes operações: faceamento, torneamento/mandrilamento da direita para esquerda e da esquerda para direita, perfilamento/cópia, sangramento, rosqueamento, alargamento, corte da peça, fresamento e recartilhamento.

Pode-se utilizar movimentação da ferramenta da direita para esquerda, bem como, da esquerda para a direita, sendo abordagens de usinagem semelhantes às descritas nas operações de desbaste.

A operação final de corte da peça, normalmente utilizada com alimentação por barras em um único *setup* da peça, é uma operação semelhante à operação de sangramento, tendo por objetivo cortar a peça, deixando-a com a dimensão especificada em projeto, sendo a última operação realizada sobre a peça. Após esta operação a peça é removida da máquina.

A figura 7.29 apresenta a trajetória da ferramenta da direita para esquerda (ferramenta lado direito) e da esquerda para direita (ferramenta lado esquerdo) em uma operação de perfilamento (movimento bi-axial) com ferramenta para esquerda (direção do avanço) e torneamento com ferramenta para direita (direção do avanço). Pode-se ter a mesma movimentação com ferramenta da esquerda para direita, utilizando-se o mesmo procedimento. Deve-se prever um sobrematerial para a operação de acabamento. Pode-se utilizar o ciclo fixo G71 tipos I e II associado à G70 do CNC Fanuc 18i-ta, que implementam a operação de torneamento longitudinal sem mergulho (*cut-out*, G71 tipo I) e com a possibilidade de mergulho (*cut-in*, G71 tipo II), respectivamente, para executar as operações de desbaste (G71) e acabamento (G70).

7.5.10 Operações de Fresamento e Furação: Eixo C

A movimentação de ferramentas rotativas associada à Eixo C podem ocorrer em duas direções:

- Axial (paralelo ao eixo Z);
- Radial (paralelo ao eixo X).

As operações na direção axial ocorrem principalmente nas faces esquerda ou direita da peça (extremidade da peça), podendo-se ter operações de furação, escareamento, alargamento, rosqueamento e fresamento, utilizando-se um sistema de coordenadas cartesiano ou um sistema de coordenadas polar (X, C). O sistema de coordenadas polar permite uma movimentação de ferramenta nos eixos X e C com interpolação linear e circular destes eixos, possibilitando a geração de trajetórias de ferramentas para usinagem de canais, cavidades e ilhas, por exemplo.

As operações na direção radial ocorrem na face cilíndrica da peça, podendo-se ter operações de furação, escareamento, alargamento, rosqueamento e fresamento, utilizando-se um sistema de coordenadas cartesiano ou um sistema de coordenadas cilíndrica (Z, C). O sistema de coordenadas cilíndrica permite uma movimentação de ferramenta nos eixos Z e C com interpolação linear e circular destes eixos, permitindo a geração de trajetórias de ferramentas para usinagem de canais, cavidades e ilhas.

A movimentação de ferramentas de fresamento é realizada sem utilização de ciclos fixos, pois os mesmos não estão disponíveis no CNC. Para operações de furação e correlatas (alargamento e escareamento) e rosqueamento com macho rígido, são utilizados ciclos fixos disponibilizados pelo CNC. A estratégia de movimentação de ferramentas de fresamento é baseado em ISO 14649 - PART 11 (2003).

7.6 Determinação das Ferramentas de Corte (Insertos e Suportes)

A seleção dos insertos/pastilhas de metal duro e porta-ferramentas (ferramentas de corte¹⁴) é baseada em dois critérios básicos, sendo adotado a terminologia Sandvik Coromant (SANDVIK, 2003):

- no material a ser usinado, podendo-se selecionar ferramentas de diversas classes Coromant (compatível com as classes ISO P, M ou K);
- no tipo de operação de usinagem associada às *features* de usinagem, incluindo a estratégia de movimentação da ferramenta considerada. Por exemplo, se a reentrância, operação de mergulho, tem um ângulo de inclinação de 55° com a direção Z, o porta-ferramenta deverá ter um ângulo de abordagem, *cut-in*, maior ou igual, para permitir a usinagem desta geometria.

Para cada *Machining_workingstep* e *Turning_workingstep*, associados às *features* de usinagem, são determinados todos os insertos e porta-ferramentas que poderão ser utilizados em função:

- da operação (torneamento, perfilamento/cópia, faceamento, mandrilamento, recartilhamento, rosqueamento, sangramento, furação, fresamento, escareamento e alargamento);

¹⁴A associação entre um inserto e um porta-ferramenta será denominado de ferramenta de corte. As opções de ferramentas de corte para operações com Eixo C são menores no sistema considerado, quando comparadas com as opções disponíveis para ferramentas de torneamento e correlatas.

- do tipo de operação, se é de desbaste, semi-acabamento ou acabamento;
- estratégia de movimentação da ferramenta considerada;
- do tipo de ferramenta (para esquerda, para direita, neutra ou circular);
- do ângulo de abordagem da ferramenta (ângulos de posição efetivo do gume principal e do gume secundário, DIN 6581, e ângulo de entrada, segundo ASA B5.22 (1950) que define os ângulos de trabalho) associado à geometria do porta-ferramenta, fixação da ferramenta no dispositivo de fixação de ferramentas (*cut-in* e *cut-out*).

Os dados associados às pastilhas de corte e porta-ferramentas são armazenados em uma base de dados relacional. Utiliza-se a linguagem SQL para realizar a pesquisa das ferramentas de corte a serem selecionadas para a usinagem de cada *feature* de usinagem que compõe a peça.

Para ferramentas rotativas, seleciona-se as alternativas de ferramentas em função da classe de material a ser usinada e dos diâmetros necessários para as operações de fresamento, furação, escareamento e rosqueamento. Durante a modelagem da peça por síntese de *features* subtrativas, escolhe-se o diâmetro da ferramenta que será utilizada a partir de uma lista de ferramentas que a GUI disponibiliza para o usuário. Esta lista de ferramentas define as ferramentas disponíveis no banco de dados do WebCAPP. Caso o usuário deseje, pode-se definir uma ferramenta com geometria diferente da especificada pelo sistema WebCAPP, o que dificultará a usinagem da peça, pois a mesma não estará disponível na base de dados de ferramentas.

Para ferramentas de torneamento, o formato da pastilha é selecionado em função do ângulo de posição necessário e das exigências de acessibilidade e versatilidade da ferramenta. Deve-se sempre que possível selecionar o maior ângulo de ponta, levando em conta a resistência e a economia, quando o fator acessibilidade (*cut-int* e *cut-out*, por exemplo) não for preponderante. Deve-se também selecionar o maior raio de arredondamento da ponta possível, para obter uma aresta de corte robusta e maiores avanços. Caso haja tendência à vibração opta-se por um raio menor.

A seleção inicia-se com a determinação dos portas-ferramenta que são compatíveis com a operação considerada e as pastilhas associadas. O sistema de fixação do inserto é então selecionado, levando em conta a movimentação da ferramenta a fim de evitar qualquer colisão com a peça, analisando o ângulo de entrada (abordagem, *cut-in*) do porta-ferramenta. Um porta-ferramenta com maior ângulo de entrada é preferível, caso não se tenha um porta-ferramenta com ângulo de entrada adequado à *feature* de usinagem. A seguir, as dimensões do porta-ferramenta são determinados (comprimento, largura e altura). Se existirem vários candidatos escolhe-se o porta-ferramenta mais largo e com menor comprimento, melhorando assim a rigidez da ferramenta.

Monta-se uma tabela relacionando as *features* de usinagem às alternativas de ferramentas a serem utilizadas em cada operação. Determina-se a geometria necessária para a ferramenta associada ao porta-ferramenta (ângulos, sentido(s) de avanço e restrições dimensionais). Através dos dados dos materiais do par peça-pastilha (segundo AISI/SAE e da classe da pastilha Coromant), faz-se uma pesquisa na base de dados de ferramentas, determinando-se as ferramentas disponíveis ou os conjuntos pastilhas/porta-ferramenta mais adequados, que poderão ser utilizados na fabricação da peça.

Dá-se preferência para aquelas ferramentas que já estejam montadas na torre de ferramentas da máquina, minimizando o tempo de *setup* de ferramenta e máquina, através de um fator de penali-

dade para as demais ferramentas. Procura-se especificar, sempre que possível, ferramenta esquerda (L), com movimento de avanço da direita para esquerda¹⁵ (mão-direita), levando em conta a necessidade de *setups* da peça, a fim de se evitar a utilização de ferramenta direita (R), da esquerda para direita (mão-esquerda). Novamente usa-se o critério da penalidade, agora para ferramenta direita (R). Pode-se gerar uma superfície intermediária (estratégia de usinagem adotada pelo *workingstep*) em um primeiro *setup*, e no segundo *setup* usina-se a superfície intermediária com a mesma ferramenta do primeiro *setup* (ferramenta esquerda), sem necessidade do uso de uma ferramenta direita. No *Setup1*, primeiro *setup*, leva-se também em consideração o custo unitário da ferramenta, a disponibilidade da ferramenta no almoxarifado e a qualidade desejada para a peça.

As condições de usinagem a serem utilizadas são obtidas diretamente da base de dados de ferramentas da Sandvik Coromant (Sandvik Coromant <http://www.coroguide.com/>), através de uma consulta a esta base de dados via formulário HTML, contendo os parâmetros necessários para obter estas informações. Optou-se por esta estratégia por não se dispor dos modelos matemáticos, constantes e coeficientes utilizados pela Sandvik nas equações de Taylor ou Kienzle. Consultando-se esta base de dados, para um único passe de usinagem, pode-se calcular as condições de corte otimizadas (velocidade de corte e avanço associadas à uma determinada profundidade de corte) para vida de ferramenta de 15 a 60 minutos. A partir destes dados determina-se o menor custo de usinagem e o menor tempo de usinagem. Têm-se assim as condições de usinagem para determinada profundidade de corte otimizadas segundo os três critérios: menor tempo, menor custo e vida de ferramenta pré-definida.

Após a seleção prévia das ferramentas (porta-ferramentas e insertos) utiliza-se os seguintes critérios para a seleção final dos insertos e do porta-ferramenta (Sandvik, 2003) relacionados à sua geometria e as condições de usinagem¹⁶, que serão utilizadas nas várias operações de usinagem com alternativas associadas às *features* de usinagem:

1. Torneamento desbaste de OD, ID, *splines* e reentrâncias (canal largo): escolhe-se a ferramenta que apresentar maior ângulo de ponta (ε), maior raio de arredondamento da ponta (r_ε), maior espessura de pastilha, maior comprimento do gume de corte e ângulo de posição (κ) próximo de 90° (minimizar força de corte). Se for uma *feature* ID deve-se levar em consideração o diâmetro mínimo para execução da operação de mandrilamento ($D \geq 22$ mm). O Porta-ferramenta é selecionado em função dos ângulos de posição e de abordagem (*cut-in* e *cut-out*), direção de avanço (L ou R) e operação interna ou externa. Determina-se para cada ferramenta selecionada as faixas de utilização de velocidade de corte, avanço, profundidade de usinagem (depende do comprimento efetivo de corte), geometria da ferramenta (ε , κ , r_ε , espessura da pastilha e comprimento do gume de corte) e diâmetro mínimo para operações de torneamento interno (mandrilamento).
2. Torneamento acabamento de OD e ID cilíndricas, *splines* e perfilamento interno e externo: escolhe-se a ferramenta que apresentar menor ângulo de ponta, maior raio de arredondamento,

¹⁵O centro de torneamento Galaxy Romi tem torre de ferramentas traseira. Assim as ferramentas de torneamento devem ser vistas e especificadas de forma invertida.

¹⁶As faixas de utilização de velocidade de corte, avanço, profundidade de usinagem, geometria da ferramenta, entre outros, são utilizados para determinação otimizada das condições de corte, segundo os critérios menor tempo de produção, menor custo de produção e vida de ferramenta.

menor espessura e menor comprimento efetivo da aresta de corte. Se for uma *feature* ID deve-se levar em consideração o diâmetro mínimo para execução da operação de mandrilamento. O Porta-ferramenta é selecionado em função dos ângulos de posição e de abordagem (*cut-in* e *cut-out*), direção de avanço e operação interna ou externa. O ângulo de posição deve permitir o contorno de todo perfil associado à geometria final da peça na região da *feature* de usinagem, onde ocorrerá a operação de corte. Determina-se para cada ferramenta selecionada as faixas de utilização de velocidade de corte, avanço, profundidade de usinagem, geometria da ferramenta (ϵ , κ , r_ϵ , espessura da pastilha e comprimento do gume de corte) e diâmetro mínimo para operações de torneamento interno (mandrilamento).

3. Sangramento: escolhe-se a ferramenta e o porta-ferramenta associados à geometria da *feature* de usinagem (interna ou externa) que tenha a maior largura possível, possibilitando a usinagem do canal. Determina-se para cada ferramenta selecionada as faixas de utilização de velocidade de corte, avanço, geometria da ferramenta (largura e comprimento da pastilha).
4. Rosqueamento: escolhe-se a ferramenta em função do tipo de rosca de uso mais geral (métrica, unificada/withworth, quadrada ou Acme/trapezoidal) e o passo da rosca (milímetro ou filetes por polegada). A seleção do porta-ferramenta leva em conta se a operação é interna ou externa. Determina-se para cada ferramenta selecionada o passo de rosca e o número de passes para usinagem da rosca e as faixas de utilização de velocidade de corte e avanço.
5. Furação: escolhe-se a ferramenta em função do diâmetro e do comprimento do furo. O dispositivo de fixação da ferramenta já está associado à ferramenta escolhida. Determina-se para cada ferramenta selecionada as faixas de utilização de velocidade de corte, avanço, profundidade de usinagem e as características da broca (comprimento e diâmetro).
6. Recartilhamento: escolhe-se a ferramenta em função da geometria do recartilhado desejado. Determina-se para cada ferramenta selecionada as faixas de utilização de velocidade de corte, profundidade de corte e geometria da ferramenta (largura do recartilhado).
7. Alargamento: escolhe-se a ferramenta de alargamento em função do diâmetro e do comprimento do furo associado e da qualidade especificada. O dispositivo de fixação da ferramenta já está associado à ferramenta escolhida. Determina-se para cada ferramenta selecionada as faixas de utilização de velocidade de corte, avanço, profundidade de usinagem e as características do alargador (comprimento e diâmetro).
8. Rosqueamento com macho rígido: escolhe-se a ferramenta em função do tipo de rosca e o passo da rosca (milímetro ou filetes por polegada) e busca-se as condições de usinagem na base de dados.
9. Fresamento: escolhe-se a ferramenta em função do diâmetro definido durante o procedimento de modelagem por *features* da peça. A partir deste diâmetro são selecionadas as opções de fresas que poderão ser utilizadas. Em função do tipo do tipo de *feature* de usinagem associada ao diâmetro da ferramenta escolhida para a operação (faceamento, fresamento de canais de 90° , fresamento de perfis ou fresamento de canais), busca-se as opções disponíveis na base de dados.

10. Escareamento: escolhe-se a ferramenta em função do diâmetro definido durante o procedimento de modelagem por *features* da peça. A partir deste diâmetro são selecionadas as opções de escareadores que poderão ser utilizados e busca-se as condições de usinagem na base de dados.

7.7 Modelo de Tempos Para Determinação das Condições de Corte Otimizadas Para Operações de Usinagem

De uma maneira geral os trabalhos associados à otimização das condições de corte no domínio de peças rotacionais simétricas sempre apresentam:

- uma formulação do problema a ser otimizado: normalmente restrito e não integrado a um sistema CAPP;
- um modelo matemático relativo ao tempo de corte para as operações de usinagem consideradas (abordagem multi-passe): normalmente voltado para operações de torneamento longitudinal e passes de acabamento com a mesma profundidade de corte. O modelo multi-passe leva em consideração as operações de desbaste e acabamento simultaneamente. As operações de desbaste e acabamento são executadas, normalmente, com a mesma ferramenta;
- uma função objetivo relacionada ao modelo de otimização da usinagem: pode ser baseada em custo de usinagem, vida de ferramenta ou taxa de produção, tendo por objetivo a minimização da função associada ao custo de usinagem ou a maximização da função associada à taxa de produção. Normalmente baseado em custo;
- um método de resolução do problema de otimização: resolução conjunta das operações de desbaste e de acabamento. Nos trabalhos mais recentes a otimização é baseada em *simulated annealing* e algoritmos genéticos.

O método proposto se diferencia dos métodos apresentados na revisão de literatura (capítulo 3) nos seguintes aspectos:

- formulação do problema a ser otimizado: as trajetórias da(s) ferramenta(s) consistem em múltiplos passes de usinagem e um passe de acabamento, onde os passes de desbaste (multi-passe) são utilizados para deixar um sobrematerial para a operação de acabamento (passe simples), que é executada ao longo do perfil da peça. São otimizadas as condições de corte para operações de torneamento/mandrilamento longitudinal linear, longitudinal cônico e longitudinal circular, faceamento, corte, sangramento, furação e recartilhamento; englobando todas as *features* concêntricas de projeto descritas na taxonomia de *features* (CAM-I, 1986) e ISO 14649 (2003), acrescido de *spline*, que não é considerado normalmente.
- modelo matemático relativo ao tempo de corte para as operações de usinagem consideradas (abordagem multi-passe): modelo matemático de tempos para operações multi-passes com diferentes profundidades de corte para torneamento/mandrilamento longitudinal linear, longitudinal cônico e longitudinal circular, faceamento, perfilamento, corte, sangramento, furação e

recartilhamento. O modelo multi-passe leva em consideração as operações de desbaste e acabamento simultaneamente, sendo o sobrematerial de acabamento uma restrição para o desbaste. A profundidade de corte de desbaste é igual à profundidade de corte total menos a profundidade de corte de acabamento. Cada *feature* de usinagem poderá ter vários modelos matemáticos em função das alternativas disponíveis no grafo e/ou das *features* de usinagem considerada (*workingstep* do *workplan*), sendo que as operações de acabamento e desbaste poderão ser executadas com ferramentas diferentes. A maioria dos métodos relatados na literatura trabalham apenas com modelo matemático relativo a operações de torneamento longitudinal retilíneo (SHIN e JOO, 1992 e REDDY *et al.*, 1988) e muito poucos com outras operações de usinagem (CHEN e SU, 1998).

- função objetivo: cada modelo de tempo associado à uma determinada operação de usinagem tem uma função objetivo. Assim têm-se várias funções objetivo para cada alternativa de usinagem considerada, associada a cada uma das *features* de usinagem da peça (*workingstep*). O problema de otimização torna-se muito mais complexo, pois são otimizadas todas as operações possíveis, levando em conta as alternativas disponíveis. Para cada operação de usinagem tratada define-se duas funções objetivo para otimização, sendo uma baseada em menor custo de produção e a segunda baseada na maior taxa de produção, determinando assim o intervalo de máxima eficiência de usinagem. Pode-se trabalhar também com vida de ferramenta fixa, quando custo e tempo não forem priorizados.
- método de resolução do problema de otimização: resolução do problema de forma conjunta para as operações de desbaste e de acabamento, baseado em algoritmos genéticos. A diferença do método concebido em relação aos propostos por REDDY *et al.* (1988), ONWUBOLU e KUMALO (2001) e CHEN e CHEN (2003), concentra-se no fato do algoritmo genético ser aplicado para operações multi-passe com diferentes profundidades de corte em torneamento e mandrilamento (longitudinal linear, longitudinal cônico e longitudinal circular), faceamento, perfilamento, corte, sangramento, furação e recartilhamento e de estar integrado a um sistema CAPP, bem como trabalhar com operações alternativas.

O método proposto baseia-se, principalmente, nos trabalhos de:

- SHIN e JOO (1992): quando trata de restrições;
- REDDY *et al.* (1988): quando trata de algoritmos genéticos como método de resolução do problema de otimização baseado em custo;
- CHEN e SU (1998) e de SPEEDDING *et al.* (1998): quando tratam do modelo de usinagem para outras operações de usinagem voltadas para peças rotacionais e baseados em modelos de custo.

Neste item são determinados os modelos de tempos de usinagem e no item seguinte é apresentado o modelo de otimização de usinagem para menor custo de produção, menor tempo de produção e vida de ferramenta fixa. Operações associadas a Eixo C não são modeladas, com exceção das operações de furação e correlatas. Operações como recartilhamento e rosqueamento só apresentam alternativas com relação à ferramentas.

7.7.1 Formulação do Problema

A seleção de condições de corte para modelo multi-passe de usinagem envolve a determinação de quatro parâmetros, que são: profundidade, avanço e velocidade de corte, bem como o número de passes de usinagem. Um modelo multi-passe generalizado (TAN e CREESE, 1995 & CHANG *et al.*, 1998) é utilizado para seleção do número de passes de usinagem ótimo, bem como, da velocidade de corte, avanço e profundidade de corte para cada passe individual. Ao contrário de um modelo de passe simples, onde a profundidade de corte é fixa e apenas um único passe é utilizado para produzir a peça, no modelo multi-passe são assumidos vários passes de usinagem para produzir a peça. Assim, a profundidade de corte (a_p) torna-se uma variável de controle. Qualquer modelo multi-passe pode ser gerado a partir de um modelo de passe único.

A otimização dos parâmetros de usinagem é realizada pela abordagem de algoritmos genéticos para as seguintes operações de usinagem:

- torneamento/mandrilamento longitudinal linear (modelo multi-passe);
- torneamento longitudinal cônico e longitudinal circular (modelo multi-passe);
- perfilamento (modelo multi-passe);
- faceamento¹⁷ (modelo multi-passe);
- corte da peça (modelo passe único);
- sangramento (modelo passe único);
- furação (modelo passe único);
- recartilhamento (modelo passe único).

7.7.2 Modelos Matemáticos para Cálculo de Tempos de Usinagem

Peças rotacionais usinadas em um torno CNC apresentam formas contínuas, sendo que os processos de torneamento incluem interpolação linear e/ou interpolação circular. A interpolação linear pode ser dividida em três operações: torneamento longitudinal retilíneo, faceamento e torneamento cônico. Interpolação circular é usada em torneamento circular gerando formas circulares, convexas ou côncavas. Para otimização deste perfil contínuo não é possível utilizar os modelos de usinagem propostos apenas para operações de torneamento longitudinal. Assim será utilizado um modelo mais genérico que possa ser aplicado para torneamento/mandrilamento longitudinal linear, torneamento longitudinal cônico e longitudinal circular, perfilamento e faceamento, em uma única fórmula de cálculo de tempo, englobando todas estas operações. É contemplada também a *feature* do tipo *spline*, que é tratada a partir de interpolação linear com resolução de 0,1 mm.

A figura 7.30 apresenta a trajetória de ferramenta para operações de desbaste e acabamento (CHEN e SU, 1998). A trajetória da ferramenta para desbaste é constituída por n passes de usinagem, sendo que o desbaste é dividido em dois estágios:

¹⁷Deve ser entendido como uma operação de torneamento transversal.

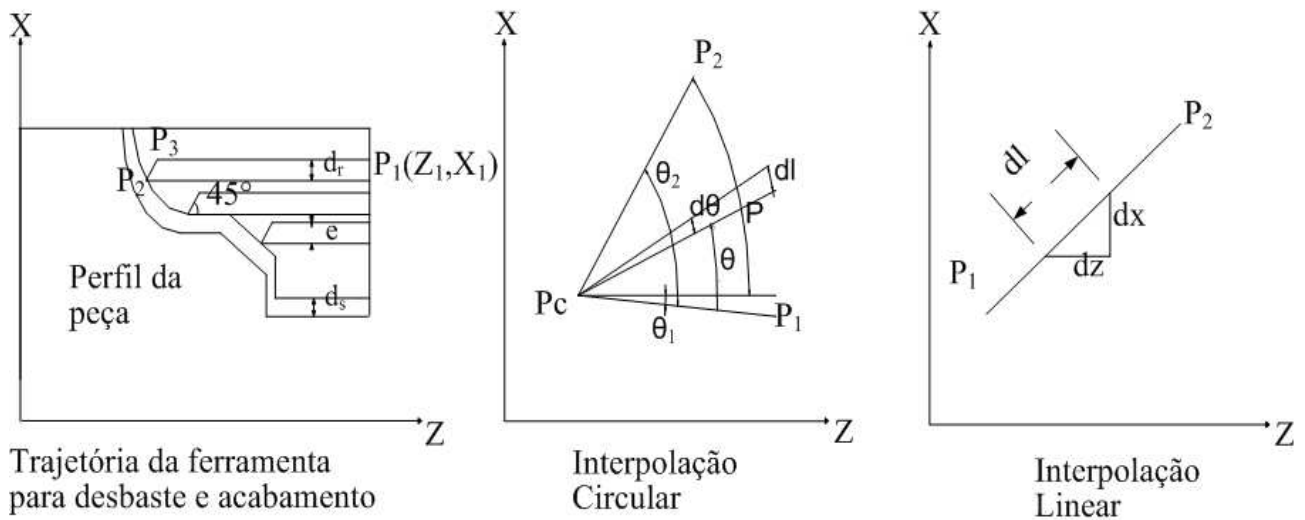


Figura 7.30: Modelo de usinagem multi-passe (CHEN e SU, 1998): (a) trajetória da ferramenta para dois estágios de desbaste e um passe de acabamento (b) Modelo interpolação circular (c) Modelo interpolação linear.

1. desbaste dos primeiros $(n-1)$ passes tendo uma trajetória de ferramenta para operação de torneamento longitudinal iniciando no ponto P_1 (com profundidade de corte de desbaste d_r , na direção X) até o ponto P_2 , distante d_s do perfil da peça (profundidade de corte de acabamento nas direções Z e X). Depois a ferramenta se afasta do ponto P_2 em direção ao ponto P_3 . O processo continua para o próximo passe de desbaste, conforme já apresentado na análise de estratégias de movimentação de ferramentas;
2. desbaste do último passe de usinagem (n) ao longo do perfil da peça, deixando um sobrematerial de acabamento no valor da profundidade de corte de acabamento (d_s), rea-lizando uma operação de perfilamento de semi-acabamento interno ou externo;

A seguir conclui-se a usinagem com um único passe de acabamento ao longo do perfil da peça, realizando uma operação de perfilamento de acabamento interno ou externo, podendo utilizar a mesma ferramenta de desbaste ou uma específica de acabamento. Este tipo de estratégia de movimentação é utilizado em ciclos fixos, com G71 e G70 (torneamento longitudinal) ou G72 e G70 (torneamento transversal, sangramento).

Os modelos matemáticos para as operações de usinagem modeladas (torneamento longitudinal, torneamento transversal (faceamento), sangramento, corte, recartilhamento, furação, alargamento e rosqueamento) são apresentados no Apêndice C. Estes modelos são utilizados para determinação das condições de usinagem otimizadas usando algoritmos genéticos.

7.7.3 Metodologia de Otimização

O procedimento de otimização é baseado em algoritmos genéticos, utilizando a abordagem de minimização proposta por YELOWLEY e GUN (1989), denominada de “ótima subdivisão” da profundidade de corte, que posteriormente foi aprimorada por GUPTA *et al.* (1995) e BHASKARA *et al.* (1998). Nesta abordagem a usinagem multi-passe é considerada como um problema multi-estágio,

como ocorre no procedimento de programação dinâmica. O problema de minimização para os passes de desbaste e acabamento é dividido em dois estágios. A minimização da função objetivo (custo ou tempo) é dividida em dois problemas de minimização separados para operações de desbaste e acabamento, visando a otimização da profundidade de corte para os “n” passes de desbaste e o passe de acabamento.

No primeiro estágio um conjunto de profundidades de corte é gerado, entre os limites máximos e mínimos da profundidade de corte para operações de desbaste e acabamento. A seguir calcula-se de forma separada, para as operações de desbaste e acabamento, o custo e/ou tempo mínimo e as velocidades de corte e avanços otimizados para cada profundidade de corte considerada, utilizando as funções objetivo apresentadas anteriormente para modelos de tempo ou de custo de produção, e as restrições associadas aos modelos.

Em função da resolução da profundidade de corte selecionada (décimo, centésimo ou milésimo de mm) as alternativas de profundidades de corte poderão aumentar de forma geométrica, o que inviabiliza a utilização de abordagens de otimização baseadas em programação dinâmica. Por exemplo, para uma faixa de profundidades de corte entre 1 e 5 mm, o cálculo das condições de corte otimizadas para cada passe de usinagem possível com resolução de 0,1 mm (profundidades de corte 1,0; 1,1; 1,2; ... 4,9 e 5,0) em um PC Pentium III leva dois segundos (BHASKARA *et al.*, 1998); então o cálculo de todos os passes (51) levaria 102 segundos. Caso a resolução fosse de 0,01 mm o cálculo de todos os passes (501) levaria 1.002 segundos, ou seja 16,7 minutos. Caso a resolução fosse de 0,001 mm o cálculo de todos os passes (5001) levaria 10.002 segundos, ou seja 2,9 horas. Desta forma deve haver um compromisso entre a resolução de passe escolhida e o tempo de resolução do problema. Os valores otimizados individualmente são armazenados e disponibilizados para serem usados no segundo estágio.

No segundo estágio, uma combinação de um pré-determinado subconjunto de profundidades de corte é selecionado, levando em conta a profundidade de corte total a ser removida da peça bruta. Calcula-se, inicialmente, um conjunto de número de passes de usinagem de desbaste (tentativas), em função da profundidade de corte total, profundidade de corte para os passes de acabamento e a máxima profundidade de corte permitida para operações de desbaste:

$$n_{(i)} = \frac{a_p - a_{ps(i)}}{a_{prU}} \quad (7.9)$$

A seguir utiliza-se uma abordagem de otimização baseada em algoritmos genéticos para determinar o número ótimo de passes de desbaste, a profundidade de corte para cada passe de desbaste e a profundidade de corte para o passe de acabamento, requeridos para obter o mínimo custo e/ou tempo de produção total. Antes da apresentação do algoritmo genético será detalhado cada estágio.

Primeiro Estágio Custos e tempos para passes individuais de desbaste e acabamento e suas velocidades de corte e avanços ótimas são determinados para valores fixos de profundidade de corte de acordo com a resolução determinada. No caso serão utilizados incrementos de 0,1 mm para operações de desbaste e acabamento. Para operações de acabamento e desbaste usa-se como referência as recomendações do fabricante de ferramentas (SANDVIK, 2003) para seleção da faixa de avanços a ser utilizado. De uma forma geral a profundidade de corte para desbaste varia de 1,0 a 10,0 mm (101

opções de passes) e para acabamento de 0,2 a 2,0 mm (19 opções de passes).

O valor ótimo do avanço de acabamento (f_s) é selecionado em função do menor valor de avanço obtido do limite superior das restrições associadas ao avanço:

$$f_s^* = \min(f_s^1, f_s^2, f_s^3, f_s^4, f_s^5)$$

Para f_s^* , T (vida ferramenta) e a_{ps} determina-se a velocidade de corte ótima $V_{c_s}^*$ através da equação expandida de Taylor, e o mínimo custo e/ou tempo para a operação de acabamento é calculado da função objetivo. Assim determina-se os parâmetros de corte otimizados, o mínimo custo e o mínimo tempo de produção para cada uma das opções de passes de acabamento, montando-se uma tabela de passe simples de acabamento para $a_{ps(j)}$, $V_{c_s(j)}$, $f_s(j)$, $UC_{s(j)}$, $UT_{s(j)}$ onde j varia de 1, 2, 3, ... s_{passe} .

Procedimento semelhante é adotado para as operações de *desbaste*, onde monta-se uma tabela de passe simples de desbaste para $a_{pr(j)}$, $V_{c_r(j)}$, $f_r(j)$, $UC_{r(j)}$, $UT_{r(j)}$ onde j varia de 1, 2, 3, ... r_{passe} .

Segundo Estágio Neste estágio calcula-se o número de passes de desbaste (n_i), que pode variar em função do intervalo de profundidade de corte de acabamento definido. Deve-se arredondar o valor calculado de n_i , trabalhando-se com um número inteiro, para o intervalo determinado:

$$\frac{a_p - a_{psL}}{a_{prU}} \leq n_i \leq \frac{a_p - a_{psU}}{a_{prU}} \quad (7.10)$$

Faz-se a seguir uma combinação otimizada dos passes de corte de acabamento (a_{ps}^*) e desbaste ($a_{pr(i)}^*$) para ($n_i + 1$) passes, e o mínimo tempo e custo de usinagem total são determinados. Um algoritmo genético é utilizado para obter a combinação otimizada das profundidades de corte e das funções objetivos, levando em conta as seguintes restrições:

- devem existir n_i passes de desbaste e um de acabamento;
- as profundidades de corte individuais de n_i dos passes de desbaste e o passe de acabamento devem estar contidas no intervalo de profundidades de corte de desbaste e acabamento definidos pelo sistema/usuário;
- a soma das profundidades de corte individuais para ($n_i + 1$) passes deve ser igual a profundidade total de material a ser removida da feature de usinagem.

7.7.4 Algoritmo Genético

O algoritmo genético (GA) é iniciado com um conjunto de profundidade de corte com ($n_i + 1$) passes escolhidos de forma aleatória dos passes individuais de desbaste e acabamento calculados no primeiro estágio, sendo um método de pesquisa adaptativo geral que permite a determinação do ótimo global a partir de funções objetivos (função custo, *fitness*). O algoritmo faz analogia com mecanismos utilizados pela evolução biológica natural. A partir do conjunto inicial, novos e melhores conjuntos são reproduzidos usando alguns indivíduos dos grupos selecionados por suas respectivas aptidões.

Um candidato à solução, um ponto no espaço de pesquisa, é representado por uma seqüência de números (binário ou decimal) denominado de cromossomo ou uma *string*. A probabilidade de

Tabela 7.3: Tabela montagem cromossomo associado ao algoritmo genético.

i	String 4 passe, ap=8mm	UC_{total} UC_{mini}	Fitness $UC_{maxi}=M-UC_{min}$	Experado $E_i=UC_{maxi}/U_{medio}$	Real
1	2.3 2.3 2.2 1.2	2.913	0.078	1.748	2
2	2.1 3.0 1.7 1.2	2.958	0.33	0.739	1
3	1.4 2.4 3.0 1.2	2.991	0	0	0
....					
10	2.5 2.9 1.6 1.0	2.982	0.009	0.201	0

seleção de um cromossomo é diretamente proporcional ao valor da sua função objetivo (*fitness*). A busca tem início sobre uma população inicial que é formada por um conjunto de cromossomos. Cada cromossomo, uma *string*, é representado por uma coleção de $n + 1$ elementos (genes) correspondendo a $n + 1$ passes, sendo n o número de passes de desbaste; e 1 o passe de acabamento.

A geração inicial é formada por uma população de 10 cromossomos. Quatro operadores podem atuar sobre a população de uma geração:

- Reprodução: seleção de cópias de cromossomos proporcionais ao valor da função objetivo (seleção de pares via *Roulette Wheel*);
- *Crossover* (Recombinação): uma troca de secções de cromossomos entre pares (definição de pontos de *crossover* e dos pares de cromossomos). Utiliza-se uma probabilidade de *crossover* de 80% (PCROSS);
- Mutação: uma modificação randômica de cromossomos. Utiliza-se uma pequena probabilidade de mutação de 20% (PMUT);
- Ajuste de valores no cromossomo: este quarto operador é introduzido a fim de ajustar os valores das *strings* dos genes para respeitar as três restrições impostas à função objetivo.

Como estratégia de solução é utilizado uma codificação em *strings* de $(n+1)$ números reais, que representam “ n ” passes de desbaste e “um” passe de acabamento. Se $n_i = 3$, ter-se-á 3 passes de desbaste e um passe de acabamento, sendo representado por uma *string* de números reais com quatro elementos (genes). Utiliza-se um algoritmo genético modificado, fazendo-se ajustes de valores nos *strings* (cromossomos), caso seus valores não respeitem as restrições associadas à profundidade de corte, após a operação de *crossover* e mutação. A tabela 7.3 apresenta a montagem dos cromossomos e a aplicação do GA baseado no método da *Roulette Wheel*. São mostrados os valores da função objetivo (UC, custo de produção), a função *Fitness*, o valor esperado e o valor real, que indica a quantidade de cromossomos mais aptos, e que serão preservados para as próximas gerações.

O tamanho da população inicial é de 10 cromossomos (*strings*) e o algoritmo gera quinhentas gerações para obtenção do ótimo global, sendo utilizado o *toolbox* GA do *software MatLab* para resolução do problema (<http://www.mathworks.com/products/gads>), apresentado na figura 7.31.

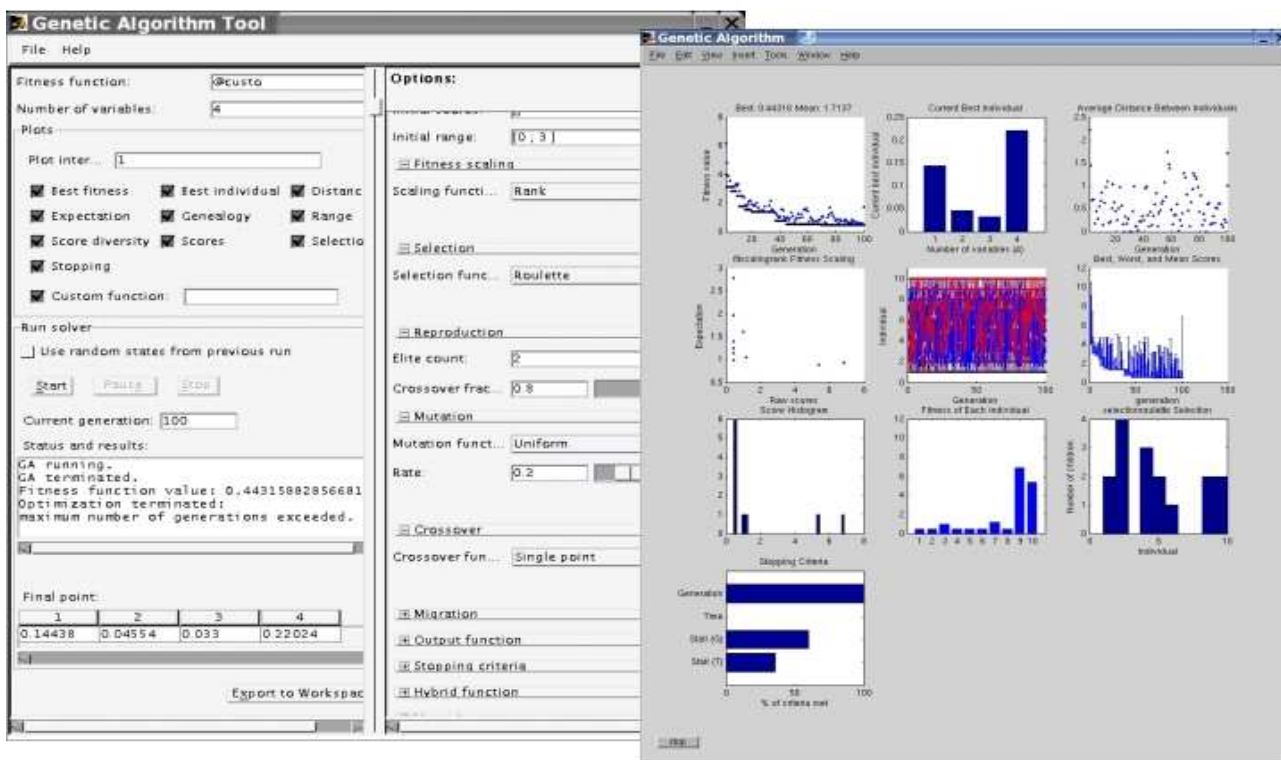


Figura 7.31: Toolbox Algoritmo Genético do MatLab, utilizado na determinação das condições de usinagem otimizadas.

7.8 Linearização do Plano de Processos com Alternativas

Até este momento tem-se especificado uma lista de *features* de usinagem a serem trabalhadas sendo associado, para cada *feature* de usinagem alternativas de operações de corte, definindo os *working-steps*. Para cada alternativa de operação é associado um modelo de tempo e custo e uma estratégia de movimentação de ferramenta, sendo realizado a seguir a otimização das condições de usinagem baseado em um modelo multi-passe e a utilização de algoritmos genéticos, conforme apresentado no item anterior. A figura 7.24 sintetiza a representação dos *workingsteps* com alternativas de operações, incluindo os dados relativos de cálculo de tempos e custos para cada operação de usinagem. Com estes dados disponíveis é possível linearizar o plano de processos (*workplan* não-linear) utilizando uma função objetivo baseada em mínimo custo ou mínimo tempo de usinagem.

A representação do plano de processo não-linear é baseada em STEP-NC através das entidades relativas a um executável (*Executable*) definidas no SUPERTYPE associada à entidade *program_structure* (estrutura de programa STEP-NC) como: Entidade *Selective* (OU) e Entidade *Non_sequential* (E). A figura 7.32 (a) apresenta os SUPERTYPE de uma entidade *Executable*, segundo STEP-NC (Part 10) para representação de planos de processos não-lineares. A figura 7.23 (c) apresenta um exemplo de um plano de processo não-linear e o *workplan* associado, sendo constituído por várias alternativas de roteamentos, descrevendo o plano de processo não-linear.

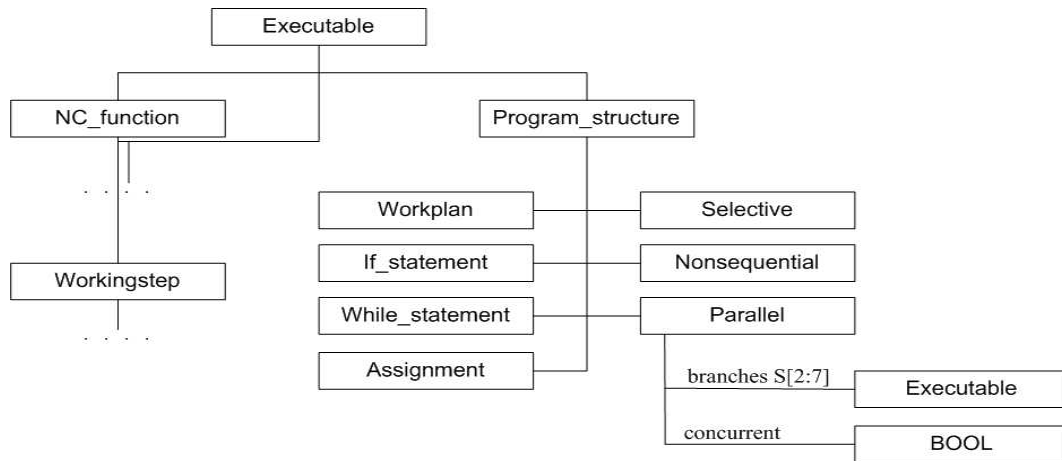


Figura 7.32: Entidades associadas à representação de planos de processos não-lineares segundo STEP-NC (Part 10), em Express-G.

7.8.1 Metodologia de Otimização

Neste item é apresentada uma abordagem de linearização do plano de processo não-linear utilizando um método de linearização baseado em algoritmo genético adaptado de ROCHA *et al.* (1999) e ZHANG *et al.* (1997), determinando o plano de processos linearizado, que será representado por um *workplan* linearizado, usando uma estrutura de programa para representação de alternativas, segundo STEP-NC.

ROCHA *et al.* (1999) descrevem um algoritmo genético usado para gerar a seqüência de operações de usinagem e selecionar máquinas e ferramentas, tendo como critério de otimização o custo de produção em termos de tempo, mostrando a viabilidade da utilização de algoritmos genéticos neste tipo de aplicação. Este algoritmo será alterado para trabalhar com o conceito de *workingstep* e planos de processos não-lineares.

A formulação do problema matemático segundo HALEVI (1999) é a seguinte: “Dada uma lista de operações a serem executadas e uma lista de recursos disponíveis, uma decisão é requerida para definir qual máquina ou máquinas usar, quais operações serão executadas em cada máquina, qual a seqüência e quais as condições de corte que serão empregadas”.

No caso do problema em análise não há a figura de várias máquinas, ou seja máquinas alternativas para executar as operações, e sim operações alternativas para usinar uma determinada *feature* de usinagem, com a utilização de diferentes tipos de ferramentas, estratégias de usinagem e *setup*, representadas através de um grafo E/OU (figura 7.23). Ou seja, já se conhece a *feature* de usinagem, volume a ser removido, podendo-se associar várias opções de operações de usinagem à esta *feature*. Assim define-se um conjunto de *workingstep* que podem usinar a mesma *feature*, conforme apresentado na figura 7.23 (b).

A formulação do problema matemático definido por HALEVI (1999) deve ser ajustado para o caso em análise, tornando-se: “Dada uma lista de operações associadas às *features* de usinagem (*workinsteps* de usinagem) a serem executadas, e uma lista de recursos disponíveis apresentando alternativas de operações com diferentes ferramentas, estratégias de usinagem e *setup* (*workplan*, *program_structure* e *workingsteps*), segundo um esquema de representação de dados STEP-NC, para

uma dada máquina-ferramenta; uma decisão é requerida para definir qual operação, dentre as alternativas (*workingsteps*) disponíveis, deve ser especificada para cada *setup*, qual a seqüência de usinagem e quais as condições de corte que deverão ser empregadas, ou seja, como linearizar o *workplan* com alternativas”.

A figura 7.33 apresenta os possíveis “j” *workplans* por *setup* (rotas no grafo E/OU) e o “n” *workingsteps* associados representados em uma matriz *workplan* x *workingstep*, devendo ser analisado em conjunto com o grafo de seqüência de *workingstep*, grafo E/OU, conforme figura 7.23. Cada célula da matriz contém um valor de tempo ou custo mínimo, já linearizados, associado à operação de usinagem sobre uma *feature*. O custo/tempo do *setup* do conjunto máquina-peça-ferramenta é calculado em paralelo como a soma dos custos/tempos relativos ao:

- *setup* da máquina;
- *setup* da ferramenta;
- troca de ferramenta entre usinagens de *features* sucessivas, adjacentes (troca de ferramenta no magazine);
- *setup* da peça entre usinagens de *features* sucessivas, adjacentes (por exemplo, inversão da peça na placa, permitindo usinagem do lado esquerdo e depois do lado direito).

Para otimização do plano de processo não-linear, utiliza-se de um critério de otimização baseado na minimização do custo de produção ou do tempo de usinagem global através da abordagem de algoritmo genético, levando em conta a minimização do *setup*.

Para avaliação do impacto do *setup* sobre as alternativas de operações para um determinado *workplan*, monta-se uma matriz relacionando um *workingstep* (“i”) com o *workingstep* posterior (“i +1”), indicando na célula o tempo ou o custo de alteração de *setup* da operação atual sobre a *feature*, para a próxima operação na *feature* seguinte, de acordo com o grafo E/OU; ou seja verifica-se se há uma relação de precedência que permita a transição de *setups* entre os dois *workingsteps*. Caso não seja permitido a alteração de *setup* de *workingstep* para outro, a célula fica vazia, estabelecendo-se as regras de precedências entre *workingsteps*. Na figura 7.34 (a-d) estes conceitos associados à peça exemplo são apresentados para seis *features* de usinagem (figura 7.34(a)). Cada *feature* de usinagem tem duas operações alternativas (figura 7.34(b)). Um grafo de precedência é apresentado em (figura 7.34(c)) e a matriz custo de *setup* é apresentado em (figura 7.34(d)).

A seguir define-se uma função objetivo baseada em custos ou tempos, que será utilizada no procedimento de linearização do plano de processos com alternativas para o domínio de peças rotacionais, utilizando a abordagem de algoritmo genético. As funções objetivos calculadas para cada *workplan* (*CustoWorkplan* (CW) e *TempoWorkplan* (TW)), associado a um *setup* da peça, são:

$$CW = \sum_{i=1}^n (CustoWorkingstep_i) + \sum_{i=1}^{n-1} (CustoSetup_{i \rightarrow i+1} * \Omega (Workingstep_i, Workingstep_{i+1})) \quad (7.11)$$

Workingsteps Workplan não linear	1	2	3	4	n
1	3	6		5	7
2		2	3	4	5
3	10	20	30		10
4		13	12	16	18
j	10	5	10		3

Figura 7.33: *Workplans* alternativos e *workingsteps* associados, onde os valores das células estão associados ao custo ou tempo relativo à operação de usinagem.

$$TW = \sum_{i=1}^n (TempoWorkingstep_i) + \sum_{i=1}^{n-1} (TempoSetup_{i \rightarrow i+1} * \Omega (Workingstep_i, Workingstep_{i+1})) \tag{7.12}$$

sendo $CustoSetup_{i \rightarrow i+1}$ ou $TempoSetup_{i \rightarrow i+1}$ o custo ou tempo de *setup* para trocar do *Workingstep_i* para o *Workingstep_{i+1}* (7.13) e n o número de *workingstep* do *workplan*:

$$\Omega (x, y) = \begin{cases} 1, & \text{se } x \neq y \\ 0, & \text{se } x = y \end{cases} \tag{7.13}$$

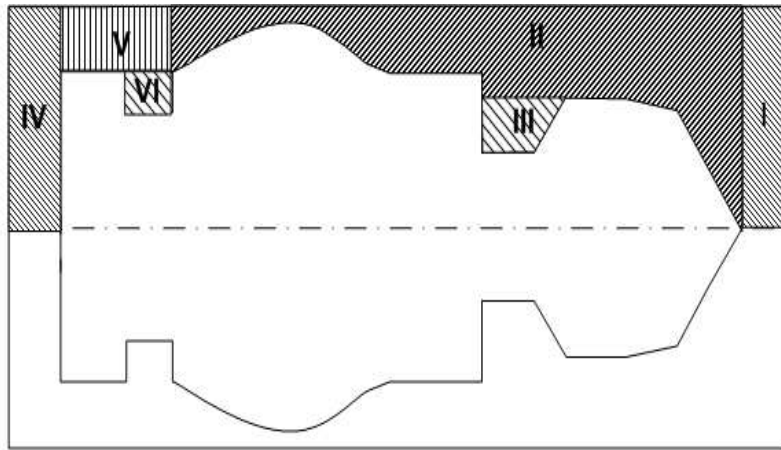
7.8.2 Algoritmo Genético

A representação dos cromossomos é feita através de *strings*. O número de posições em cada *string* é igual ao número de *workingsteps* do *workplan* analisado. Cada posição contém uma referência à operação associada à uma *feature* de usinagem (*workingstep*). Cada cromossomo, uma *string*, é representado por uma coleção de “n” elementos (genes) correspondendo a “n” *workingsteps*.

Faz-se a geração da população inicial de forma randômica, obedecendo a relação de precedência entre os *workingsteps*, sendo as *strings* constituídas de seqüências válidas, sem violação da precedência pré-estabelecida. São utilizados os *workplans* já definidos anteriormente. A geração inicial é formada por uma população de 4 cromossomos. Quatro operadores podem atuar sobre a população de uma geração:

- Reprodução: seleção de pares via *Roulette Wheel*;
- *Crossover* (Recombinação): utiliza-se uma probabilidade de *crossover* de 80% (PCROSS);
- Mutação: utiliza-se uma probabilidade de mutação de 20% (PMUT);
- Ajuste de valores no cromossomo: este quarto operador é introduzido a fim de ajustar os valores das *strings* dos genes para respeitar o grafo de precedência, garantido a geração de *strings* exequíveis sem a violação de precedência.

Utiliza-se um algoritmo genético modificado, fazendo-se ajustes de valores nos *strings* (cromossomos), caso seus valores não respeitem as restrições de precedência, após a operação de *crossover* e



(a)

Nº	Operação	Duração
1	Facer G75	60
2	Tornear Desbaste G71	74
3	Sangrar G72	161
4	Tomear Acabamento G70	122
5	Sangrar Acabamento G70	172
6	Tornear Desbaste G71	150
7	Sangrar G72	145
8	Tomear Acabamento G70	77
9	Sangrar Acabamento G70	77

(b)

Setup1

I: Faceamento G75 (1)

II: Tornear Desbaste G71 (2)
ou Sangrar G72 (3)

II: Tornear Acabamento G70 (4) ou
Sangrar Acabamento G70 (5)

III: Tornear Desbaste G71 (6)
ou Sangrar G72 (7)

III: Tornear Acabamento G70 (8) ou
Sangrar Acabamento G70 (9)

(c)

		1	2	3	4	5	6	7	8	9
1	Facer G75		3	5						
2	Tornear Desbaste G71				3	15				
3	Sangrar G72				15	5				
4	Tornear Acabamento G70						3	15		
5	Sangrar Acabamento G70						15	5		
6	Tornear Desbaste G71								3	15
7	Sangrar G72								15	5
8	Tornear Acabamento G70									
9	Sangrar Acabamento G70									

(d)

String Nº	String	Valor
1	1,2,4,6,8	515
2	1,3,5,7,9	635

(e)

String de crossover			
Pai 1	Pai 2	Pontos Cruzados	Filho 1
1,2,4,6,8	1,3,5,7,9	2,5	1,3,4,6,9

(f)

String de mutação		
String	Resultado da string	Exequível?
1,2,4,6,8	1,2,4,6,9	Sim

(g)

Figura 7.34: Exemplo: (a) peça (b) *features* de usinagem e operações (c) grafo de precedência de *workingsteps* (d) matriz de *setup* de *workingstep* (e) *strings* (f) operador *crossover* (g) operador de mutação.

mutação. Utiliza-se a função objetivo (*Fitness*) baseada em custo ou tempo de *setup*. A figura 7.34 (e) apresenta duas *strings* que representam dois possíveis *workplan*, onde os genes da *string* estão associados aos *workingsteps* do *workplan* analisado. Na figura 7.34 (f) e (g) são apresentados os operadores de *crossover* e mutação, respectivamente.

Desde que todas as alternativas de operações são consideradas (ferramentas, *setup* peça, estratégias de movimentação de ferramenta, entre outros) todo o espaço de solução é analisado, permitindo a determinação do ótimo global.

7.9 Geração do Programa NC (ISO 6983)

O programa NC é gerado diretamente para o conjunto máquina-ferramenta/CNC segundo o padrão ISO 6983 (ISO 274 D), podendo utilizar-se de algumas funções especiais da máquina CNC, os chamados ciclos fixos, que executam uma determinada operação de usinagem utilizando uma estratégia de movimentação de ferramenta padronizada pelo fabricante do CNC. A máquina-ferramenta é selecionada no início da modelagem da peça pelo usuário, no caso um centro de torneamento Galaxy 15M da Romi com CNC Fanuc 18i-ta. Pode-se ter vários módulos de macros, em função dos conjuntos de máquinas-ferramenta/CNCs disponíveis na base de dados de máquinas, que poderão ser utilizados para a geração do programa NC. Outra alternativa seria gerar o programa NC em STEP-NC para ser utilizado em uma máquina-ferramenta com controlador aderente à STEP-NC.

A abordagem para geração do programa de usinagem é aderente à STEP-NC, onde cada *Machining_workingsteps* ou *Turning_workingstep*, tem associado uma ou mais operações de usinagem (alternativas) com uma estratégia de movimentação de ferramenta, que pode ser baseada em ciclo fixo ou não, padronizando determinada operação de usinagem, descrevendo a definição de movimentação de ferramentas e condições de usinagem através de um trecho de programa parametrizado.

O plano de processos linearizado, gerado pelas atividades de CAPP descritas anteriormente, apresentadas através de *workingsteps* de usinagem e torneamento para um determinado *workplan*, contém as informações necessárias para geração do programa NC, que são: máquina; *feature* de usinagem e de torneamento; operação; ferramenta; condições de usinagem.

Desta forma, conhecendo-se o código "G" da máquina especificada para a execução da operação, pode-se utilizar uma "máscara" cujos os parâmetros serão instanciados com os valores correspondentes para cada operação. Assim cada *workingstep* associado à uma operação de desbaste ou de acabamento modelada pelo sistema tem um gabarito da operação, que poderá ser utilizado na composição do programa NC da peça. Para cada máquina é conhecido o ponto de troca de ferramenta em relação ao zero máquina (G53). Utiliza-se G54 (*Setup1*) e G55 (*Setup2*) para definição do zero peça.

Dentre os ciclos fixos que podem ser utilizados nos *Workingsteps*, e normalmente disponibilizados pelos fabricantes de tornos CNC e centros de torneamento, têm-se para o CNC 18i-ta (ROMI, 2003):

- G76: Ciclo automático de rosca;
- G78: Ciclo de rosca semi-automático;
- G70: Ciclo de acabamento, após utilização dos ciclos de desbaste G71, G72 e G73;

- G71 (Tipo I): Ciclo de desbaste automático longitudinal com pré-acabamento (perfil com aumento (perfil externo) ou diminuição (perfil interno) uniforme ao longo do eixo X), não permitindo *cut-in* e *cut-out* ao longo do eixo X;
- G71 (Tipo II): Ciclo de desbaste automático longitudinal com pré-acabamento, permitindo *cut-in* e *cut-out* de até dez concavidades (operações de mergulho);
- G72: Ciclo automático de desbaste transversal com pré-acabamento;
- G73: Ciclo automático de desbaste paralelo ao perfil final (utilizado para *blanks* fundidos e forjados);
- G74: Ciclo de furação axial (ferramenta acionada);
- G87: Ciclo de furação radial (ferramenta acionada);
- G74: Ciclo de desbaste longitudinal;
- G75: Ciclo de canais;
- G75: Ciclo de faceamento;
- G77: Ciclo de desbaste longitudinal;
- G77: Ciclo de desbaste cônico;
- G79: Ciclo de desbaste transversal;
- G79: Ciclo de desbaste cônico;
- G83: Ciclo de furação axial (ferramenta não-acionada);
- G84: Ciclo de rosqueamento com macho rígido axial (ferramenta não-acionada);
- G86: Ciclo de mandrilamento ou alargamento axial;
- G63: Ciclo zeração de ferramentas automático (utilizando *tooleye*).

O algoritmo (5) apresenta um trecho de programa em linguagem Java (POO) definindo um gabarito para um *Turning_workingstep* associado à uma operação de torneamento longitudinal multi-passe, com operações de desbaste e acabamento, implementado através dos ciclos fixos G71 (tipo I) e G70, gerando a inclusão de comentários, códigos G, plano de aproximação e segurança, compensação de raio de ferramenta, entre outros, no programa NC que será gerado.

O programa NC completo para a fabricação de uma peça numa determinada máquina é obtido pela combinação seqüencial dos trechos de programa para cada operação associada a uma *feature* (*Machining_workingstep*) no plano de processos linearizado, segundo o grafo de precedência de *features* de usinagem para cada *setup* da peça. O mesmo procedimento é aplicado para a usinagem nas outras máquinas presentes no sistema de manufatura. Assim o programa NC linearizado é composta por uma seqüências de *Machining_workingstep* e de *Turning_workingstep*.

Algoritmo 5 Programa em Java para construção de blocos em código G.

```

codigoIntermediario.add(getLinhaDoCodigoG() +
"(Operacao - CICLO FIXO G71 TIPO1 DESBASTE E G70 ACABAMENTO)");
//Ir para o plano de seguranca
_codigoIntermediario.add(getLinhaDoCodigoG() + "G0X" + WS.planoDeSeguranca.x + "Z"
"Z" + (/*blankUtilizado.L1 +*/ WS.planoDeSeguranca.z));
//Adiciona a primeira linha do ciclo fixo
//adiciona um pedaço da segunda linha do ciclo fixo
_codigoIntermediario.add(getLinhaDoCodigoG() + "G71U" + U + "R" + R);
P = numeroDaLinha+20;
String tmp = getLinhaDoCodigoG() + "G71P" + P;
//Coloca no _codigoIntermediario o plano de Aproximacao
_codigoIntermediario.add(getLinhaDoCodigoG() + "G0X" + WS.planoDeAproximacao.x);
_codigoIntermediario.add(getLinhaDoCodigoG() + "G42");
_codigoIntermediario.add(getLinhaDoCodigoG() + "G1W-10.0");

```

7.10 Geração da Documentação Final do Plano de Processo Linearizado

O plano de processo linearizado é apresentado no formato de uma planilha eletrônica (tabela) formatado em XML (*Extensible Markup Language*) indicando todas as informações referentes ao plano de processo linearizado e programa NC (figura 7.35), disponibilizado através de uma base de dados relacional:

- Dados administrativos (projetista, nome da peça, descrição, tamanho do lote e data);
- especificação da máquina CNC;
- *setups* da peça;
- nome operações elementares de usinagem;
- número da operação;
- ferramentas de corte e porta-ferramenta;
- condições tecnológicas de usinagem (a_p , f e V_c);
- estimativa de tempo por operação (tempo padrão);
- estimativa de tempo para fabricação da peça;
- custo estimado por peça;
- URL referente ao arquivo do programa NC;
- Programa NC.

n°	Operação	Castanha (mm)	Ferramenta	Posição Torre	Ap (mm)
n°	SANGRAMENTO	Diâmetro_50	LF123G20-2020B N123G200300003-GM4025	0 3 0 3	2.0
n°	SANGRAMENTO	Diâmetro_50	LF123G20-2020B N123G200300003-GM4025	0 3 0 3	2.0
n°	SANGRAMENTO	Diâmetro_50	LF123G20-2020B N123G200300003-GM4025	0 3 0 3	2.0
Setup					
n°	Operação	Castanha (mm)	Ferramenta	Posição Torre	Ap (mm)
n°	FACEAMENTO	Diâmetro_50	SVJBL220K16 VBMT160408-MM2025	0 7 0 7	2.0
n°	TORNEAMENTO EXTERNO	Diâmetro_50	SVJBL220K16 VBMT160408-MM2025	0 7 0 7	2.0
Código G					
O1000(PECA R1 - UFSC) N10G21G40G90G95 N20M36 N30M37 N40(INICIO SETUP1) N50(WORKINGSTEP - FEATURE DE USINAGEM Fu1) N60G53 N70G0X390Z350T00 N80T0707(Tool DESBASTE GERAL) N90G54 N100G96S300 N110G92S4000 N120M4 N130M8 N140(Operação - CICLO FIXO G75 FACEAMENTO)					

Figura 7.35: Documentação final do plano de processo linearizado em XML.

Capítulo 8

Abordagem Lógica Para CAPP: Métodos e Algoritmos Concebidos

Neste capítulo são apresentados, de forma detalhada, os métodos e algoritmos desenvolvidos para resolução das dez atividades previstas no CAPP. É utilizada uma abordagem baseada em um estudo de caso; peça exemplo constituída por *features* concêntricas, onde é explicado a utilização de cada um dos métodos e algoritmos concebidos, associados às dez atividades do CAPP.

8.1 Peça Exemplo: Estudo de Caso

A figura 8.1, apresenta o desenho mecânico em 2D e 3D, projeto detalhado, da peça exemplo modelada pelo sistema WebCADbyFeatures e que será usinada, sendo constituída por um total de 39 *features* de forma dos tipos OD, ID, *groove*, face, rosca, arredondamento e chanfro. A *feature* de material especificada é o aço 1025. O sistema CAPP generativo inicia a execução automática das dez atividades de planejamento de processo, a partir do modelo de *features* gerado pelo sistema WebCADbyFeatures, para a peça exemplo modelada com *features* rotacionais concêntricas (regiões interna e externa), não tendo *features* Eixo C (não-concêntricas).

8.2 Mapeamento de *Features* de Projeto para *Features* de Usinagem

O CAPP lê o arquivo associado à biblioteca de *features* de forma de projeto gerado pelo sistema WebCADbyFeatures e inicia o mapeamento de *features* da visão de projeto para a visão fabricação, mapeando as *features* de usinagem, a partir do método desenvolvido, denominado de decomposição baseada em *setup*, geometria e operações. O algoritmo 6 apresenta os principais passos que são realizados pelo WebCAPP após o carregamento do arquivo com a biblioteca de *features*.

Antes de iniciar a decomposição é necessária uma normalização da biblioteca de *features* (algoritmo (6)) gerada pelo WebCADbyFeatures, para permitir uma identificação mais adequada das *features* de torneamento e usinagem (ISO 14649, 2003). Por exemplo, um *groove* modelado por faces e OD não está identificado de fato na biblioteca de *features*, gerado pelo WebCADbyFeatures,

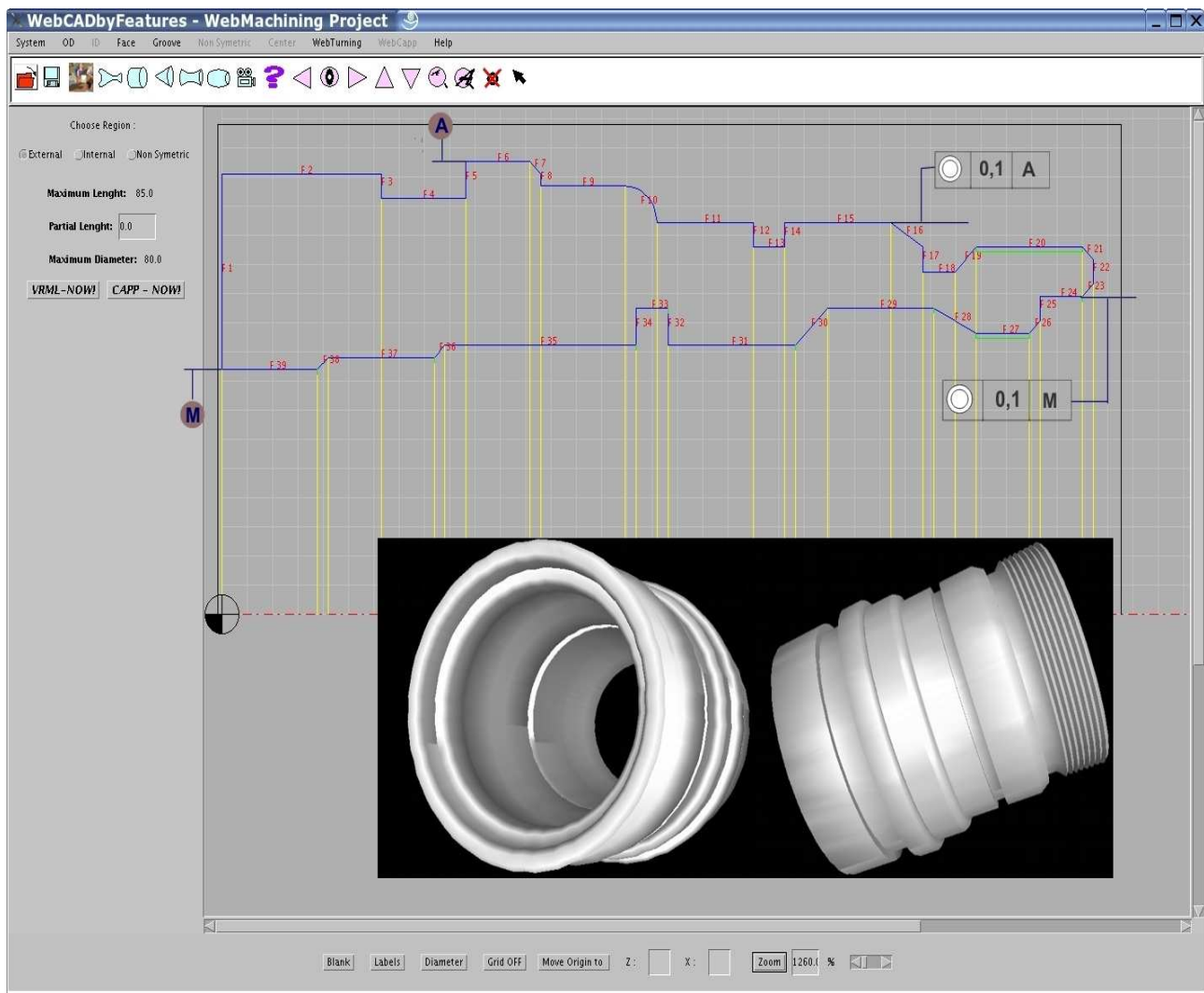


Figura 8.1: Peça exemplo em 2D e 3D sendo constituída por 39 *features* de forma do tipo OD, ID, canal, face, arredondamento e chanfro, modelado pelo sistema WebCADbyFeatures.

Algoritmo 6 Principais passos para implementação do WebCAPP.

1. *Normalização do vetor de features (biblioteca de features)*
 2. *Cria vetor “_featureInvertida”*
 3. *DecomposiçãoOrientadaASetup*
 - 3.1 *faz a separação por setup*
 - 3.2 *ordena os setups*
 - 3.3 *escolhe a featureDeFixação de cada um dos setups*
 - 3.4 *seleciona qual o vetor de features (normal ou o invertido) a ser utilizado para perfil interno ou externo crescente*
 - 3.5 *armazenar dados no vetor de Setups*
 4. *DecomposiçãoOrientadaAGEometria: cria features de usinagem*
 5. *DecomposiçãoOrientadaAOperação: cria Workingsteps*
 6. *Gera OtimizaçãoCondiçõesdeUsinagem*
 7. *Lineariza Workplan*
 8. *Gera Programa NC*
 9. *Documentação XML*
-

como um *groove* (canal), e sim como uma sucessão de *features* de faces e ODs. É necessário uma normalização da biblioteca de *features*, mapeando estas *features de forma* de projeto em *features* de torneamento, sendo no caso denominado como um *groove complexo*, constituído por uma lista de *features* básicas de faces e ODs. O mesmo ocorre com relação à uma lista de *features* ODs e faces que são mapeadas como uma *feature* de torneamento, na decomposição orientada à geometria e à operação, como será apresentado neste item.

Splines são normalizadas através do conceito de *SplineVirtual*, sendo esta *feature* virtual (na realidade não existe no modelo) mapeada a partir da *spline* original como uma *feature* do tipo OD ou ID retilínea com diâmetro igual ao maior diâmetro da *spline* e comprimento igual ao da *spline*. O conceito de *SplineVirtual* é usado para facilitar o procedimento de decomposição orientada ao *setup*, geometria e operação, gerando um volume de usinagem que sofrerá uma operação de torneamento longitudinal de desbaste em conjunto com as *features* de torneamento.

Após a identificação da *SplineVirtual* e seu agrupamento em um dos *setups*, a *SplineVirtual* é remapeada como *spline*, e a seguir é aproximada por um conjunto de *features* do tipo OD cônicas, tendo no mínimo 10 *features* do tipo OD cônica entre dois pontos de controle da *spline*. Por exemplo, uma *spline* com seis pontos de controle será aproximada por 50 *features* do tipo OD cônica, que irá gerar um programa NC com sessenta blocos de código G associados a uma interpolação linear em X e Z (G01 X Z) contendo os pontos (Z,X) do início ao final da *spline*, tendo uma resolução variável entre 0,1 a 0,01 mm. Como a memória do CNC é limitada a 20 mil caracteres deve haver um compromisso entre a resolução e a memória disponível do CNC.

A biblioteca de *features* é implementada através de um conjunto de objetos de classes constituído por *features* de forma, tolerâncias, referências, entre outras, definida na taxonomia de *features* CAM-I (CAM-I, 1986) e armazenadas na memória do computador do Agente de Interface WebCADbyFeatures em um vetor, como um objeto de biblioteca de *features* (conjunto de *features* apresentando relacionamentos geométricos e não-geométricos). Usa-se o processo de serialização de objetos para realizar a conversão da representação de um objeto em memória (biblioteca de *features*) para uma sequência de *bytes* enviada para um *ObjectOutputStream*¹, que por sua vez está associado a um arquivo em disco (armazenamento local, computador do Agente de Interface WebCADbyFeatures) e por uma conexão de rede via Agente Facilitador (JATLite), fazendo o *upload* e armazenamento do arquivo via FTP (armazenamento no servidor WebMachining).

O processo de normalização da biblioteca de *features*, armazenada em um vetor na memória denominado de *DadosDeProjeto.ArmazenadorDeFeatures*, ocorre através da varredura deste vetor, identificando-se determinados padrões que definem um *groove complexo*, como o aparecimento de pontos de inflexão que invertem o padrão de crescimento ou decrescimento do diâmetro da peça (cota X) ao longo do eixo Z, bem como a identificação de uma *spline*. O algoritmo de varredura percorre o perfil da peça a partir da face esquerda para a direita, e depois da face direita para a esquerda, identificando o aparecimento de um padrão *groove complexo*, e listando as *features* de forma de projeto da biblioteca de *features* original, que compõem o *groove complexo* identificado ou mesmo de uma *spline*.

A seguir são apresentados os métodos e algoritmos concebidos para o mapeamento.

¹Objetos de classes (em Java) para os quais são previstas serializações e desserializações devem implementar a interface *java.io.Serializable*.

Algoritmo 7 Algoritmo para decomposição orientada ao *setup*.

Setup ();
 if ($L/D \geq 4$) or ($(L/D \geq 3)$ and $D \geq 100$) then "Peça Longa";
 if ($L/D \leq 2$) then "Peça Curta";
 if ($2 < L/D < 4$) then "Peça Intermediária";

Algoritmo 8 Algoritmo para definição do número de fixações da peça.

NúmeroDeFixações ();
TipoDeBlank ();
TipoDePerfilPeçaAcabada ();
GeometriaCastanha ();
 If *NúmeroSetup* = 1 then *VerificaPosições1DeFixação* ();
 If *NúmeroSetup* = 2 then *VerificaPosições2DeFixação* ();
 If *NúmeroSetup* = 3 then *VerificaPosições3DeFixação* ();
SaídaAlternativasViáveisNúmeroSetup ();
DefineMelhorAlternativaSetup ();

8.2.1 Decomposição Orientada ao Setup

A decomposição orientada ao *setup* é realizada a partir do algoritmo (7), que faz a classificação da peça em função do tipo de peça (longa, curta ou intermediária), devendo-se verificar as alternativas de *setup* para fixação entre pontas (placa e contra-ponta) ou fixação na castanha, associado ao número de *setups* necessários: um, dois ou três.

Considerando que o centro de torneamento Romi Galaxy 15M possui apenas o cabeçote principal (uma placa de fixação), não tem instalado um alimentador de barras e nem mecanismos de acionamento para fixação entre pontas; ter-se-á como opções para fixação da peça bruta, designada na biblioteca de *features* por *blank*, ou de uma peça semi-acabada, a possibilidade de se ter dois *setups* (fixação lado esquerdo e fixação lado direito (peça invertida)) ou um único *setup* (fixação lado esquerdo e corte da peça).

Analisando a peça exemplo, tem-se $L=80\text{mm}$ e $D=75\text{mm}$, logo $L/D=1,07$. Assim a peça é classificada como uma peça curta, que poderá ter um ou dois *setups*. A seguir é analisado a necessidade de um ou dois *setups* para usinagem da peça, a partir da geometria do *blank* e da geometria da castanha. Utiliza-se o algoritmo (8) para esta finalidade.

Têm-se as opções de cinco tipos de *blank* (barra sólida, pré-furo, pré-usinado, tubo ou fundido). O próprio usuário, Agente de Interface, durante a modelagem define o tipo de *blank* a ser usado. Tem-se a opção de deixar a definição da matéria-prima a cargo do CAPP, que pode especificar automaticamente o *blank* mais adequado para geometria da peça acabada modelada, levando em conta o estoque de *blanks* presente no depósito de peças brutas do chão-de-fábrica, o sobrematerial necessário (calculado de modo reverso, acrescentando-se ao perfil da peça acabada todas as operações e sobremateriais necessários até chegar ao perfil do *blank*) e a geometria da castanha.

Quando opta-se pela definição de *blank* gerado pelo CAPP, pode-se utilizar, caso seja possível, o *setup* atual da geometria de castanhas, não sendo necessário um novo *setup* associado à geometria de castanha ou um ajuste no conjunto de castanha presente na máquina. Estes aspectos são levados em

Algoritmo 9 Algoritmo para definir candidatas à superfícies de fixação (*features* internas e externas) para o *Setup1* e *Setup2*.

SuperfícieFixação ();

GeometriaBlank ();

GeometriaPeça ();

IF (GeometriaBlank = Barra) THEN "Fixação OD_{LE} (OD Lado Esquerdo) ou OD_{LD} (OD Lado Direito) só Região externa é Viável";

IF (GeometriaBlank = Tubo) THEN "Fixação OD/ID_{LE} (OD/ID Lado Esquerdo) ou OD/ID_{LD} (OD/ID Lado Direito) são Viáveis, OD e ID";

VerificarRestriçõesOD ();

ComprimentoOD ≥ 5 mm;

VerificarRestriçõesID ();

ComprimentoID ≥ 5 mm;

VerificarGeometriaCastanhaPlaca ();

ElegerMelhoresCandidatasSuperfícieDeContato ();

conta pelo sistema de programação da produção da Célula Flexível de Manufatura (FMC), sendo o centro de torneamento uma das unidades desta FMC (ÁLVARES *et al.*, 2005). Estas informações e as restrições associadas são disponibilizadas pelo CAPP para o sistema de programação da produção da FMC, visando a fabricação da peça a distância.

Em função do tipo de perfil da peça acabada, define-se a necessidade de um, dois ou três *setups*. Por exemplo, se o perfil for escalonado no diâmetro (eixo X) ao longo do eixo Z, em um único sentido (crescente ou decrescente), define-se um único *setup*. Caso o perfil seja escalonado em dois sentidos ao longo do eixo Z, ou seja a dimensão X cresce até um máximo, próximo ao centro da peça, e depois decresce para a extremidade oposta, será necessário a definição de dois *setups*. Se há necessidade de garantir algum requisito de tolerância (dimensional, forma, posição, orientação, batimento ou rugosidade) nas faces esquerda e direita, será necessário a fixação em dois *setups*. É necessário analisar o perfil externo e o perfil interno de forma conjunta, para determinar o tipo de escalonamento do perfil e os relacionamentos não-geométricos das *features*.

Tem-se como saída do mapeamento baseado na decomposição orientada ao *setup*, a classificação do tipo de peça modelado (curta, intermediária ou longa), o tipo de fixação necessário (placa ou placa e contra-ponta), a definição do tipo de *blank* mais adequado (forma, diâmetro e comprimento), o número de *setups* necessários (um ou dois) e a geometria da castanha (estilos 2, 3 ou 4), descritas no capítulo 3.1.

Analisando o perfil da peça exemplo, busca-se nas extremidades esquerda e direita da peça, *features* do tipo OD e ID retilíneas (superfícies cilíndricas externas e internas) conectadas às faces retilíneas, sendo que as *features* OD e ID deverão ter uma superfície de contato de no mínimo 5 mm, para servirem de superfícies de fixação e apoio para a peça no primeiro e no segundo *setup*, se necessário.

O algoritmo (9) define quais são as *features* internas e externas que poderão ser candidatas à superfície de fixação. O modelo geométrico da castanha é avaliado, tendo maior prioridade os estilos 3, 4 e 2 (figura 3.1). O estilo 1 é usado apenas no caso de diâmetro do *blank* de 51 mm (alimentação por barras).

Na peça exemplo são identificadas quatro *features* candidatas à superfície de fixação, sendo duas externas (F2 e F20) e duas internas (F24 e F39). Já o *blank*, barra sólida (80x85 mm), poderá ser fixado com uma geometria de castanha com diâmetro de 80 mm. A geometria da castanha montada

na máquina permite a fixação de peças entre $70 \leq \phi \leq 85$, com três degraus entre os diâmetros permitidos, para fixação em superfícies externas (tipo de montagem de castanha estilo 3, fig. 3.1). Elege-se então as melhores candidatas à superfície de fixação, para o *blank* e *Setup2*, que no caso da peça exemplo serão:

- OD_FeatureEsquerda: F2 ou *Blank*
- OD_FeatureDireita: *Blank* (F20² tem rosca)
- ID_FeatureEsquerda: F39
- ID_FeatureDireita: F24

Analisando-se o perfil externo da peça, através da varredura do vetor que contém a biblioteca de *features* da peça acabada, o qual procura identificar a presença de pontos de inflexão no perfil da peça, identifica-se a existência de um escalonamento crescente de F1 a F5 e de um escalonamento decrescente de F5 a F21, mostrando a necessidade de se utilizar dois *setups*.

O mesmo é feito para o perfil interno, onde tem-se a presença do menor diâmetro na extremidade esquerda e do maior diâmetro na extremidade direita, classificando o perfil como sendo escalonado em apenas um sentido, sendo necessário um único *setup* para usiná-lo.

Identifica-se assim o número de *setups* necessários, no caso dois *setups*, as possíveis superfícies de fixação disponíveis na peça semi-acabada (três) e na peça bruta (duas).

A seguir é realizado o mapeamento baseado na decomposição geométrica, onde são identificados os grupos de *features* de projeto que serão usinados em cada *setup* da peça acabada (*setup* esquerdo ou direito) e da peça bruta (*setup* direito ou esquerdo).

Para a peça bruta, no caso de uma barra sólida, tubo ou pré-furo, o lado esquerdo é igual ao lado direito. Na peça exemplo, a peça bruta é uma barra sólida, disponível no estoque. Seu diâmetro e comprimento são definidos em função do sobrematerial necessário, após definição das operações necessárias para gerar o perfil da peça. Supondo que o projetista defina o diâmetro da peça bruta como 85 mm e o comprimento como sendo 85 mm, deixando o sobrematerial de 2,5 mm para o maior diâmetro e para o comprimento da barra.

8.2.2 Decomposição Orientada à Geometria

Nesta atividade são agrupadas as *features* de forma que serão usinadas em cada *setup*. No caso da peça exemplo, *Setup1* e *Setup2*, pois há a necessidade de realizar os dois *setups*. O *Setup1*, fixação do *blank* e o *Setup2* fixação da peça semi-acabada (lado esquerdo ou direito) serão determinados. Assim determina-se o *main workplan*, que será constituído pelo *WorkplanSetup1* (peça bruta) e *WorkplanSetup2* (lado direito ou esquerdo da peça acabada).

As *features* de forma da biblioteca de *features* normalizada, apresentam relacionamentos geométricos e não-geométricos, como tolerâncias e referências, e estes também são considerados na decomposição baseada na geometria. Analisa-se as tolerâncias de posição, orientação, batimento,

²A *feature* F20 tem uma *feature* secundária do tipo rosca, devendo ser descartada como superfície de fixação para o *Setup2*. Assim sendo esta *feature* deve ser usinada no *Setup2*.

Algoritmo 10 Algoritmo para agrupamento de *features* no *Setup1* e *Setup2*.

InícioAgrupamentoFeaturesDoisSetups ();

GeometriaPeçaAcabada();

GeometriaBlank();

IF (*GeometriaBlank* = (*BarraCilíndrica* or *Tubo*));

THEN "*Setup Esquerdo ou Direito Admissível, Analisar GeometriaPeçaAcabada, LinhaDelimitadora e AgrupamentoEmDoisSetups*";

IF (*GeometriaBlank* = (*Fundido* or *PréUsinado*)) ;

THEN "*Assumir que GeometriaBlank Será igual à GeometriaPeçaAcabada + SobreMaterial, Visando Usinagem*" ;

forma, rugosidade e dimensional, além de referências de projeto (*datum*) para determinação do agrupamento das *features* de projeto para cada *setup*, levando em conta a minimização de erros geométricos e a cadeia de tolerâncias. Deve-se respeitar também o princípio de coincidência de referências de projeto, apoio, medição e fabricação.

Para minimizar a propagação de erros de macrogeometria deve-se agrupar as *features* de projeto para serem usinadas em conjunto, mesma fixação da peça, se estas tiverem relacionamentos definidos através de referências de projeto em relação às tolerâncias de localização, posição, batimento e orientação. Estas restrições associadas às tolerâncias são fundamentais para definir as *features* que deverão ser usinadas em conjunto ou não. O mesmo acontece para *features* secundárias, que estão associadas à uma *feature* principal, e devem ser levadas em consideração durante o procedimento de decomposição.

Por exemplo, as *features* F6 e F15, onde a F6 é referência (A) para F15 (tolerância de concentricidade) devem ser usinadas no mesmo *setup*. O mesmo ocorre com as *features* internas, F39 (referência M) e a F24 (tolerância de concentricidade).

Caso não seja possível agrupar as *features* de projeto no mesmo *setup*, deve-se utilizar o método baseado em *erros do tipo II* (capítulo 3.5), para minimizar erros de macrogeometria: duas superfícies (*features*) sendo referência uma da outra. Neste caso usa-se a referência, caso seja uma *feature* do tipo OD, como superfície de apoio (superfície de fixação) na usinagem da segunda superfície.

Usa-se o conceito da Linha Delimitadora³ (LD), que é posicionada no início e no final das *features* do tipo OD de maior diâmetro (região externa) e nas *features* ID de menor diâmetro (região interna). A LD é usada como referência para o procedimento de agrupamento de *features* de projeto, que estão antes e depois desta referência. LD é posicionada na *feature* OD de maior diâmetro e na ID de menor diâmetro da peça. Caso haja mais de uma *feature* OD ou ID com o mesmo valor de diâmetro maior e menor, respectivamente, será necessário incluir mais opções de LD para cada OD ou ID extra que tiver na peça.

Para três diâmetros externos máximos iguais, pode-se ter seis alternativas de LDs (2 (antes e depois da *feature*) * 3 (número de diâmetros iguais, *features* OD)). Se tiver, também, dois diâmetros ID, tem-se 24 combinações de alternativas de LDs para serem analisadas, e compor os grupos de *features* a serem usinadas no *Setup1* e *Setup2*.

O algoritmo (10) apresenta o início do agrupamento de *features* de usinagem para dois *setups*. As LDs deverão ser analisadas em conjunto para as regiões externa e interna. O algoritmo (11) determina as LDs e os grupos de *features* de forma que serão usinados no mesmo *setup*.

³Também denominada Superfície Delimitadora, que está associada às *features* OD de maior diâmetro e ID de menor diâmetro, sendo usada para definir os agrupamentos de *features* para o *Setup1* e *Setup2*.

Algoritmo 11 Algoritmo para determinação das linhas delimitadoras (LD) e agrupamento de *features*.**Determinação Grupos Features ();**

IF ($\text{NúmeroFeaturesODMaiorDiâmetro} = 1$)

THEN " $\text{NúmeroLD}_{\text{externa}} = 2$ e ODMaiorDiâmetro será a *feature* externa escolhida para agrupamento à esquerda e à direita (região externa)";

ELSE " $\text{NúmeroLD}_{\text{externa}} = 2 * \text{NúmeroFeaturesODMaiorDiâmetro}$ e teremos várias candidatas para agrupamento à esquerda e à direita";

IF ($\text{NúmeroFeaturesIDMenorDiâmetro} = 1$)

THEN " $\text{NúmeroLD}_{\text{interna}} = 2$ e IDMenorDiâmetro será a *feature* interna escolhida para agrupamento à esquerda e à direita (região interna)";

ELSE " $\text{NúmeroLD}_{\text{interna}} = 2 * \text{NúmeroFeaturesIDMenorDiâmetro}$ e teremos várias candidatas para agrupamento à esquerda e à direita";

SWITCH (NúmeroOD , NúmeroID)

case 1 ($\text{NúmeroOD} = 1$ e $\text{NúmeroID} = 0$);

case 2 ($\text{NúmeroOD} > 1$ e $\text{NúmeroID} = 0$);

case 3 ($\text{NúmeroOD} = 1$ e $\text{NúmeroID} = 1$);

case 4 ($\text{NúmeroOD} > 1$ e $\text{NúmeroID} > 1$);

case 5 ($\text{NúmeroOD} = 0$ e $\text{NúmeroID} = 1$);

case 6 ($\text{NúmeroOD} = 0$ e $\text{NúmeroID} > 1$);

Algoritmo 12 Algoritmo para detecção de *features* secundárias, para evitar seu uso como superfície de fixação no *Setup2*.**Superfície De Fixação Feature Secundaria Setup2 ();**

IF ($\text{OD}_{\text{Esquerdo}}$, $\text{OD}_{\text{Direito}}$, $\text{ID}_{\text{Esquerdo}}$, $\text{OD}_{\text{Direito}}$ não têm *features* secundárias como rosca ou recartilhado)

THEN "Qualquer das candidatas à Superfícies de Fixação Poderão Ser Utilizadas no Setup 1";

ELSE "As Features Candidatas a Superfície de Fixação com Features Secundárias deverão ser Usinadas no Setup 2 (segundo setup)";

A peça exemplo corresponde ao *case 3*, do algoritmo (11), onde o maior diâmetro na região externa ocorre na *feature* 6 (F6) e o menor diâmetro na região interna ocorre na *feature* 39 (F39). Assim as LDs (linhas delimitadoras) serão posicionadas antes (início) de depois (final) de F6 e F39 (fig. 8.2), ou seja: se LD estiver posicionada antes de F6 e F39 não inclui as *features* delimitadoras (F6 e F39); se LD estiver depois de F6 e F39, as *features* delimitadoras são incluídas (F6 e F39).

Na peça exemplo, as opções de agrupamento de *features* que serão usinadas no mesmo *setup* são (fig. 8.2):

- região externa: lista de *features* de F1 a F5 ou de F1 a F6
- região interna: lista de *features* de F24 a F38 ou F24 a F39

Como F6 é uma superfície de referência para F15, esta deve ser incluída no segundo *setup*, deixando F6 e F15 para serem usinadas no mesmo *setup*, minimizando erros de macrogeometria.

Verifica-se também a presença de *features* secundárias em uma das superfícies de fixação externa (lado direito), fazendo com que estas *features* (primária e secundária) sejam usinadas no segundo *setup*. Assim no primeiro *setup*, usina-se o lado que não contém *features* secundárias, no caso de ocorrerem em um superfície de fixação. O algoritmo (12) sintetiza esta ocorrência.

A determinação dos grupos de *features* para cada *setup* leva em conta algumas restrições que têm um peso específico (fig. 8.2), segundo sua importância relativa, sendo elas:

- Tolerâncias de posição, localização, orientação e batimento: peso 10 e peso -10 (se a *feature* referência não está presente no grupo);

de *features* da peça, buscando identificar os relacionamentos que ocorrem entre *features* e na própria *feature*, quantificando estes relacionamentos (inter e intra *features*) que estão à esquerda e à direita da LD analisada. A análise é conduzida varrendo a *feature* até LD, do início para o final do vetor que contém a biblioteca de *features*, e vice-versa. A opção que obter o maior índice, após somatório, será escolhida como o grupo de *features* por *setup*.

A seguir é apresentado o cálculo deste índice associado as duas LDs colocadas antes e depois da *feature* F6 na peça exemplo (fig. 8.2), correspondendo ao *case 3* do algoritmo (11):

case 3 (LD em F6: região externa)

$$\text{de F1 ao final de F6 (inclui F6): } F_{1a6} = \sum_{i=1}^n \frac{F_i}{n} = \sum_{i=1}^6 \frac{0,8+0,8-0,8+0,8+0,8+0,8-10}{6} = -1,13$$

$$\text{de F1 ao final de F5 (exclui F6): } F_{1a5} = \sum_{i=1}^n \frac{F_i}{n} = \sum_{i=1}^5 \frac{0,8+0,8-0,8+0,8+0,8}{5} = 0,48$$

Assim a melhor opção, segundo o método, é a opção de F1 a F5, para o primeiro grupo de *features* da região externa. Logo o segundo grupo será constituído pelas *features* de F6 a F20. Em F6 há um *datum* (A) para uma tolerância de concentricidade. Por este motivo aparece a penalidade (-10), pois a *features* de tolerância não está presente neste grupo.

Fazendo a mesma análise para as *features* internas, tem-se que a melhor opção de agrupamento de *features* internas é agrupar todas as *features*, de F22 a F39, em um mesmo *setup*. Não há relacionamento de tolerâncias entre as *features* internas e externas.

A peça bruta é do tipo barra sólida. Tem-se uma *feature* secundária na *feature* F20, rosca, no lado direito da peça. Assim a F20 deverá ser usinada no *Setup2*. As *features* internas deverão ser usinadas no *Setup2*, pois a peça deve ter um escalonamento decrescente da direita para esquerda, para facilitar a usinagem interna da peça.

Abaixo é apresentado o *workplan*, que descreve o número de *setups* da peça (*Setup1* e *Setup2*), ordem de *setups*, a lista de *features* de projetos associadas a cada *setup* e as superfícies de fixação:

#1=workplan principal (#2,#3);

#2=workplan blank (#4);

#3=workplan peça_acabada (#5);

#4=Setup1 (F1-F5, fixado no diâmetro 85 mm);

#5=Setup2 (F6-F22 (*features* externas) e F23-F39 (*features* internas), fixado em F2);

8.2.3 Decomposição Orientada à Operações

Tendo-se determinado o agrupamento de *features* de projeto para cada *setup*, inicia-se a terceira etapa do mapeamento de *features* de projeto para *features* de usinagem, através da decomposição baseada em operações, determinando-se os volumes de material a serem removidos da peça bruta até a acabada.

Para cada *setup*, determina-se as alternativas de operações e ferramentas associadas, para usinagem de determinada *feature* de torneamento ou de usinagem. Inicialmente, determina-se o plano de processo para a região externa, operações de desbaste e de acabamento, levando em conta a peça



Figura 8.3: Roteamento para o *Setup1* e *Setup2* da peça exemplo, relacionando as *features* de projeto (F1 - F39), sendo representado na ordem inversa.

bruta e a geometria do perfil (peça acabada) para as *features* de projeto consideradas. A seguir o procedimento é executado para a região interna da peça e por último para operações de Eixo C.

A partir da biblioteca de *features* de projeto normalizada, que tem a identificação das *features* do tipo *groove complexo*, *SplineVirtual* e *TurningFeatures*, e de um roteamento padrão (figura 6.11) para operações externas e internas, busca-se um relacionamento entre as *Machining_features* e as *Turning_features* com as operações de usinagem de acordo com o roteamento fixo definido. Em função das *Machining_features* e *Turning_features* presentes na peça, um determinado caminho no grafo de precedência de operações (figura 6.11) será seguido.

O problema da decomposição orientada à operação está relacionado à associação dos volumes de usinagem, composto por uma lista de *features* de torneamento (*Turning_features*), *groove complexo* (outra lista de *features*), *thread*, *spline*, recartilhado, face, furo, cavidade, rasgo de chaveta, entre outras *features*; e para cada um destes volumes deve-se associar as operações de usinagem e alternativas, definindo um *Machining_workingstep* ou um *Turning_workingstep*. O roteamento de operações inicia com faceamento do lado esquerdo ou direito da peça, onde será definido o Zero Peça (ZP) associado à usinagem; ou com uma operação de furação concêntrica, seguido por faceamento.

Sempre que houver um canal largo (largura ≥ 20 mm), identificado no procedimento de normalização da biblioteca de *features* de projeto, pode-se utilizar como alternativas de usinagem ferramentas esquerda e direita (zig-zag), ferramenta de sangramento ou deixar uma superfície intermediária no *Setup1* e a sua usinagem no *Setup2*, por exemplo, com a mesma ferramenta. As estratégias de movimentação de ferramenta associadas à *Groove Complexo* (canal largo), são as mais complexas de se tratar.

Em função da tolerância dimensional e da rugosidade pode-se usar a mesma ferramenta para operações de desbaste e acabamento. O sobrematerial a ser deixado de uma operação de desbaste para semi-acabamento e acabamento será obtido de tabelas de sobrematerial, e estes valores variam em função do IT (intervalo de tolerância) e da dimensão da *feature* de usinagem, associado à uma determinada operação de usinagem. Assim cada *Workingstep* terá um sobrematerial associado.

Os volumes de material a serem removidos são associados sempre que possível aos ciclos fixos do fabricante, no caso o centro de torneamento Romi Galaxy 15M com CNC Fanuc 18i-ta, já descritos no capítulo 7. Os ciclos fixos mais utilizados em operações de desbaste associado a um perfil da peça acabada, são o G71, G75, G72 e G73. Para operação de acabamento utiliza-se o ciclo fixo G70.

A figura 8.3 apresenta o roteamento da peça exemplo para o *Setup1* e o *Setup2* na ordem inversa (perfil da peça acabada para o *blank*), sendo necessário prever um sobrematerial entre uma operação e outra.

As *features* de usinagem e torneamento, designadas de Ft e Fu, são obtidas por mapeamento da visão de projeto para usinagem. Para cada *setup* (*Setup1* e *Setup2*) ter-se-á as Ft e Fu associadas aos volumes de usinagem e operações de usinagem, onde pode-se ter várias alternativas. A figura 8.4 apresenta os volumes de usinagem da peça exemplo, associados ao roteamento definido anteriormente, para os dois *setups*.

Para o *Setup1* tem-se os seguintes sub-volumes de material a serem removidos: Vol1.1, Vol1.2, Vol1.3, Vol1.4 e Vol1.5, associados ao roteamento de operações apresentado na figura 8.3. Para o *Setup2* tem-se os sub-volumes: Vol2.1, Vol2.2, Vol2.3, Vol2.4, Vol2.5, Vol2.6 e Vol2.7, associados ao roteamento de operações apresentado na figura 8.3.

Pode-se ter outros sub-volumes associados à outras operações de usinagem, definindo assim alternativas de operações/sub-volumes, ou seja, *Machining_Workingsteps* e *Turning_Workingsteps* alternativos.

Para efeito de otimização, deve-se minimizar a movimentação de ferramenta em vazio e maximizar a remoção de cavaco associados aos sub-volumes de material a serem removidos e às operações de usinagem (torneamento longitudinal ou transversal). O roteamento mais eficaz para torneamento (movimentação longitudinal) ocorre quando a dimensão $Z \geq X$. Quando $X > Z$ o faceamento (movimentação transversal) é mais eficaz. Assim pode-se reduzir as alternativas de roteamento apresentado na figura 7.6, para um roteamento com menos alternativas entre sub-volumes diferentes para faceamento e torneamento (figura 6.11), levando em conta a relação $\frac{Z}{X}$ das dimensões características do volume de usinagem.

8.3 Determinação dos *Workingsteps* de Usinagem e de Torneamento com Alternativas

Para cada *feature* de usinagem determinada e relacionada a uma operação de usinagem genérica, conforme apresentado no roteamento de peças, são associadas as alternativas de operação de usinagem, segundo a abordagem descrita nos grafo E/OU, já detalhada no capítulo 7 para cada tipo de *feature*.

No caso da peça exemplo, tem-se alternativas de operações associadas às *features* de usinagem, que são representadas por uma tabela de *Workingsteps* e por um grafo E/OU associado, elaborado a partir do roteamento de operações, para o *Setup1* e *Setup2*. A tabela 8.1 apresenta os *Workingsteps* associados a peça exemplo, descrevendo as *features* de usinagem, volumes de usinagem e operações de usinagem.

A figura 8.5 mostra o plano de processo não-linear, apresentando os *Workingsteps* (W_{ID}) através de um grafo E/OU. Por exemplo, Fu1.3 e V2.5 definem a identificação da *feature* de usinagem e do

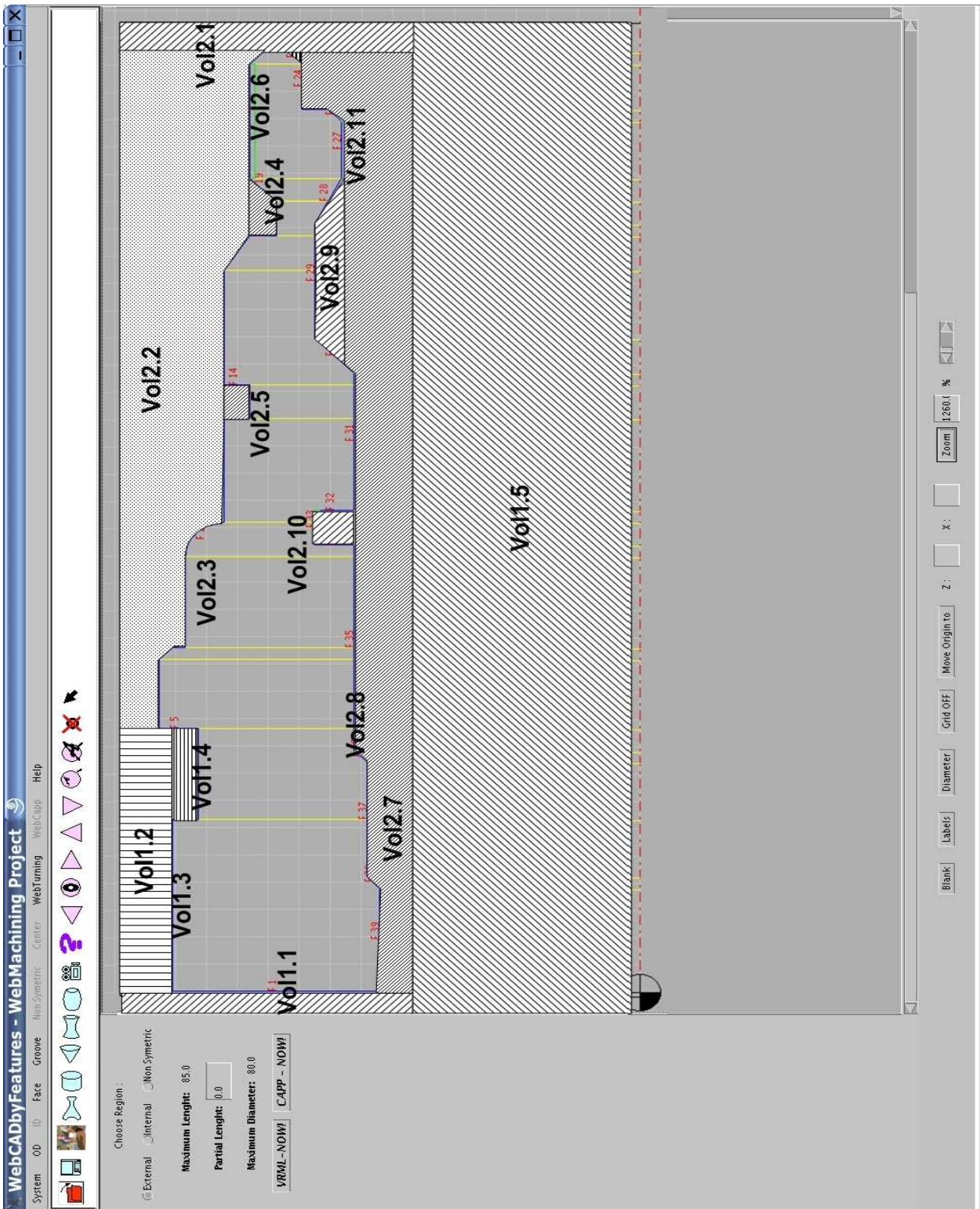


Figura 8.4: Volumes de usinagem associados à peça exemplo.

Tabela 8.1: Tabela *Workingsteps* de torneamento e usinagem (W_{ID}).

W_{ID}	Feature_Projeto (F)	Feature_usinagem (Fu)	Volume_Usinagem (V)	Operação de Usinagem
1	F1	Face - Fu1.1	Face - V1.1	Faceamento - G75
2	F2-F5	OD - Fu1.2a	OD - V1.2a	Tornear Desbaste - G71 Tipo I
3	F2-F5	OD - Fu1.2b	OD - V1.2b	Tornear Desbaste - G71 Tipo II
4	F2-F5	OD - Fu1.3	OD - V1.3	Tornear Acabamento - G70
5	F3-F5	Groove - Fu1.4a	Canal Longo - V1.4a	Tornear G71 Tipo II e G70
6	F3-F5	Groove - Fu1.4b	Canal Longo - V1.4b	Tornear Zig-Zag - Manual
7	F3-F5	Groove - Fu1.4c	Canal Longo - V1.4c	Sangramento - G72 e G70
8	F3-F5	Groove - Fu1.4d	Canal Longo - V1.4d	Tornear Intermediário - Setup2
9	F39	ID - Fu1.5	Furo - V1.5	Furação - G83
10	F23	Face - Fu2.1	Face - V2.1	Faceamento - G75
11	F6-22	OD - Fu2.2a	OD - V2.2a	Tornear Desbaste - G71 Tipo I
12	F6-22	OD - Fu2.2b	OD - V2.2b	Tornear Desbaste - G71 Tipo II
13	F6-22	OD - Fu2.3	OD - V2.3	Tornear Acabamento - G70
14	F18-20	Groove Externo- Fu2.4a	Canal Longo - V2.4a	Tornear G71 Tipo II e G70
15	F18-20	Groove Externo- Fu2.4b	Canal Longo - V2.4b	Tornear Zig-Zag - Manual
16	F18-20	Groove Externo- Fu2.4c	Canal Longo - V2.4c	Sangramento - G72 e G70
17	F13-15	Groove Externo- Fu2.5	Canal Curto - V2.5	Sangramento - G75
18	F21	Rosca - Fu2.6	Rosca - V2.6	Rosqueamento - G76
19	F24-39	ID - Fu2.7a	ID - V2.7b	Tornear Desbaste - G71 Tipo I
20	F24-39	ID - Fu2.7b	ID - V2.7b	Tornear Desbaste - G71 Tipo II
21	F24-39	ID - Fu2.8	ID - V2.8	Tornear Acabamento - G70
22	F29-31	Groove Interno- Fu2.9a	Canal Longo - V2.9a	Tornear G71 Tipo II e G70
23	F29-31	Groove Interno- Fu2.9b	Canal Longo - V2.9b	Sangramento - G72 e G70
24	F29-31	Groove Interno- Fu2.9c	Canal Longo - V2.9c	Tornear Zig-Zag - Manual
25	F35-35	Groove Interno - Fu2.10	Canal Curto - V2.10	Sangramento - G75
26	F27	Rosca Interna - Fu2.11	Rosca Interna - V2.11	Rosqueamento - G76

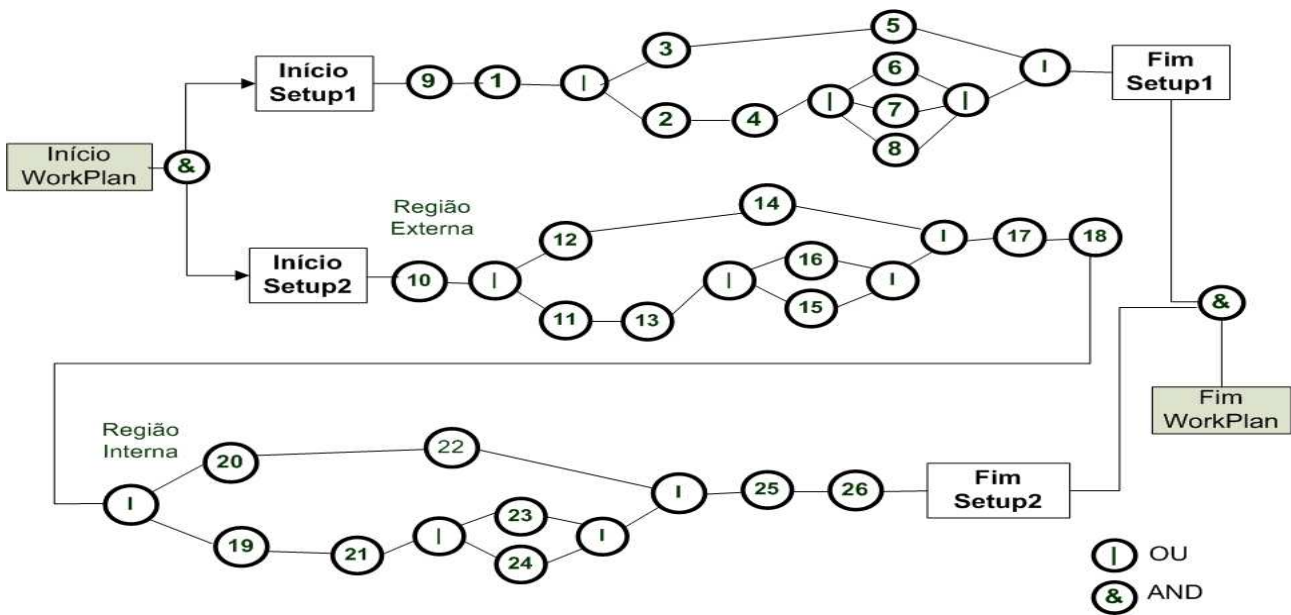


Figura 8.5: Grafo E/OU de *workingsteps* associado à peça exemplo.

volume de usinagem associado, sendo Fu1.3 a *feature 3* do *Setup1* e V2.5 o volume 5 do *Setup2*.

O *Workplan* principal (início *Workplan*) é dividido em dois *Workplans* para o *Setup1* e *Setup2*, representado no grafo E/OU. Para o *Workplan* Início *Setup1* tem-se a representação da sequência de *Workingsteps* (tabela 8.1). Para *Workplan* Início *Setup2*, tem-se os *Workingsteps* de região externa e de região interna, mostrando a sequência de operações de usinagem e suas alternativas. Este grafo E/OU é montado a partir das informações da tabela 8.1, permitindo a apresentação do plano de processo não-linear.

8.3.1 Macroplanejamento: Algoritmos e Estruturas para Determinação do Roteamento

São determinados os roteamentos para o *Setup1* e *Setup2*, realizado através do algoritmo macroplanejamento, que define a partir do roteamento padrão o plano de usinagem de cada operação associada aos *Workingsteps*, conforme o algoritmo (13).

As informações associadas às operações de acabamento são geradas antes que o desbaste, trabalhando-se na ordem inversa, do perfil final da peça para o *blank*. Ao final do procedimento, determina-se a ordem correta, ou seja, do *blank* para a peça acabada (desbaste para acabamento). Cada região de usinagem (região externa, região interna e Eixo C) tem as operações que são reconhecidas e determinadas através do algoritmo (14), que sintetiza a geração do plano de processos para cada região da peça. Por exemplo, um *PlanoRegiãoInternaDesbaste* associado a uma peça poderá ter um plano para *groove* complexo (desbaste e acabamento) seguido por um plano para *Spline Virtual* (desbaste e acabamento).

Algoritmo 13 Estrutura para algoritmo macroplanejamento.

MacroPlanejamento();
RoteamentoPadrão();
PlanoRegiãoInternaAcabamento();
PlanoRegiãoInternaDesbaste();
PlanoRegiãoExternaAcabamento();
PlanoRegiãoExternaDesbaste();
PlanoFuraçãoCorrelatasAxial();
PlanoFuraçãoCorrelatasRadial();
PlanoFresamentoAxial();
PlanoFresamentoRadial();

Algoritmo 14 Algoritmos para atividade de macroplanejamento de operações de usinagem.

<p> RoteamentoPadrão(); <i>análiseTolerânciaDimensional();</i> <i>análiseRugosidade();</i> <i>análiseTolerânciasFormaPosição();</i> <i>definiçãoSobreMaterial();</i> </p> <hr/> <p> PlanoRegiãoInternaAcabamento(); <i>planoAlargador();</i> <i>planoRosqueamento();</i> <i>planoTorneamentoAcabamento();</i> <i>planoFaceamentoAcabamento();</i> <i>planoPerfilamentoAcabamento();</i> <i>planoGrooveComplexoAcabamento();</i> <i>planoSplineVirtualAcabamento();</i> </p> <hr/> <p> PlanoRegiãoExternaAcabamento(); <i>planoRosqueamento();</i> <i>planoRecartilhamento();</i> <i>planoTorneamentoAcabamento();</i> <i>planoFaceamentoAcabamento();</i> <i>planoPerfilamentoAcabamento();</i> <i>planoGrooveComplexoAcabamento();</i> <i>planoSplineVirtualAcabamento();</i> </p> <hr/> <p> PlanoRegiãoInternaDesbaste(); <i>planoGrooveSimples();</i> <i>planoGrooveComplexo();</i> <i>planoFuraçãoAxial();</i> <i>planoTorneamentoDesbaste();</i> <i>planoFaceamentoDesbaste();</i> <i>planoPerfilamentoDesbaste();</i> <i>planoGrooveComplexoDesbaste();</i> <i>planoSplineVirtualDesbaste();</i> </p>	<p> PlanoRegiãoExternaDesbaste(); <i>planoGrooveSimples();</i> <i>planoGrooveComplexo();</i> <i>planoCortePeça();</i> <i>planoTorneamentoDesbaste();</i> <i>planoFaceamentoDesbaste();</i> <i>planoFaceamentoDesbaste();</i> <i>planoPerfilamentoDesbaste();</i> <i>planoGrooveComplexoDesbaste();</i> <i>planoSplineVirtualDesbaste();</i> </p> <hr/> <p> PlanoFuraçãoCorrelatosAxial(); <i>planoFuraçãoAxial();</i> <i>planoAlargadorAxial();</i> <i>planoRosqueamentoMachoAxial();</i> </p> <hr/> <p> PlanoFuraçãoCorrelatosRadial(); <i>planoFuraçãoRadial();</i> <i>planoAlargadorRadial();</i> <i>planoRosqueamentoMachoRadial();</i> </p> <hr/> <p> PlanoFresamentoRadial(); <i>planoFresamentoRadialHelicoidal();</i> <i>planoFresamentoRadialCoordenadaCilindrica();</i> <i>planoChanfrarEscareadorRadial();</i> </p> <hr/> <p> PlanoFresamentoAxial(); <i>planoFresamentoAxialCircular();</i> <i>planoFresamentoAxialCoordenadaPolar();</i> <i>planoChanfrarEscareadorAxial();</i> </p>
---	---

8.3.2 Microplanejamento: Algoritmos e Estruturas para Determinação das Operações de Usinagem

Os algoritmos e estruturas concebidos para o microplanejamento das operações de usinagem são sintetizados nos algoritmos (15) e (16), que apresentam a relação de métodos para gerar o planejamento detalhado de cada operação de usinagem, determinada pelo macroplanejamento. Cada operação de usinagem é vista como uma classe (POO), que tem associada diversos métodos para executar as atividades de planejamento do processo.

Para cada operação determinada no macroplanejamento, que gerou um plano de operação de usinagem, faz-se um microplanejamento da operação determinando o perfil necessário para remoção de material (sobrematerial), condições de usinagem, estratégias de movimentação de ferramenta, cálculo de tempos e custos e código G. Por exemplo, um microplanejamento para operações de usinagem em uma *Spline Virtual* poderá utilizar um ciclo fixo G71 tipo II ou G72 para desbaste, seguido de G70 para acabamento.

8.3.3 Montagem dos *Turning_workingsteps* e *Machining_workingsteps*

As operações de usinagem são associadas às *features* de usinagem ou torneamento (volumes de usinagem), constituindo assim os *Machining_workingsteps* e os *Turning_workingsteps*, respectivamente. As *features* de usinagem e torneamento podem ser associadas a uma ou mais operações de usinagem, e cada par defini um *Workingstep*⁴ para esta associação. Pode-se ter várias alternativas de operações associadas à uma *feature* de usinagem, definindo alternativas de *Workingstep* para dada *feature*. Assim uma mesma *feature* de usinagem (volume a ser removido) pode ter uma relação de um para “n” operações (vetor), definindo uma relação de cardinalidade de 1 para “n”, sendo sua estrutura apresentada no algoritmo (17).

8.4 Determinação e Representação da Seqüências de Operações por Setup

Para cada *setup*, *Setup1* e *Setup2*, deve-se montar uma matriz de precedência associada a um grafo de precedência de *features* de usinagem e determinar a melhor seqüências de usinagem, baseado nos algoritmos descritos no item anterior.

Para operações de torneamento, a precedência está associada ao roteamento de operações adotado. Este roteamento é baseado em restrições associadas à tolerâncias (macrogeometria, rugosidade e dimensional), tecnologia de usinagem e custo; e foi determinada na atividade de mapeamento de *features* focada em *features* de torneamento. *Features* de Eixo C, associadas às operações de furação e fresamento axial e radial, tem mapeamento 1 para 1 (1=>1).

Após a conclusão do mapeamento têm-se disponível os *Workingsteps*, que levam em conta as alternativas de operações para remover determinado sub-volume de material da peça. Baseado no

⁴*Workingstep* será a designação genérica para *Machining_workingstep* ou para um *Turning_workingstep*.

Algoritmo 15 Estruturas para microplanejamento de operações de usinagem.**OperaçãoAlargadorRadial());**

gerarPerfilFuroAlargado();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG85();

OperaçãoFuraçãoRadial());

acharFeatureFuro();
 gerarPerfilFuroRadial();
 OselecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG87();

OperaçãoRosqueamentoMachoRadial());

gerarPerfilFuroMachoRadial();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGM60M29();

OperaçãoRecartilhado());

acharFeatureRecartilhado();
 gerarPerfilRecartilhado();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG75();

OperaçãoTorneamentoAcabamentoInternoExterno());

OperaçãoPerfilamentoAcabamentoInternoExterno());
OperaçãoFaceamentoAcabamentoInternoExterno());
OperaçãoGrooveComplexoAcabamentoInternoExterno());
OperaçãoSplineVirtualAcabamentoInternoExterno());

acharFeaturesAcabamento();
 gerarPerfilAcabamento();
 especificarOperaçãoAcabamento();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG70();

OperaçãoAlargadorAxial());

$Diâmetro_Furo \leq Diâmetro_Limite_Barra_Mandrilar$;
 gerarPerfilAlargador();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG85();

OperaçãoGrooveComplexoDesbasteInternoExterno());

OperaçãoSplineVirtualDesbasteInternoExterno());
 gerarPerfilGrooveComplexo();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG71TipoII();
 gerarCodigoGCicloFixoG72();
 gerarCodigoGManual();

OperaçãoTorneamentoDesbasteInternoExterno());

acharListaFeaturesTorneamento();
 gerarPerfilTorneamento();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG71TipoI();
 gerarCodigoGCicloFixoG71TipoII();
 gerarCodigoGManual();

OperaçãoPerfilamentoDesbasteInternoExterno());

acharListaFeaturesUnidirecional();
 gerarPerfilTorneamento();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG73();
 gerarCodigoGManual();

OperaçãoFaceamentoDesbasteInternoExterno());

acharListaFeaturesTorneamento();
 gerarPerfilFaceamento();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGCicloFixoG72();
 gerarCodigoGCicloFixoG75();

OperaçãoFresamentoRadialHelicoidal());**OperaçãoFresamentoAxialCircular());**

acharFeatureUsinagem();
 gerarPerfilFresamento();
 selecionarFerramentas();
 calcularCondiçõesUsinagemOtimizadas();
 calcularTempoCusto();
 gerarCodigoGManual();
 calcularTempoCusto();
 gerarCodigoGManualG07.1();

Algoritmo 16 Estruturas para microplanejamento de operações de usinagem.

<p>OperaçãoChanfrarEscareadorRadial(); OperaçãoChanfrarEscareadorAxial(); <i>acharFeatureUsinagem();</i> <i>gerarPerfilFresamento();</i> <i>selecionarFerramentas();</i> <i>calcularCondiçõesUsinagemOtimizadas();</i> <i>calcularTempoCusto();</i> <i>gerarCodigoGManual();</i></p> <hr/> <p>OperaçãoFresamentoRadialCoordenadaCilíndrica(); <i>acharFeatureUsinagem();</i> <i>gerarPerfilFresamento();</i> <i>selecionarFerramentas();</i> <i>calcularCondiçõesUsinagemOtimizadas();</i> <i>calcularTempoCusto();</i> <i>gerarCodigoGManualG07.1();</i></p> <hr/> <p>OperaçãoFuraçãoAxial(); <i>acharMenorDiâmetro();</i> <i>gerarPerfilFuro();</i> <i>selecionarFerramentas();</i> <i>calcularCondiçõesUsinagemOtimizadas();</i> <i>calcularTempoCusto();</i> <i>gerarCodigoGCicloFixoG83();</i></p>	<p>OperaçãoFresamentoAxialCoordenadaPolar(); <i>acharFeatureUsinagem();</i> <i>gerarPerfilFresamento();</i> <i>selecionarFerramentas();</i> <i>calcularCondiçõesUsinagemOtimizadas();</i> <i>calcularTempoCusto();</i> <i>gerarCodigoGManualG12.1();</i></p> <hr/> <p>OperaçãoGrooveSimplesInternoExterno(); OperaçãoCortePeça(); <i>Largura_Canal_Curto ≤ 10 mm;</i> <i>gerarPerfilGroove();</i> <i>selecionarFerramentas();</i> <i>calcularCondiçõesUsinagemOtimizadas();</i> <i>calcularTempoCusto();</i> <i>gerarCodigoGCicloFixoG75();</i></p> <hr/> <p>OperaçãoGrooveSimplesInternoExterno(); OperaçãoCortePeça(); <i>Largura_Canal_Curto ≤ 10 mm;</i> <i>gerarPerfilGroove();</i> <i>selecionarFerramentas();</i> <i>calcularCondiçõesUsinagemOtimizadas();</i> <i>calcularTempoCusto();</i> <i>gerarCodigoGCicloFixoG75();</i></p>
--	---

Algoritmo 17 Estrutura *Workingstep*.

Workingstep();
featureDoWorkingstep : featureDeUsinagem();
operaçãoDeUsinagem : operaçãoDeUsinagem[] ();
planoDeAproximação();
planoDeSegurança();

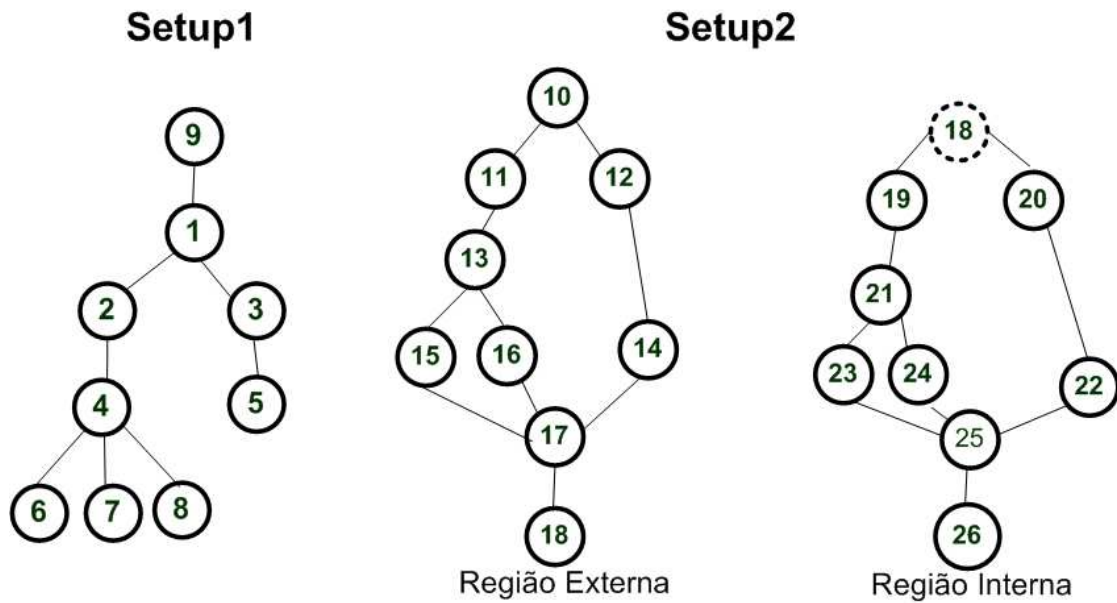


Figura 8.6: Grafo de precedência de *features* de usinagem associado aos *Workingsteps*.

macroplanejamento de usinagem monta-se um grafo de precedência de *features* ou de *Workingsteps* (fig. 8.6), e determina-se pelo algoritmo da matriz de acessibilidade (capítulo 7.3), a seqüências ótima de operações a ser seguida, incluindo também operações de fresamento e furação associadas às *features* de Eixo C, que também apresentam um roteamento pré-estabelecido em função das *features* presentes, podendo o método ser aplicado a qualquer processo de fabricação.

A matriz de acessibilidade, que é obtida da matriz adjacente, é uma forma utilizada para representar os relacionamentos entre as *features* de usinagem e torneamento (*Workingsteps*) através de uma representação matricial, que é facilmente implementado computacionalmente.

A partir do grafo de precedência de *features* de usinagem para o *Setup1* e *Setup2* (fig. 8.6), define-se o grafo E/OU que no caso da peça exemplo é apresentado na figura 8.5.

8.5 Estratégias para Geração de Trajetórias de Ferramentas

Utiliza-se estratégias para geração de trajetórias de ferramentas baseadas nos ciclos fixos de torneamento e furação definidos para o CNC Fanuc 18i-ta, a partir dos volumes de usinagem determinados no microplanejamento e a sua associação com as operações de usinagem, definindo os *Workingsteps* de torneamento ou de usinagem, baseando-se em:

Região Interna e Externa: ferramenta de torneamento

1. *Features* primárias (OD, ID e faces) e transição (*round*, chanfro e filete): lista de *features* de projeto que serão usinadas (*Turning_workingstep*);
2. *Feature* primária: *groove complexo* (*Turning_workingstep*);
3. *Feature* primária: *groove simples* (*Machining_workingstep*);

4. *Feature* primária: *splinevirtual* (*Machining_workingstep*);
5. *Feature* secundária: rosca e recartilhado (*Machining_workingstep*).

Eixo C: ferramenta rotativa axial ou radial

1. *Features* primárias de fresamento: canal, rasgo de chaveta, face plana, *slot*, cavidade (*Machining_workingstep*)
2. *Features* primárias de furação: furos e padrão de furação (*Machining_workingstep*);
3. *Feature* secundária: furo em uma cavidade ou *slot* (*Machining_workingstep*);
4. *Feature* secundária: rosca em um furo (*Machining_workingstep*);
5. *Feature* secundária: escareado em um furo (*Machining_workingstep*).

Workingsteps: definem as operações sobre as *features* de usinagem que irão gerar as *features* de projeto, sendo mapeados na maioria da vezes para ciclos fixos.

Roteamentos padrões (pré-definidos): *workplan* não-linear

1. Ferramentas de torneamento: furar + facear + torneiar + sangrar + recartilhar + rosquear;
2. Ferramentas rotativas: fresar + furar + escarear + alargar + rosquear.

Poliforma: vértice de todos os pontos da peça bruta e da peça acabada.

Sobrematerial: definir sobrematerial para cada operação baseado em tabelas de sobrematerial.

Ciclo fixo: cada ciclo fixo tem estratégia própria de movimentação de ferramenta, devendo esta ser atendida.

Plano de segurança da peça: definido nos Eixos X e Z, sendo acrescido 5 mm para a maior dimensão da peça.

Plano de aproximação do Workinstep:

1. definido nos Eixos X e Z, sendo acrescido 5 mm para a maior dimensão associada à geometria da *feature* de usinagem do *Workingstep* considerado;
2. Pode-se associar o plano de aproximação do *Workingstep* ao plano de segurança de cada *Workingstep* ao invés da peça.

Algoritmo 18 Estratégia para movimentação de ferramenta associada aos *Workingsteps* utilizando ciclos fixos.

Posiciona no plano de segurança da peça em avanço rápido;

Desloca para o plano de aproximação do Workingstep em avanço rápido;

Chama definição do ciclo fixo de desbaste (G71 - W2);

Posiciona em avanço rápido no início do posicionamento de usinagem, já estando dentro do ciclo fixo G71;

Em avanço programado (interpolação linear ou circular) inicia-se a definição de movimentação para a ferramenta percorrer o perfil da peça, associado a feature de torneamento do workingstep;

Após definição do perfil da peça, chama-se o ciclo fixo de acabamento (G70 - W3);

Ao concluir a usinagem, a ferramenta é posicionada no plano de aproximação do workingstep (avanço rápido);

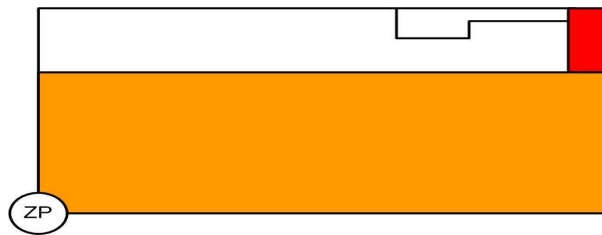


Figura 8.7: Definição Zero Peça (ZP) no Setup1, peça exemplo.

Para operações de fresamento (Eixo C) e em algumas alternativas de operações com ferramentas de torneamento, principalmente em canais largos (*grooves* complexos) necessita-se gerar toda a movimentação de ferramenta de forma manual, ou parte dela. Na usinagem de canais através de movimentação zig-zag, é necessário fazer a movimentação de forma manual. O mesmo ocorre com superfícies intermediárias, que pode necessitar de uma ferramenta esquerda no mesmo *setup* ou usar a mesma ferramenta no segundo *setup*, na usinagem da superfície intermediária.

8.5.1 Primeiro Setup: Setup1

O Zero Peça (ZP) é definido no lado esquerdo da peça (figura 8.7), na proximidade da superfície de apoio no lado da castanha, onde a peça está apoiada e fixada na castanha. A peça bruta tem as seguintes dimensões, levando em conta o sobrematerial para todas as operações:

$$Z_{\max} = 80 \text{ (peça)} + 2,5 \text{ (faceamento)} + 2,5 \text{ (faceamento)} = 85 \text{ mm}$$

$$X_{\max} = 75 \text{ (peça)} + 2,5 \text{ (torneamento desbaste)} + 1,8 \text{ (torneamento acabamento)} = 79,3 \text{ mm}$$

Tem-se um barra sólida (tarugo) em estoque de 80 mm de diâmetro. Logo o *blank* terá 80 mm de diâmetro e 85 mm comprimento.

A definição do ZP no *Setup1* é realizado através da função G54 (figura 8.7), devendo-se subtrair a dimensão 82,5 mm, quando da atribuição do zero peça, deixando um sobrematerial de 2,5 mm para o *Setup2*. Os dois ZP são definidos no lado esquerdo da peça.

Na peça exemplo as movimentações são executadas através de ciclos fixos (tabela 8.1), que devem obedecer a seqüência de definição descrita no algoritmo (18), tomando como exemplo a utilização de G71 tipo I e G70 (torneamento desbaste e acabamento), para o *Turning_workingstep* (W2 e W3), associado aos volumes de usinagem V1.2 e V1.3. Os ciclos fixos do *Setup1* são apresentados na tabela 8.1.

8.5.2 Segundo Setup: Setup2

No *Setup2* a fixação é feita na *feature* de projeto F2, e o ZP é definido na face do lado esquerdo. A definição do ZP no *Setup2* é realizada através da função G55, sendo a dimensão $Z=82,5$ mm, quando da atribuição do zero peça, deixando um sobrematerial de 2,5 mm para iniciar a usinagem. O diâmetro de fixação é de 73 mm, sendo a peça fixada no segundo degrau do modelo geométrico da castanha, que corresponde ao diâmetro de 75 mm.

A estratégia de movimentação de ferramenta é baseada apenas em ciclos fixos, como ocorreu no *Setup1*. Os ciclos fixos do *Setup2* são apresentados na tabela 8.1.

8.6 Determinação das Ferramentas de Corte

Para cada *Workingstep* faz-se a seleção das prováveis ferramentas de corte a serem utilizadas através de uma consulta ao banco de dados de ferramenta, que é constituído por dados sobre insertos, porta-ferramentas e ferramentas já montadas. Cada operação de corte tem três tabelas associadas às informações sobre a ferramenta⁵: tabela insertos, tabela porta-ferramentas e tabela ferramentas montadas.

Em função do tipo de material da peça e das classes das pastilhas Sandvik compatível com o material da peça disponível na base de dados, elege-se os possíveis candidatos a insertos. Têm-se alternativas de ferramentas para serem utilizadas em cada *Workingstep*. No banco de dados estão cadastradas 50 porta-ferramentas com geometrias diferentes, 18 tipos de diferentes insertos e 30 tipos de ferramentas rotativas, estando disponíveis para serem utilizados na fabricação das peças.

8.6.1 Ferramentas Para Torneamento Geral, Sangramento e Rosqueamento

A seleção do formato da pastilha necessária é função do ângulo de posição e das exigências de acessibilidade e versatilidade. Nas operações externas e internas para operações de torneamento geral (torneamento, faceamento, perfilamento e mergulho) têm-se cadastrado formatos de pastilhas C (80°), D (55°), W (80°), R (circular) e V (35°) e diversas geometrias de porta-ferramentas externos e internos (barras de mandrilar). Tem-se várias opções de ferramentas para corte e ranhuramento (sangramento), bem como para rosqueamento.

Para operações de torneamento geral de acabamento ($f= 0,1 - 0,3$ mm/rot e $a_p= 0,5 - 2,0$ mm), semi-acabamento ou desbaste leve ($f= 0,2 - 0,5$ mm/rot e $a_p= 1,5 - 5,0$ mm) e desbaste ($f= 0,5 - 1,5$ mm/rot e $a_p= 5 - 15$ mm), pode-se recomendar o tamanho da pastilha, caracterizado pelo tamanho do comprimento da aresta de corte, podendo variar em função do formato da pastilha de 6 a 32 mm. As pastilhas com menor comprimento de aresta, ou seja menor tamanho, devem ser usadas em operações de acabamento, já as maiores são usadas em operações de desbaste. Assim, o tamanho da pastilha é um parâmetro importante para definir a aplicação da mesma em desbaste ou acabamento. Maior o ângulo do gume da pastilha (C ou S) implica em uma aplicação mais adequada para desbaste que acabamento.

⁵Têm-se ferramentas para torneamento e ferramentas rotativas contemplando as operações de torneamento, sangramento, corte, rosqueamento, recartilhamento, furação, alargamento, escareamento, rosqueamento com macho rígido e fresamento.

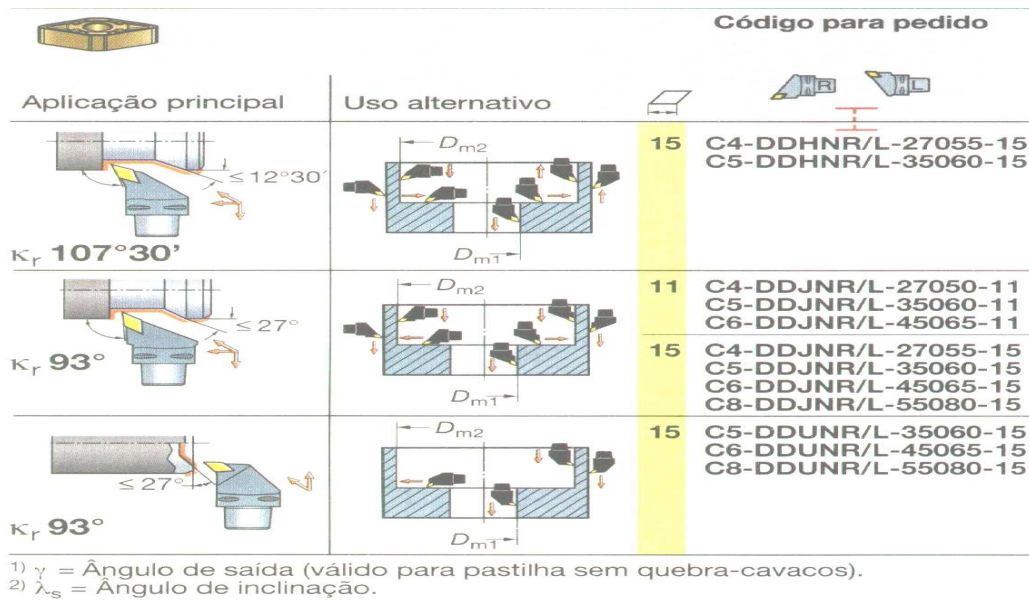


Figura 8.8: Chaves de código do porta-ferramenta DDJNR/L (SANDVIK, 2003).

Tabela 8.3: Operações de usinagem modeladas na decomposição orientada à operação.

Código_Porta-Ferramenta	Tornear_Sentido	Facear	Chanfro	κ_r (Cut-out)	Mergulho (Cut-in)
DDJNL	esquerda	+90	null	93	≤ 27
DCKNL	null	- 90	15	75	null
DWLNL	esquerda	± 90	null	63	null

Utiliza-se a chave de código adotada pela SANDVIK (2003) para designação dos insertos, sendo constituído por 12 símbolos para ferramentas de torneamento. Para o porta-ferramentas, operações de torneamento externo, adota-se a chave de código Sandvik formada por 13 símbolos, sendo que o primeiro campo da chave de código do inserto deve coincidir com o terceiro campo da chave do porta-ferramenta (formato da pastilha). O quarto campo (tipo de suporte) define o ângulo de posição κ_r , que está associado ao campo três (formato da pastilha), estabelecendo as direções e sentidos possíveis para a usinagem, permitindo a seleção da ferramenta (porta-ferramenta) para determinada geometria da peça.

As informações, relativas as direções e sentidos de usinagem estão disponíveis na base de dados de ferramentas. Por exemplo a chave de código do porta-ferramenta DDJNL e seu significado é mostrada na figura 8.8. Esta ferramenta pode executar operações de torneamento para esquerda, faceamento saindo da peça (*cut-out*, 90°) e mergulho (*cut-in*) com ângulo de abordagem para geometria da peça com conicidade $\leq 27^\circ$. Assim ter-se-á na base de dados, relativo aos campos associadas às direções de usinagem, a representação destas direções e sentidos que podem ser usinadas na peça, descrito na tabela 8.3.

Estas informações permitem a seleção automatizada das alternativas de ferramental que poderão ser utilizadas por determinado *Workingstep*. Um exemplo de consulta à base de dados em linguagem SQL é:

```
SELECT código_porta_ferramenta FROM tabela_porta_ferramenta WHERE mergulho ≤ 25 AND
torneamento_sentido = esquerda AND cut_out ≤ 90
```

Neste caso serão selecionados todos os códigos de porta-ferramentas que apresentem ângulo de mergulho (cut-in) ≤ 25 , torneamento_sentido = esquerdo e κ_r (cut-out) ≤ 90 . No caso o porta-ferramenta DDJNL seria selecionado como uma das opções de ferramental. O resultado da seleção de ferramentas são armazenadas em uma lista (vetor) contendo as ferramentas que poderão ser utilizadas neste *Workingstep*.

Como exemplo pode-se utilizar a *feature* de usinagem Fu2.9, região interna (figura 8.4), associada a reentrância, canal largo (*groove complexo*), onde uma possível barra de mandrilar (porta-ferramenta) tem o seguinte código com chave da ferramenta: A16R-SVULC11-ER.

No caso de operações internas é necessário, inicialmente, pesquisar se o diâmetro da barra (segunda posição do código) é menor que o diâmetro da peça, no caso é 32 mm. A profundidade da cavidade, reentrância, não pode ser superior à profundidade máxima de corte que a ferramenta permite, pra evitar interferência e choque com a peça. No caso a profundidade máxima de usinagem será de 5 mm (diâmetros 50-40 mm).

A seguir verifica-se as direções e ângulos de abordagem necessários. No caso tem-se uma conicidade à esquerda (cut-out) e à direita (cut-in) da reentrância de 45° , sendo necessário selecionar uma ferramenta que execute uma operação de perfilamento com capacidade de acessar esta geometria. Para a seleção do código da ferramenta, a seguinte consulta deve ser feita ao sistema de gerenciamento de base de dados MySQL:

```
SELECT código_barra_mandrilar FROM tabela_barra_mandrilar WHERE diâmetro_barra ≤ 32 AND
mergulho ≤ 45 AND torneamento_sentido = esquerdo AND cut_out ≤ 45 AND profundidade
≥ 5
```

As ferramentas a serem utilizadas são selecionadas a partir do ângulo de posição necessário e das exigências de acessibilidade e versatilidade (SANDVIK, 2003). Deve-se selecionar o maior ângulo de ponta adequado à pastilha (operação de desbaste), levando em conta a resistência e a economia, quando o requisito de acessibilidade e versatilidade não são prioritários, ao contrário do que ocorre em uma operação de perfilamento, com *cut-in* e *cut-out*.

Para a peça exemplo, considerando os *workingsteps* W₁₁ (Fu2.2a) e W₁₃(Fu2.3), opta-se por pastilhas D, W ou V. Não há restrição com relação ao ângulo de abordagem, pois não há *cut-in* ou *cut-out* para os *Workingsteps* considerados, que irão utilizar-se de estratégias de movimentação de ferramenta baseadas nos ciclos fixos G71 Tipo I e G70. Escolhe-se sempre que possível o porta-ferramenta com κ_r próximo de 90° (menor força), podendo-se escolher também a mesma pastilha para a operação de desbaste e de acabamento.

Para os *Turning_workingsteps* W₁₁ (Fu2.2a) e W₁₃(Fu2.3), peça exemplo, têm-se as seguintes alternativas de porta-ferramenta:

desbaste: DWLNL 2020K06 (usinagem média)

acabamento: DDJNL 2020K15 (acabamento)

perfilamento: DVJNL 2020K16

Caso uma das alternativas de ferramental já esteja montada na torre da máquina, esta será a ferramenta selecionada. Assim seleciona-se o porta-ferramenta DWLNL 2020K06, que está montada na posição sete da torre de ferramentas do Galaxy 15M, para ser utilizada nas operações de acabamento, usando a pastilha WNMG060412PF4025; e o porta-ferramenta DWLNL 2020K06, que está montada na posição três da torre de ferramentas do Galaxy 15M, para ser utilizada nas operações de desbaste, usando a pastilha WNMG060408PM4025 sendo recomendado pela SANDVIK COROGUIDE, <http://www.coroguide.com/CuttingDataModule/CDMMainMenu.asp>, (material peça ANSI/SAE 1025, pastilha classe 4025 e geometria pastilha) as seguintes faixas de condições de usinagem :

acabamento: $a_p = (0,4-1,5 \text{ mm})$; $f = (0,15-0,5 \text{ mm/rot})$.

desbaste: $a_p = (0,5-3 \text{ mm})$; $f = (0,15-0,5 \text{ mm/rot})$.

A partir destes dados em conjunto com os dados associados ao tempo e custo, é aplicado o algoritmo de otimização das condições de corte para cada *Workingstep*.

8.7 Determinação dos Modelos de Tempos e Custos

Os modelos de tempos e custos são elaborados para cada *Workingstep*, sendo calculados para cada estratégia de movimentação de ferramenta associada a um ciclo fixo ou movimentação manual (sem uso de ciclo fixo) previstos no *Workplan* relativo ao *Setup1* e *Setup2*.

Como exemplo será analisado o volume de material Vol2.2 e Vol2.3, a serem removidos da peça bruta no *Setup2*, através de uma operação de desbaste (Vol2.2) e outra de acabamento (Vol2.3). São utilizados os ciclos fixos G71 (desbaste) e G70 (acabamento), determinando-se assim as *features* de torneamento Fu2.2 e Fu2.3 (tabela 8.1), não tendo previsão de operações alternativas, *Workingsteps*, para estas *features* de torneamento. Têm-se apenas alternativas de ferramental que poderão ser utilizados, em função das ferramentas disponíveis no banco de dados, e já definidas anteriormente, através de uma lista de ferramentas associada ao *Workingstep*.

O modelo matemático adotado para a estratégia de usinagem dos ciclos fixos G71 e G70, é o modelo multi-passe para tempo de corte de torneamento longitudinal (equações 7.30-C.38).

Para cada passe de usinagem é necessário calcular o tempo de usinagem associado, devendo-se fazer o somatório de todos os passes de usinagem ao longo da área/volume de usinagem, para determinar o comprimento total de usinagem e calcular o tempo total de usinagem.

A fim de facilitar e simplificar este cálculo, para cada *Workingstep*, é utilizado na determinação do tempo total de usinagem sobre determinado volume de usinagem, a equação 8.1 baseada na área de usinagem (2D) a ser removida, associada à poliforma fechada, onde a profundidade de corte para os “*n*” passes de desbaste é constante:

$$t = \frac{A}{N * a_p * f} * 60[s] \quad (8.1)$$

A = área associada à *feature* de torneamento [mm²]

N = rotação da árvore [rpm]

a_p = profundidade de corte [mm]

f = avanço [mm/rot]

Para demonstrar a validade da equação 8.1, a mesma será utilizada para calcular o tempo de corte de desbaste longitudinal de uma *feature* de usinagem retangular com 50 mm de comprimento por 10 mm de largura. Assumindo que $a_p = 2$ mm, ter-se-á cinco passes de usinagem (n). Pela formulação tradicional:

$$t = \frac{\pi |X(Z_2 - Z_1)|}{500 * V_c * f}, N = \frac{1000 * V_c}{\pi * D}, Z_2 - Z_1 = 50, D = 2X$$

$$\therefore t_i = \frac{2X * 50}{f * N * D} \therefore t = \sum_{i=1}^5 t_i \therefore t = \frac{250}{f * N} * 60$$

Pela equação (8.1):

$$t = \frac{50 * 10}{N * 2 * f} \therefore t = \frac{250}{N * f} * 60$$

Este cálculo também é uma boa aproximação quando se trabalha com V_c constante, ao invés de rotação constante, onde é necessário a variação da rotação em função do diâmetro, devendo-se utilizar a equação (C.3) para calcular a rotação média, entre o maior e o menor diâmetro (cotas em X) da área considerada, para manter V_c constante.

Para operações de perfilamento associado ao n -ésimo passe de desbaste e ao passe $n+1$ de acabamento é executado apenas um passe de usinagem. Assim deve-se calcular o tempo associado ao comprimento total de usinagem, somando-se cada contribuição de retas, cones e arcos para o cálculo final do comprimento total de corte. Nas operações de desbaste, este valor deve ser somado ao corte longitudinal ($n-1$ passes) calculado através do procedimento da área. Estes cálculos são boas aproximações, não sendo exatos, ao contrário dos comprimentos de corte calculados no item 7.7.2.

Deve-se ainda calcular, de forma estimada, as parcelas de tempo associadas à movimentação rápida (tempos improdutivos), deslocamento em vazio da ferramenta. Para esta estratégia de movimentação de ferramenta, o deslocamento de ferramenta em vazio está associado ao avanço rápido G00 com velocidade de 18 m/min (eixo X) e de 24 m/min (eixo Z). Tem-se que o avanço rápido longitudinal para o ciclo fixo G71 é calculado como:

$$t_{G00} = \frac{\sum_{i=1}^n Z_i}{24000} * 60$$

Para o exemplo acima, tem-se cinco passes e o comprimento de cada passe é de 50 mm, resultando em um $t_{G00} = 0,63$ s. Já o tempo de usinagem, para $f=0,5$ mm/rot e $N=2000$ é igual a 15s. Pode-se desprezar a parcela de tempo associado a G00, por ser bem menor (5%) do que o tempo de usinagem total. Desta forma, para facilitar o cálculo do tempo total de usinagem para cada *Workingstep*, que é constituído pelo tempo principal de usinagem + tempos secundários de usinagem associados a G00, será utilizada a abordagem descrita no algoritmo(19):

Para a peça exemplo, são calculadas as áreas das *features* de torneamento (desbaste e acabamento) Fu2.2 e Fu2.3, para cada a_{ps} (a_p de acabamento) considerado. Por exemplo, o sobrematerial $a_{ps} = 0,3$ mm para a operação de acabamento, tem-se:

$$A_{Fu2.2} = 56 * 5 + 4 * 50 + 11 * 40 + 15 * 3 = 970 \text{ mm}^2 \therefore t_{w2.2} = \frac{970}{N * a_p * f} * 60$$

$$A_{Fu2.3} = 0,4 * \left(49 + \frac{(80-60)+2,2}{2}\right) = 24,04 \text{ mm}^2 \therefore t_{w2.3} = \frac{20,4}{N * a_p * f} * 60$$

A partir da determinação do tempo para cada operação de usinagem, *Workingstep*, pode-se calcular os custos envolvidos na usinagem.

Algoritmo 19 Cálculo de tempos para *Workingsteps* baseado na área de usinagem da *feature* de torneamento.

calcula-se as áreas das features de usinagem ou de torneamento;

calcula-se o tempo de usinagem associado aos Workingstep

a partir da área da feature de usinagem ou de torneamento;

estima-se que os tempos secundários associados a G00 é de 5% do tempo principal de usinagem;

computa-se o tempo total de usinagem para cada Workingstep associado

à feature de usinagem ou de torneamento;

8.8 Determinação das Condições de Usinagem com Algoritmos Genéticos

A metodologia de otimização das condições de usinagem multi-passe, conforme descrito anteriormente, divide o problema em dois sub-problemas de minimização separados para operações de desbaste e acabamento, visando a otimização da profundidade de corte, avanço e velocidade de corte para “n” passes de desbaste e um passe de acabamento.

Para cada *Workingstep* (operação de usinagem + *feature* de usinagem) que está associado a um ciclo fixo, a ser executado com uma determinada ferramenta de corte, aplica-se o primeiro e segundo estágio da metodologia de otimização, determinando-se o número de passes de desbaste e acabamento (“n” + 1), as profundidades de corte de desbaste e acabamento, as velocidades de avanço para desbaste e acabamento e as velocidades de corte de acabamento e desbaste, segundo o critério de otimização utilizado (menor tempo, menor custo ou vida de ferramenta).

Para demonstrar a aplicação da metodologia será utilizado as *features* de torneamento Fu2.2a e Fu2.3b da peça exemplo, e as operações de usinagem usando o ciclo fixo G71 tipo I e G70, respectivamente.

8.8.1 Determinação das Propriedades, Coeficientes e Constantes do Modelo de Otimização

Utiliza-se as seguintes faixas de condições de usinagem, restrições, propriedades, constantes e coeficientes associadas ao material da peça (aço 1025) e ferramenta de corte para operações de acabamento com ciclo fixo G70; e operações de desbaste leve com ciclo fixo G71 tipo I. Define-se o mesmo porta-ferramenta (*DWLNL2020K-06*) e as pastilhas *WNMG060408PM4025* para desbaste e *WNMG060412PF4025* para acabamento.

1. **GEOMETRIA:** $D_1 = 69$ mm; $D_2 = 80$ mm; $A_{FU2.2a} = 970\text{mm}^2$; $A_{FU2.3} = 20,4\text{mm}^2$; $a_p = 14,5$ mm; $L = 60,1$ mm
2. **DESBASTE:** $a_{pL} = 0,5$ mm; $a_{pU} = 3$ mm; $f_{L} = 0,15$ mm/rot; $f_{U} = 0,5$ mm/rot; $V_{cL} = 265$ m/min; $V_{cU} = 405$ m/min; $\kappa_r = 95^\circ$; $r_\epsilon = 0,8$ mm
3. **ACABAMENTO:** $a_{pL} = 0,4$ mm; $a_{pU} = 1,5$ mm; $f_{L} = 0,15$ mm/rot; $f_{U} = 0,4$ mm/rot; $V_{cL} = 265$ m/min; $V_{cU} = 405$ m/min; $\kappa_r = 95^\circ$; $r_\epsilon = 1,2$ mm
4. **POTÊNCIA GALAXY 15M:** Potência = 11 KW; rendimento = 0,9 (limite operações de desbaste)

Material de trabalho			Recomendação de dados de corte			Material de trabalho			Recomendação de dados de corte		
Padrão nacional AISI/SAE			Velocidade de corte (vc): 315 m/min			Padrão nacional AISI/SAE			Velocidade de corte (vc): 230 m/min		
Denominação: 1025 Dureza: 150 HB			Rotação do eixo (n): 1699 - 1255 rpm			Denominação: 1025 Dureza: 150 HB			Rotação do eixo (n): 1241 - 915 rpm		
Classe de pastilha/geometria: 4025 Conventional			Taxa da remoção do metal (Q): 38 cm ³ /min			Classe de pastilha/geometria: 4025 Conventional			Taxa da remoção do metal (Q): 322 cm ³ /min		
Parâmetros (escolha fn, hex ou hm)			Tempo por passe (Tc): 0.14 min			Parâmetros (escolha fn, hex ou hm)			Tempo por passe (Tc): 0.12 min		
Potência líquida (Pc): 1.8 kW			Altura máxima do perfil (Rt,Rmax): 9.4 μm			Potência líquida (Pc): 11 kW			Altura máxima do perfil (Rt,Rmax): 39.1 μm		
Ângulo de posição: 95 °			Rugosidade média (Ra): 1.91 μm			Ângulo de posição: 95 °			Rugosidade média (Ra): 10.28 μm		
Raio de ponta (re): 1.2 mm			Rugosidade média (Rq,RMS): 2.08 μm			Raio de ponta (re): 0.8 mm			Rugosidade média (Rq,RMS): 11.30 μm		
Espessura máxima do cavaco (hex): 0.30 mm/r						Espessura máxima do cavaco (hex): 0.50 mm/r					
Espessura média de cavaco (hex): 0.22 mm						Espessura média de cavaco (hex): 0.43 mm					
Espessura média de cavaco (hm): 0.12 mm						Espessura média de cavaco (hm): 0.43 mm					
Profundidade do corte (ap): .4 mm						Profundidade do corte (ap): 2.8 mm					
Diâmetros usinados (Dm1,Dm2): 59 80 mm						Diâmetros usinados (Dm1,Dm2): 59 80 mm					
Comprimento axial do corte (lz): 80.01 mm						Comprimento axial do corte (lz): 60.1 mm					
Max RPM: 5000			Tempo de vida das ferramentas: (nap): 30.0			Max RPM: 5000			Tempo de vida das ferramentas: (nap): 30.0		
			Número de passes: 213						Número de passes: 259		

Figura 8.9: Páginas HTML da Sandvik utilizadas para pesquisar as condições de usinagem para operações de acabamento e desbaste.

5. **VIDA FERRAMENTA:** $T_L = 15$ min; $T_U = 60$ min (será utilizada uma Vida T de 30 min)
6. **PRESSÃO ESPECÍFICA DE CORTE:** $K_{c0,4} = 2000$ N/mm²
7. **RUGOSIDADE:** $R_a = 100\mu m$ (desbaste) ; $R_a = 2\mu m$ (acabamento)
8. **CUSTO INSERTO POR GUME DE CORTE:** R\$ 10,00
9. **CONSTANTES DE TAYLOR E COEFICIENTES:** não estão disponíveis, sendo a velocidade de corte calculada, para determinada vida, a partir dos dados disponibilizados pela Sandvik (figura 8.9), obtido diretamente de seu banco de dados na URL <http://www.coroguide.com/CuttingDataModule/CDMMainMenu.asp>.

8.8.2 Primeiro Estágio

Gera-se um conjunto de profundidades de corte entre os limites máximos e mínimos, para as operações de desbaste e acabamento. Como o *Workingstep* de desbaste está associado a um ciclo fixo G71, deve-se utilizar a mesma profundidade de corte para todos os “n-1” passes de desbaste longitudinal de torneamento e o enésimo passe de desbaste, perfilamento. As tabelas 8.4⁶ e 8.5⁷ apresentam as condições de usinagem otimizadas e o tempo de corte para cada passe de desbaste e acabamento, usando como critério de otimização a vida de ferramenta para 30 minutos, onde os dados são obtidos diretamente da base de dados da SANDVIK (fig. 8.9). Caso o critério fosse baseado em custo ou

⁶Porta-ferramenta DWLNL2020K-06 e inserto WNMG060408PM4025.

⁷Porta-ferramenta DWLNL2020K-06 e inserto WNMG060412PF4025.

Tabela 8.4: Condições de usinagem para passe simples em operações de acabamento, considerando um comprimento de corte de 60,1 mm (L).

passo (j)	a_p (mm)	V_c (m/min)	f (mm/rot)	Tempo usinagem (s)	Ra (μm)
1	0,4	315	0,3	0,14	1,91
2	0,5	305	0,3	0,15	1,91
3	0,6	300	0,3	0,15	1,91
4	0,7	295	0,3	0,15	1,91
5	0,8	290	0,3	0,15	1,91
6	0,9	285	0,3	0,15	1,91
7	1,0	285	0,3	0,16	1,91
8	1,1	285	0,3	0,16	1,91
9	1,2	285	0,3	0,16	1,91
10	1,3	285	0,3	0,16	1,91
11	1,4	285	0,3	0,16	1,91
12	1,5	285	0,3	0,16	1,91

tempo, seria necessário calcular a velocidade de corte para várias vidas de ferramenta e determinar o menor tempo ou o menor custo para cada caso.

Calcula-se para cada a_p o avanço (f) correspondente, levando em conta as restrições do modelo, de acordo com as equações apresentadas no item C.8.3. Deve-se calcular a potência de corte, através da equação 8.2 definida pela SANDVIK (2003), ao invés de se utilizar k_f e os coeficientes μ e ν :

$$P_c = \frac{V_c * a_p * f * K_{c0,4}}{60 * 10^3} \left[\frac{0,4}{f * \sin \kappa_r} \right]^{0,29} \quad (8.2)$$

Em função da geometria do inserto determina-se o comprimento efetivo da aresta de corte (l_a), o qual está associado com a profundidade de corte através da relação $a_p = \sin \kappa_r * l_a$, sendo: $l_a = 0,4 * d$ (pastilha R); $l_a = 2/3 * d$ (pastilha S e C); $l_a = 1/2 * d$ (pastilhas T, D, K); $l_a = 1/4 * d$ (pastilhas E, D).

Utiliza-se as recomendações da SANDVIK (2003) para determinação dos avanços máximos em relação ao raio de arredondamento da ponta do inserto, devendo-se adotar os valores apresentados na tabela 8.6.

Determinação Condições de Usinagem Passe Simples

Os critérios de otimização para passe simples e multi-passes podem ser baseados na máxima taxa de produção (menor tempo), menor custo ou na vida da ferramenta. As recomendações de usinagem otimizadas para passe simples são determinadas através das recomendações do fabricante, no caso da Sandvik. Estas informações estão disponíveis na URL <http://www.coroguide.com/CuttingDataModule/CDMMainMenu.asp>, podendo ser consultadas *on-line* para operações de torneamento geral, sangramento, corte, rosqueamento, furação e fresamento.

Assim as condições de usinagem são otimizadas inicialmente para passe simples, segundo um dos critérios: menor custo, menor tempo ou vida de ferramenta. Determina-se as condições de usinagem utilizando algoritmos genéticos, a partir das recomendações de corte do catálogo *on-line* de

Tabela 8.5: Condições de usinagem para passe simples em operações de desbaste, considerando um comprimento de corte de 60,1 mm (L).

passo (j)	a_p (mm)	V_c (m/min)	f (mm/rot)	Tempo usinagem (s)	Potência (KW)
1	0,5	235	0,5	0,11	2,2
⋮	⋮	⋮	⋮	⋮	⋮
6	1,0	230	0,5	0,12	4
7	1,1	230	0,5	0,12	4,4
⋮	⋮	⋮	⋮	⋮	⋮
16	2	230	0,5	0,12	7,6
⋮	⋮	⋮	⋮	⋮	⋮
22	2,6	230	0,5	0,12	9,8
23	2,7	230	0,5	0,12	10
24	2,8	240	0,45	0,12	10
25	2,9	255	0,40	0,13	10
26	3,0	265	0,35	0,14	10

Tabela 8.6: Faixa de avanços máximos recomendado pela SANDVIK (2003) em relação ao raio de arredondamento da ponta do inserto.

r_ϵ (mm)	0,4	0,8	1,2	1,6	2,4
fmax (mm/rot)	0,25-0,35	0,4-0,7	0,5-1	0,7-1,3	1,0-1,8

ferramenta do fabricante.

Escolhe-se inicialmente o maior avanço possível, em função dos limites da ferramenta (recomendações do fabricantes) para determinada faixa de avanço válida para a classe de inserto utilizada, para uma a_p determinada (tabela 8.6). Utiliza-se a equação de Taylor expandida (Apêndice C), sabendo-se que para cada avanço selecionado tem-se uma velocidade de corte associada ao menor custo de produção e outra associada a maior taxa de produção.

Pelas equações para passe simples (Apêndice C):

$$T_R = t_e \frac{T_M}{t_t}, UT = T_M + T_I + (t_e \frac{T_M}{t_t}), UC = k_o T_M + k_o T_I + k_o (t_e \frac{T_M}{t_t}) + K_t (\frac{T_M}{t_t})$$

pode-se determinar as condições de usinagem para o menor custo e para o menor tempo de produção. A influência da profundidade de corte sobre a velocidade de corte para determinado avanço e vida da ferramenta é bem menor que a influência do avanço em relação à velocidade de corte para a mesma vida da ferramenta, sendo este aspecto bem refletido nas recomendações de usinagem do fabricante, no caso SANDVIK.

A relação determinante, com relação ao menor custo e menor tempo de fabricação é dado pela relação $\frac{T_M}{t_t}$, ou seja a vida da ferramenta dividida pelo tempo de usinagem (calculado através da área da *feature* de usinagem) para o *Workingstep* considerado. Quanto menor for esta relação menor será o custo, e quanto maior for $T_R (t_e \frac{T_M}{t_t})$, maior será a taxa de produção associada ao *Workingstep*, onde

t_e é o tempo de troca de ferramenta.

Se o critério adotado é uma vida de ferramenta pré-definida (15min)⁸, o procedimento de determinação das condições de usinagem para passe simples é direto, pois basta escolher o maior valor de f e a_p possível, e verificar as restrições associadas principalmente à potência de corte e força de corte (desbaste) e rugosidade (acabamento). Se necessário diminui-se f até satisfazer as restrições, ou se não for possível diminui-se a_p .

Para otimização baseada em menor custo e menor tempo, deve-se consultar o banco de dados de condições de usinagem e determinar, para um passe de usinagem padrão, qual o menor e maior valor para a relação $\frac{T_M}{t_t}$, variando a vida da ferramenta (T_M) de 15 min a 60 min. O menor valor da relação $\frac{T_M}{t_t}$ está associado ao menor custo e o maior valor está associado ao menor tempo de fabricação. Desta forma determina-se o intervalo de máxima eficiência de condições de usinagem.

A seguir verifica-se as restrições associadas principalmente à potência de corte e força de corte (desbaste) e rugosidade (acabamento). Se a potência de corte for superior a disponível na máquina (11 KW), deve-se reduzir f , já que a profundidade de corte está sendo mantida constante, e recalcular a nova V_c e as restrições.

O algoritmo (20) apresenta um fluxograma genérico para otimização das condições de usinagem, um único passe de usinagem, sintetizando todos os requisitos para este cálculo, como: banco de dados (DB) da SANDVIK, material ferramenta, critério de otimização, entre outros.

O algoritmo (21) detalha o cálculo das condições de usinagem otimizada pelo menor custo, maior taxa de produção ou para uma vida pré-determinada, aplicado em operações de desbaste e acabamento, a partir dos dados da Sandvik, levando em contas as restrições associadas às operações de desbaste e acabamento.

O método de otimização das condições de usinagem de desbaste inicia-se a partir das faixas de a_p e f recomendadas pelo fabricante, para determinada classe de pastilha, a fim de se ter uma boa formação de cavaco. Para cada a_p , associado ao intervalo permitido de a_p e ao comprimento efetivo de corte da geometria do inserto, escolhe-se o maior f possível associado à pastilha (raio de arredondamento da ponta). As restrições devem ser verificadas. Se necessário deve-se diminuir o valor de f , para satisfazer os requisitos para desbaste (potência de corte é o principal) e acabamento (rugosidade é o principal).

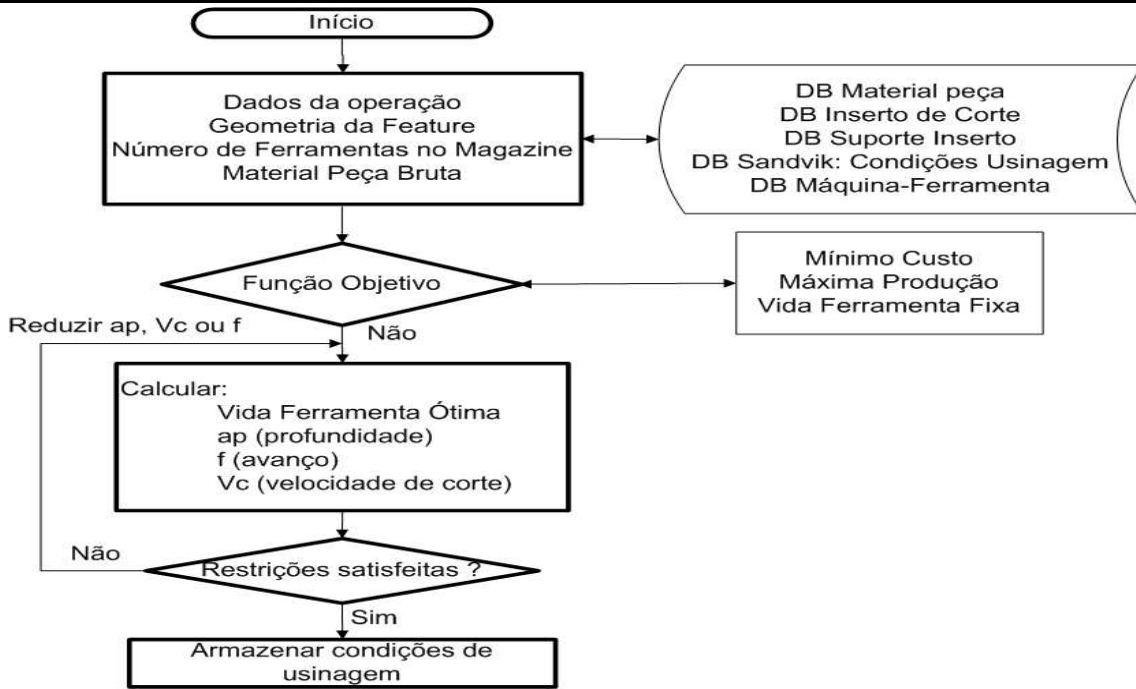
Com o critério de otimização definido e os valores máximos de a_p e f selecionados, calcula-se a seguir a velocidade de corte (V_c) otimizada para o menor custo ou menor tempo de produção, que pode ser avaliado pela relação $\frac{T_M}{t_t}$, conforme descrito anteriormente, para uma a_p determinada e variando a vida de 15 min a 60 min (15, 30, 40, 50 e 60). A seguir aumenta-se a a_p em 0,1 mm e calcula-se as condições de usinagem novamente, até concluir todas as profundidades de corte definidas. As tabelas 8.4 e 8.5 apresentam uma compilação das condições de acabamento e desbaste após a aplicação dos algoritmos (20) e (21).

8.8.3 Segundo Estágio

O número de passes (inteiro) para desbaste é:

⁸A Sandvik adota em seus catálogos impressos uma vida de 15 min para recomendação das condições de usinagem. Para valores diferentes de V_c , deve-se utilizar fatores de correção fornecido pelo fabricante. No caso em questão os dados serão obtidos através da consulta *on-line* à base de dados CoroGuide.

Algoritmo 20 Fluxograma apresentando algoritmo genérico para otimização das condições de usinagem.



Algoritmo 21 Fluxograma detalhado para seleção de condições de usinagem para passe simples: a) operações de desbaste b) operações de acabamento.

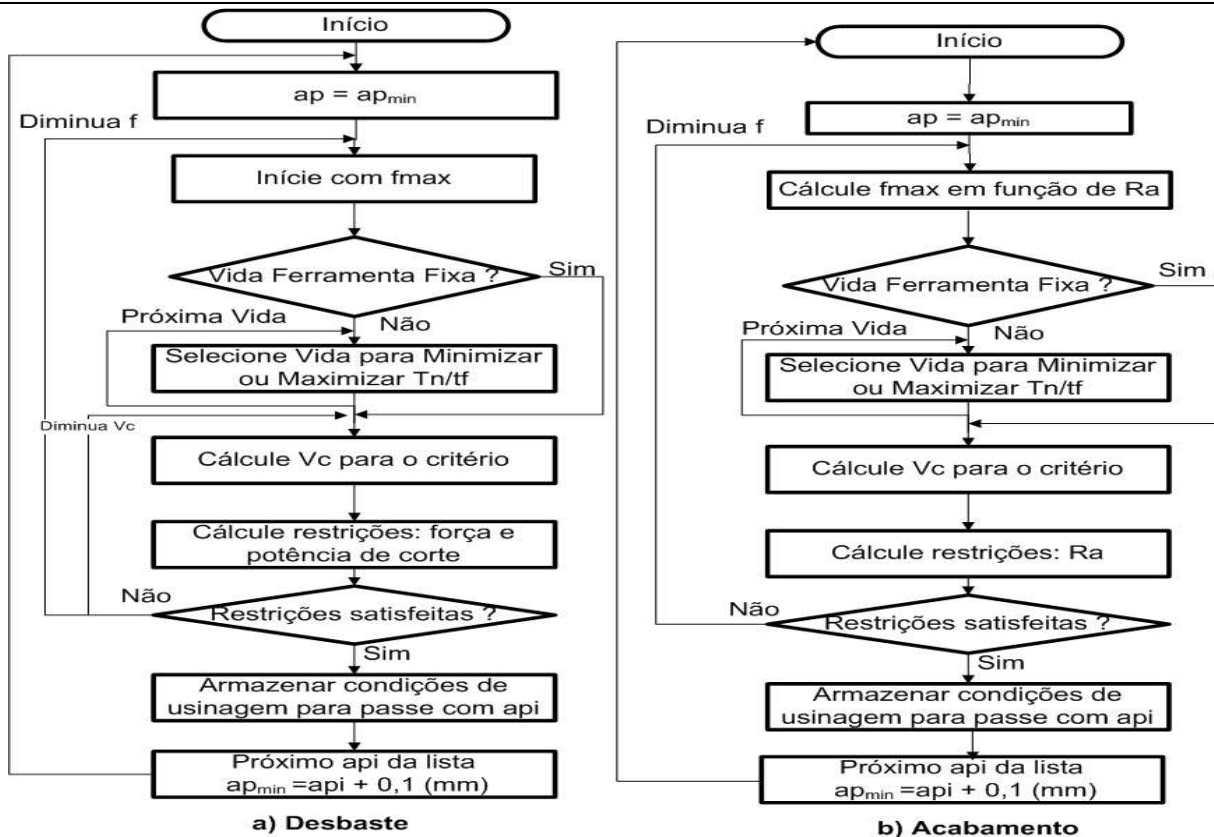


Tabela 8.7: Possibilidades de a_{ps} e a_{pr} para a peça exemplo: $a_{pr} = \frac{12,5 - a_{ps}}{4}$

$a_{ps}(mm)$	0,4	0,5	0,6	0,7	0,8	0,9	1	1,1	1,2	1,3	1,4	1,5
$a_{pr}(mm)$	3,03	3	2,97	2,95	2,92	2,9	2,87	2,85	2,82	2,8	2,77	2,75

$$\frac{a_p - a_{psU}}{a_{prU}} \leq n_i \leq \frac{a_p - a_{psL}}{a_{prU}} \therefore \frac{12,5 - 1,5}{3} \leq n_i \leq \frac{12,5 - 0,4}{3} \therefore 3,7 \leq n \leq 4,03$$

No caso determina-se o desbaste em função do sobrematerial previsto de acabamento necessário no diâmetro (1,0 mm), que já foi estimado anteriormente durante o roteamento da peça, e está dentro do intervalo calculado acima:

$$n = \frac{12,5 - 0,5}{3} = 4 \text{ passes}$$

Todos os passes de desbaste terão a mesma a_p , devido ao uso de ciclo fixo G71 e G70. Com relação ao passe de acabamento, a menor profundidade de corte a ser definida não necessita ser igual ao sobrematerial especificado para o acabamento (0,5 mm no raio), poderá ser inferior ou superior a este valor de referência. Se fosse utilizado como um único valor, o procedimento de otimização perderia o sentido.

Para $n=4$, tem-se três passes de desbaste de torneamento longitudinal, 1 passe de desbaste de perfilamento da peça e um passe de acabamento de perfilamento. Supondo a_p variando de:

Desbaste: $0,5 \leq a_{pr} \leq 3$ mm, ou seja: $a_{pr} = (0,5, 0,6, 0,7, 0,8, 0,9, \dots, 3)$

Acabamento: $0,4 \leq a_{ps} \leq 1,5$ mm, ou seja: $a_{ps} = (0,4, 0,5, 0,6, 0,7, 0,8, \dots, 1,5)$

8.8.4 População Inicial: n Passes de Desbaste + 1 Passe Acabamento

$a_p = 12,5$ mm; $a_{psU} = 3$ mm; $a_{psL} = 0,4$ mm; $a_{prU} = 1,5$ mm

$$n_{a_{psL}} = \frac{12,5 - 0,4}{3} = 4,03 \text{ e } n_{a_{psU}} = \frac{12,5 - 1,5}{3} = 3,7, \text{ logo } n = 4 \text{ (já calculado anteriormente)}$$

Tem-se uma *String* com $(n+1)$ elementos, ou seja, o cromossomo será constituído por 5 genes. Escolhe-se uma população com dez indivíduos, dez cromossomos, selecionando inicialmente os n passes de desbaste com mesma profundidade de corte a_p , seguido pela operação de acabamento, resultando nas seguintes combinação de a_{ps} e a_{pr} apresentadas na tabela 8.7.

A seguir monta-se a população de 10 cromossomos com 5 genes cada, e aplica-se o algoritmo genético para otimização das condições de usinagem para vida de ferramenta de 30 minutos. A tabela 8.8 apresenta a população inicial de cromossomos montada a partir das tabelas 8.7, 8.4 e 8.5.

A função objetivo é baseada no critério vida de ferramenta para 30 minutos, sendo o tempo de usinagem para esta vida calculado por: $T_{30} = \sum_{i=1}^n T_{a_{pr}} + T_{a_{ps}}$. Para o caso mais genérico pode-se considerar a utilização de diferentes profundidades de corte para calcular o tempo total, apesar do ciclo fixo utilizado trabalhar apenas com profundidade de corte constante.

A partir da tabela 8.8 elege-se os melhores indivíduos que são apresentados na coluna *Valor Real*, que indica a quantidade de indivíduos que tem melhor aptidão pra evolução e que irão compor a próxima população no processo de evolução.

Tabela 8.8: População inicial de cromossomos, função *fitness*, valor esperado e valor real baseado no método da roleta (probabilidade de um indivíduo ser melhor que outro).

<i>n. Cromossomo</i> (<i>i</i>)	<i>Cromossomo</i>	<i>Tempo Total</i> (<i>Tmin</i>) <i>i</i>	<i>Valor Fitness</i> (<i>Tfitness</i>) <i>i</i> = <i>Tmax</i> -(<i>Tmin</i>) <i>i</i>	<i>Valor Esperado</i> $E_i=(Tfitness)_i/\bar{T} fitness$	<i>Valor Real</i> <i>Treal</i>
1	3.03, 3.03, 3.03, 3.01, 0.4	0,7	0,01	0,27	0
2	3, 3, 3, 3, 0,5	0,71= <i>Tmax</i>	0	0	0
3	2.9, 2.8, 3, 2.9, 0,9	0,68	0,03	0,81	1
4	2.75, 2.75, 2.75, 2.75, 1.5	0,64	0,07	1,89	2
5	2.85, 2.85, 2.85, 2.85, 1.2	0,68	0,03	0,81	1
6	2.95, 2.95, 2.95, 2.95, 0.7	0,71= <i>Tmax</i>	0	0	0
7	2.82, 2.82, 2.82, 2.82, 1.2	0,68	0,03	0,81	1
8	2.8, 2.77, 2.92, 2.97,1.04	0,66	0,05	1,35	2
9	2.8, 2.8, 2.8, 3, 1.1	0,66	0,05	1,35	2
10	2.9, 2.9, 2.9, 2.9, 0.9	0,67	0,04	1,08	1

Tabela 8.9: Operadores de reprodução e *crossover* com probabilidade de 80%.

<i>Pais 1</i>	<i>n.</i>	<i>Crossover</i>	<i>Cruza</i>	<i>Pais 2</i>	<i>Local</i>	<i>Resultado</i>	<i>Ajuste</i>
2.9, 2.8, 3, 2.9, 0,9	1	Sim	6	2.8, 2.77, 2.92, 2.97,1.04	2	2.77, 2.92, 2.97, 1.04, 2.9	2.77, 2.92, 2.97, 2.8, 1.04
2.75, 2.75, 2.75, 2.75, 1.5	2	Sim	8	2.8, 2.8, 2.8, 3, 1.1	4	3, 1.1, 2.75, 2.75, 2.75	3, 2.9, 2.75, 2.75, 1.1
2.75, 2.75, 2.75, 2.75, 1.5	3	Não				2.75, 2.75, 2.75, 2.75, 1.5	2.75, 2.75, 2.75, 2.75, 1.5
2.85, 2.85, 2.85, 2.85, 1.2	4	Sim	1	2.9, 2.8, 3, 2.9, 0,9	4	3, 2.9, 0,9, 2.85, 2.85	2.9, 2.85, 2.85, 2.85, 1.05
2.82, 2.82, 2.82, 2.82, 1.2	5	Sim	4	2.85, 2.85, 2.85, 2.85, 1.2	2	2.85, 2.85, 2.85, 1.2, 2.82	2.85, 2.85, 2.85, 2.75, 1.2
2.8, 2.77, 2.92, 2.97,1.04	6	Sim	9	2.8, 2.8, 2.8, 3, 1.1	4	3, 1.1, 2.8, 2.77, 2.92	3, 2.83, 2.8, 2.77, 1.1
2.8, 2.77, 2.92, 2.97,1.04	7	Sim	2	2.75, 2.75, 2.75, 2.75, 1.5	3	2.75, 2.75, 1.5, 2.8, 2.77	2.75, 2.75, 2.7, 2.8, 1.5
2.8, 2.8, 2.8, 3, 1.1	8	Sim	7	2.8, 2.77, 2.92, 2.97,1.04	2	2.77, 2.92, 2.97,1.04, 2.8	2.77, 2.92, 2.97, 2.8, 1.04
2.8, 2.8, 2.8, 3, 1.1	9	Sim	5	2.82, 2.82, 2.82, 2.82, 1.2	4	2.82, 1.2, 2.8, 2.8, 2.8	2.82, 2.88, 2.8, 2.8, 1.2
2.9, 2.9, 2.9, 2.9, 0.9	10	Não				2.9, 2.9, 2.9, 2.9, 0.9	2.9, 2.9, 2.9, 2.9, 0.9

Tabela 8.10: Operador de mutação com probabilidade de 80%.

<i>n.</i>	<i>População Crossover</i>	<i>Mutação</i>	<i>Local</i>	<i>Resultado</i>	<i>Valor Fitness</i>
1	2.77, 2.92, 2.97, 2.8, 1.04	Não		2.77, 2.92, 2.97, 2.8, 1.04	0
2	3, 2.9, 2.75, 2.75, 1.1	Não		3, 2.9, 2.75, 2.75, 1.1	0,01
3	2.75, 2.75, 2.75, 2.75, 1.5	Sim	2 e 5	2.75, 2.8, 2.75, 2.75, 1.45	0,04
4	2.9, 2.85, 2.85, 2.85, 1.05	Não		2.9, 2.85, 2.85, 2.85, 1.05	0
5	2.85, 2.85, 2.85, 2.75, 1.2	Não		2.85, 2.85, 2.85, 2.75, 1.2	0
6	3, 2.83, 2.8, 2.77, 1.1	Não		3, 2.83, 2.8, 2.77, 1.1	0,01
7	2.75, 2.75, 2.7, 2.8, 1.5	Não		2.75, 2.75, 2.7, 2.8, 1.5	0,02
8	2.77, 2.92, 2.97, 2.8, 1.04	Sim	1 e 3	2.9, 2.92, 2.58, 2.8, 1.3	0,01
9	2.82, 2.88, 2.8, 2.8, 1.2	Não		2.82, 2.88, 2.8, 2.8, 1.2	0,02
10	2.9, 2.9, 2.9, 2.9, 0.9	Não		2.9, 2.9, 2.9, 2.9, 0.9	0,01

Os operadores de reprodução e *crossover* é apresentado na tabela 8.9, onde a seleção é baseada nos melhores indivíduos (cromossomos representados por *strings*) e calculados pela tabela 8.8, sendo que a população reproduzida, chamada de *Pais 1*, é usada pelo próximo operador genético, o *crossover*. Oito indivíduos da população anterior, são eleitos como *Pais 1* no operador de reprodução e apresentados na primeira coluna da tabela 8.9. É utilizado para o operador *Crossover*, uma probabilidade de 80%, ou seja, o *Crossover* não ocorrerá em duas populações. A coluna *Pais 2* é definida aleatoriamente a partir da coluna de *Pais 1*, determinando assim os indivíduos que serão utilizados no cruzamento de *Pais 1* x *Pais 2*. Um operador de ajuste é necessário para compatibilizar as profundidades de corte total e as limitações associadas às operações de acabamento e de desbaste.

A seguir é aplicado um operador de mutação à população gerada pelo operador *crossover* com uma probabilidade de 20%, e calculado o valor da função *Fitness*, recomeçando novamente o algoritmo genético. A tabela 8.10 apresenta o operador mutação. Novamente é necessário aplicar um operador de ajuste para compatibilizar as profundidades de corte total e as limitações associadas às operações de acabamento e de desbaste.

Após 100 gerações define-se através do GA qual o melhor indivíduo, ou seja, os valores para os *n* passes de desbaste e para o passe de acabamento. Para os *Workingsteps* considerados, que tem operações de usinagem baseados em ciclos fixos, com passes de desbaste com profundidade de cortes iguais, tem-se uma diminuição da quantidade de combinações.

O GA sugere como resposta uma profundidade de corte de desbaste de 2.8 mm e 1,3 mm para acabamento, bem superior ao sobrematerial de acabamento recomendado inicialmente. As condições de usinagem para desbaste são definidas na tabela 8.4, passe *j*=24. As condições de usinagem para acabamento são apresentadas na tabela 8.5, passe *j*=10.

8.9 Linearização do Plano de Processos com Alternativas (Work-plan)

Para cada *Workingstep* é calculado o custo e o tempo de usinagem usando as condições de usinagem calculadas a partir de um dos critérios de otimização que foi selecionado anteriormente. É adotado o

Tabela 8.11: Tabela *Workingsteps* para linearização *Workplan* (W_{ID}).

W_{ID}	Feature_Projeto (F)	Feature_usinagem (Fu)	Operação de Usinagem	Tempo (min)
11	F6-22	OD - Fu2.2a	Tornear Desbaste - G71 Tipo I	0,48
13	F6-22	OD - Fu2.3	Tornear Acabamento - G70	0,16

critério baseado em vida de ferramenta para 30 minutos. Caso se deseje outro critério, é necessário defini-lo antes de iniciar o CAPP.

Aplica-se o algoritmo genético para otimização da linearização dos *Workplan*, para os passes de acabamento e de desbaste para o *Workingstep* considerado. Após analisar todos os *Workingsteps*, têm-se a determinação de todos os parâmetros de usinagem para todas as operações, incluindo as alternativas presentes no *Workplan*⁹. A tabela 8.11 apresenta os tempos associados aos dois *Workingsteps* para as *features* Fu2.2a e Fu2.3, após linearização das condições de usinagem, segundo o critério de vida de ferramenta de 30 min.

A tabela 8.12 apresenta a matriz adjacente associado ao custo ou ao tempo para mudar o *setup* de um *Workingstep* para o seguinte (sucessor) de acordo com o grafo de precedência de *Workingstep* (grafo E/OU) ou de *features*. A partir desta matriz aplica-se a função objetivo calculada para o *Workplan baseado em tempo* (*TempoWorkplan* (TW)), associado a um *setup* da peça:

$$TW = \sum_{i=1}^n (TempoWorkingstep_i) + \sum_{i=1}^{n-1} (TempoSetup_{i \rightarrow i+1} * \Omega (Workingstep_i, Workingstep_{i+1})) \tag{8.3}$$

$$\Omega (x, y) = \begin{cases} 1, & \text{se } x \neq y \\ 0, & \text{se } x = y \end{cases} \tag{8.4}$$

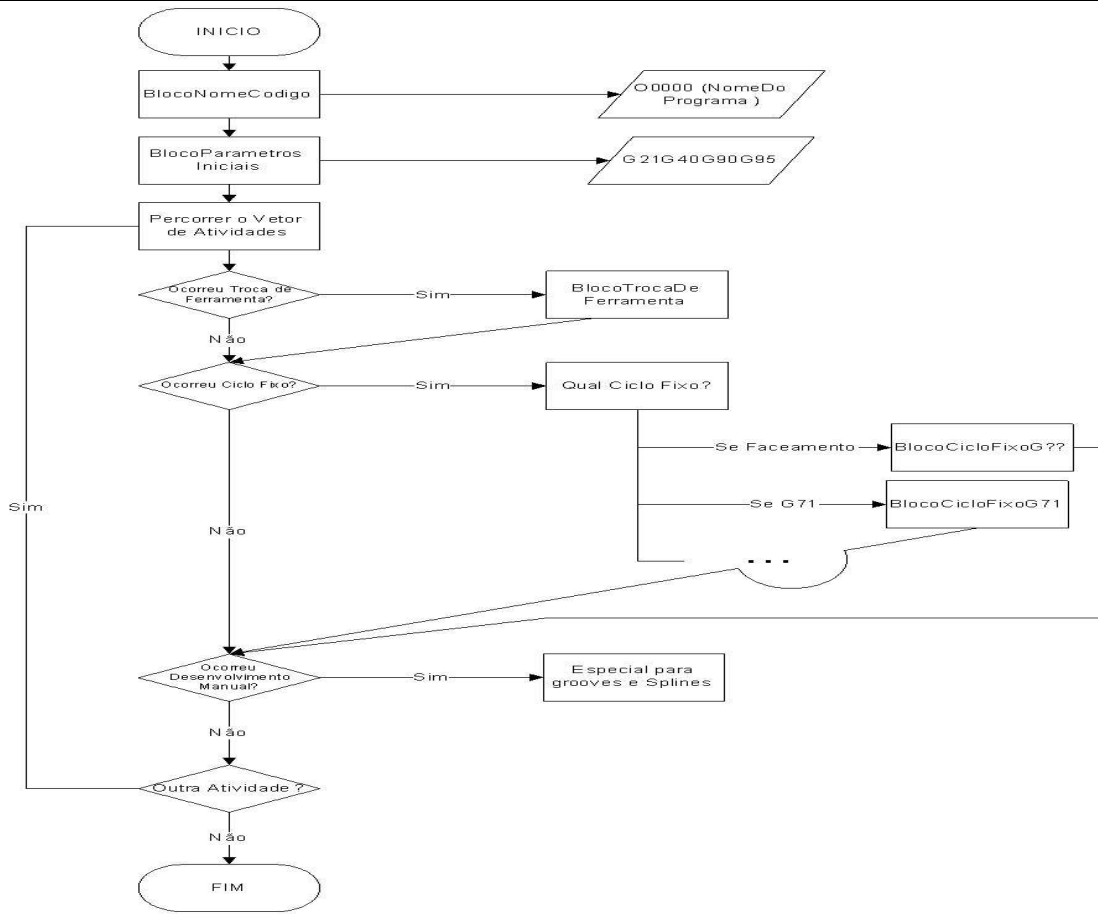
Para *Setup1* são analisados os *Workingsteps* de 1 a 9 (W1 a W9). Para o *Setup2* são analisados os *Workingsteps* de 10 a 26 (W10 a W26). Todas as ferramentas necessárias estão disponíveis na torre de ferramentas. A troca de ferramenta tem um custo de 10 unidades para ferramentas de torneamento interno e externo. Caso a mudança seja para uma ferramenta de sangramento o custo será de 15 unidades. Se a mudança for para uma ferramenta de rosqueamento o custo será de 20 unidades. Não é necessário nenhum *setup* adicional de ferramental, para a peça exemplo. A célula em branco da tabela 8.12 significa “Não Aplicável”.

Aplica-se o GA da mesma forma que no caso da otimização das condições de usinagem, realizando a linearização do plano de processo. Para a peça exemplo a sequência linearizada de *Workingstep* será:

- *Setup1*: W9 → W1 → W3 → W5
- *Setup2*: W10 → W12 → W14 → W17 → W18 → W20 → W22 → W25 → W26

⁹*Workplan* não foi linearizado até este momento.

Algoritmo 22 Fluxograma determinação de movimentação de ferramenta baseado em ciclo fixo ou movimentação manual.



8.10 Geração do Código G e Plano de Processo

O código G é montado a partir da lista de *Workingstep* do *Workplan* linearizado. Cada *Workingstep* tem associado uma *feature* de usinagem e uma operação de usinagem. Cada operação de usinagem tem associado uma ferramenta, uma tecnologia, uma estratégia e um caminho de ferramenta. Sempre que possível uma operação de usinagem estará associada a um ciclo fixo, e este terá uma parametrização especial em função do CNC utilizado na máquina-ferramenta, sendo normalmente uma função específica do fabricante do CNC. O algoritmo (22) apresenta o fluxograma de determinação do ciclo fixo ou e movimentação manual associado a um determinado *Workingstep* de torneamento.

Um ciclo fixo já tem uma definição pré-estabelecida sobre a estratégia de usinagem e a movimentação de ferramenta. Assim um *Workingstep* associa uma determinada *feature* de usinagem a um ciclo fixo, obtendo-se todos os dados geométricos necessários para movimentação de ferramenta em relação ao perfil acabado da peça e do *blank*. As informações associadas à profundidade de corte de desbaste e acabamento são incluídas na definição do ciclo fixo. Ao *Workingstep* é associado uma ferramenta, e à ferramenta a definição de velocidade de corte e avanço otimizadas. São definidos os planos de segurança e aproximação do *Workingstep*, associado ao primeiro posicionamento relativo ao ciclo fixo, antes do início do ciclo fixo.

A figura 8.10 apresenta o código G para a peça exemplo.

O3000(PECAEXEMPLO)
 N10G21G40G90G95
 N20M36
 N30M37
 N40(INICIO SETUP1)
 N50(WORKINGSTEP - FEATURE DE USINAGEM Fu1)
 N60G53
 N70G0X390Z350T00
 N80T0707(Tool DESBASTE GERAL EXTERNO)
 N90G54
 N100G96S300
 N110G92S4000
 N120M4
 N130M8
 N140(Operacao - CICLO FIXO G75 FACEAMENTO)
 N150G0X82Z114
 N160G0X77Z103.998
 N170G75X31Z102P800000Q2000R1F0.2
 N180(WORKINGSTEP - FEATURE DE TORNEAMENTO F12)
 N190G0X72
 N200(Operacao - CICLO FIXO G71-TIPO1 DESBASTE E G70 ACABAMENTO)
 N210G0X82Z112
 N220G0X72Z107
 N230G71U1R2
 N240G71P250Q310U0.2W0.2F0.2
 N250G0X65
 N260G42
 N270G1W-5
 N280G1W-15.5
 N290X70W-5
 N300W-5
 N310X72
 N320G70P250Q310
 N330(WORKINGSTEP - FEATURE DE TORNEAMENTO F13)
 N340G53
 N350G0X390Z350T00
 N360T0303(Tool SANGRAMENTO EXTERNO)
 N370G54
 N380G96S300
 N390G92S4000
 N400M4
 N410M8
 N420(Operacao - CICLO FIXO G72 GROOVE COMPLEXO E G70 ACABAMENTO)
 N430G0X75Z96.5
 N440G0X70Z86.5
 N450G72W0.6R0
 N460G72P470Q520U0W0F0.2
 N470G0Z81.5
 N480G41
 N490G1X65
 N500G1X60
 N510W5
 N520X65
 N530G70P470Q520
 N550G53
 N560G0X390Z350T00
 N570T0909(Tool DESBASTE GERAL INTERNO)
 N580G54
 N590G96S300
 N600G92S4000
 N610M4
 N620M8
 N630(Operacao - CICLO FIXO G71-TIPO1 DESBASTE E G70 ACABAMENTO)
 N640G0X26Z112
 N650G0X36Z107
 N660G71U1R2
 N670G71P680Q750U-0.2W0.2F0.2
 N680G0X46
 N690G41
 N700G1W-5
 N710G1W-12
 N720
 N730W-7
 N740X38W-14
 N750X36
 N760G70P680Q750
 N760(FIM SETUP1)
 N770(TROCA DE SETUP)
 N780G53
 N790G0X390.0Z350.0T00
 N800M00
 N810M36
 N820M37
 N830(INICIO SETUP2)
 N840(WORKINGSTEP - FEATURE DE USINAGEM Fu1)
 N850G53
 N860G0X390Z350T00
 N870T0707(Tool DESBASTE GERAL EXTERNO)
 N880G55
 N890G96S300
 N900G92S4000
 N910M4
 N920M8
 N930(Operacao - CICLO FIXO G75 FACEAMENTO)
 N940G0X82Z114
 N950G0X77Z103.998
 N960G75X31Z102P800000Q2000R1F0.2
 N970(WORKINGSTEP - FEATURE DE TORNEAMENTO F12)
 N980G0X72
 N990(Operacao - CICLO FIXO G71-TIPO1 DESBASTE E G70 ACABAMENTO)
 N1000G0X82Z112
 N1010G0X72Z107
 N1020G71U1R2
 N1030G71P1040Q1160U0.2W0.2F0.2
 N1040G0X56
 N1050G42
 N1060G1W-5
 N1070G1X58W-1
 N1080W-12
 N1090W-37
 N1100W-9
 N1110X62
 N1120X64W-1
 N1130W-6.5
 N1140X67
 N1150X70W-8
 N1160X72
 N1170G70P1040Q1160
 N1180(WORKINGSTEP - FEATURE DE TORNEAMENTO F13)
 N1190G53
 N1200G0X390Z350T00
 N1210T0303(Tool SANGRAMENTO EXTERNO)
 N1220G55
 N1230G96S300
 N1240G92S4000
 N1250M4
 N1260M8
 N1270(Operacao - CICLO FIXO G72 GROOVE COMPLEXO E G70 ACABAMENTO)
 N1280G0X68Z99
 N1290G0X63Z89
 N1300G72W0.6R0
 N1310G72P1320Q1430U0W0F0.2
 N1320G0Z52
 N1330G41
 N1340G1X58
 N1350G2X54W2R2
 N1360G1W9
 N1370X50
 N1380W3
 N1390X54
 N1400W10
 N1410X52W3
 N1420W5
 N1430X58W5
 N1440G70P1320Q1430
 N1450(WORKINGSTEP - FEATURE DE USINAGEM Fu4)
 N1460G53
 N1470G0X390Z350T00
 N1480T0101(Tool ROSCAMENTO EXTERNO)
 N1490G55
 N1500G96S300
 N1510G92S4000
 N1520M4
 N1530M8
 N1540(Operacao - CICLO FIXO G76 ROSCAMENTO AUTOMATICO)
 N1550G97S3333
 N1560M4
 N1570G0X68Z111
 N1580X63Z101
 N1590G76P010060Q0R0
 N1600G76X56.05W-12P975Q325F1.5
 N1620G53
 N1630G0X390Z350T00
 N1640T0909(Tool DESBASTE GERAL INTERNO)
 N1650G55
 N1660G96S300
 N1670G92S4000
 N1680M4
 N1690M8
 N1700(Operacao - CICLO FIXO G71-TIPO1 DESBASTE E G70 ACABAMENTO)
 N1710G0X26Z112
 N1720G0X36Z107
 N1730G71U1R2
 N1740G71P1750Q1850U-0.2W0.2F0.2
 N1750G0X50
 N1760G41
 N1770G1W-5
 N1780G1X46W-2
 N1790W-7
 N1800X44
 N1810X42W-2
 N1820W-20
 N1830
 N1840W-30
 N1850X36
 N1860G70P1750Q1850
 N1860(FIM SETUP2)
 N1870G53G40M9M5
 N1880G0X390Z350T00
 N1890M36
 N1900M37
 N1910M30(FIM DA PECA)

Figura 8.10: Código G gerado para a peça exemplo.

Capítulo 9

Projeto Colaborativo: WebCADByFeatures

Este capítulo apresenta uma descrição de uso e detalhes de implementação do WebCADbyFeatures (<http://WebMachining.AlvaresTech.com>). O sistema WebCADbyFeatures é um agente de interface com o usuário que permite o desenvolvimento de projeto colaborativo de uma peça rotacional simétrica (*features* concêntricas) e assimétricas (*features* de Eixo C) no contexto do sistema WebMachining via Internet.

9.1 Introdução

O desenvolvimento do sistema computacional WebCADbyFeatures para projeto de produto colaborativo se diferencia dos sistemas relatados na revisão de literatura, por ser voltado para modelagem no domínio de peças rotacionais, baseado em síntese por *features* de projeto (*features* concêntricas, incluindo *splines*, e *features* de Eixo C (não-concêntricas)), tendo por motivação a proposição e o desenvolvimento de um sistema integrado para CAD/CAPP/CAM que permita o projeto colaborativo via *Web*, em um contexto de Engenharia Simultânea.

O procedimento de desenvolvimento de produto na arquitetura WebMachining inicia-se na modelagem colaborativa de uma peça por *features*, onde dois ou mais agentes de projeto cooperam na modelagem da peça em 2D e 3D, num contexto de manufatura remota, utilizando a *Web* como meio de comunicação e um modelo computacional cliente-servidor (<http://WebMachining.AlvaresTech.com>).

O cliente, agente de interface WebCADbyFeatures (figura 9.1), conecta-se ao modelador de *features* neutro via *Web*, e inicia a instanciação de uma nova peça a ser modelada a partir de uma base de dados, usando uma biblioteca de *features* de forma padronizada, disponibilizada pelo sistema.

A Interface Gráfica com o Usuário (GUI) é implementada em Java e HTML. Nesta GUI, disponibilizada por *browser*, o usuário introduz as informações sobre as *features* de projeto, que irão constituir a peça em um contexto de projeto colaborativo. A seguir estes dados são encaminhados ao servidor WebMachining. Como a peça é cilíndrica, o usuário modela a peça em duas dimensões, podendo visualizá-la em 3D, através de VRML, via *Web*. Foi implementado um banco de dados em MySQL que armazena as informações sobre a modelagem de produto por *features*, contendo informações associadas às *features* de forma, *features* de material, *features* de tolerâncias e *features* tecnológicas (tratamento superficial, tratamento térmico e dados de produção). Estas informações conjugadas per-

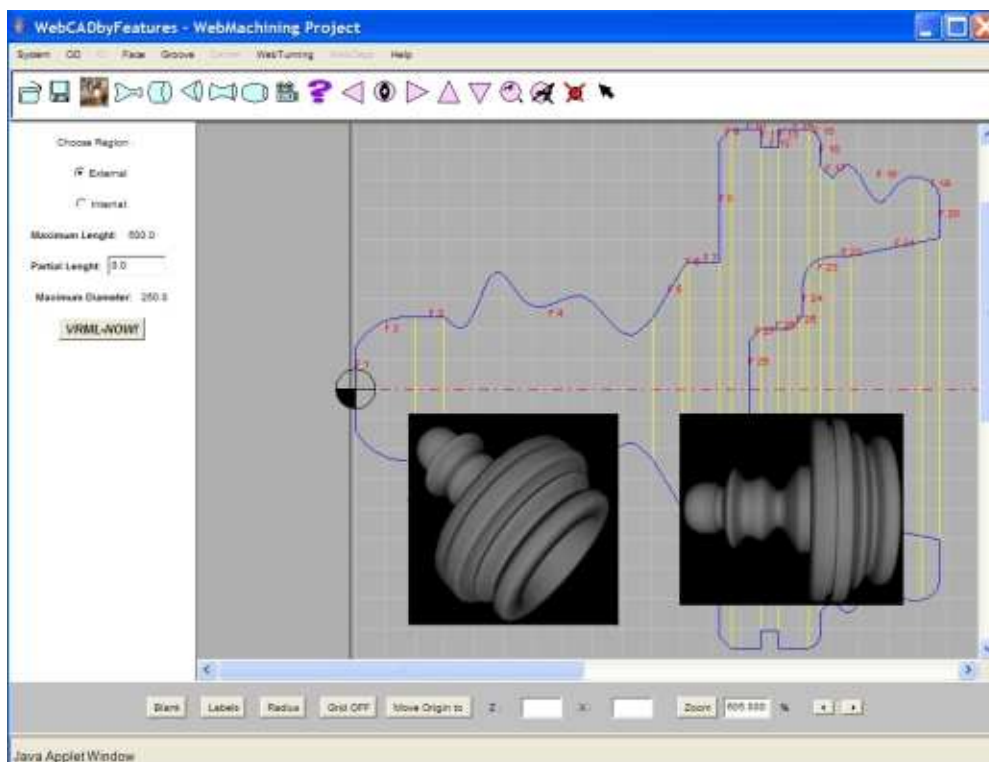


Figura 9.1: Janela principal do WebCADbyFeatures, apresentando o perfil de uma peça rotacional em 2D. A peça é constituída por 29 *features*, incluindo *features* do tipo OD, ID, Face, Groove, Splines, entre outras. É apresentado sólido VRML correspondente, gerado pelo sistema.

mitem o mapeamento das *features* de projeto para *features* de usinagem, o que é fundamental para o planejamento do processo da peça.

Depois da conclusão e validação do modelo, a peça criada é armazenada e disponibilizada para a metodologia CAPP gerar o plano de processo com alternativas para a peça, sua linearização e a representação do plano de processo linearizado baseado em STEP-NC, bem como, a geração do programa NC para um torno CNC específico, no caso o centro de torneamento Galaxy 15M da Romi (<http://video.graco.unb.br>).

A comunicação com o centro de torneamento Galaxy/CNC Fanuc 18i-ta é realizada através de uma conexão ethernet (camada física e enlace do padrão ISO/OSI), usando os protocolos TCP/IP (camadas rede e transporte do padrão ISO/OSI) associado ao protocolo de aplicação *FOCAS1/ethernet libraries* da Fanuc. FOCAS1 é uma API para desenvolvimento de aplicações usando uma estrutura de dados padronizada, para ter acesso a 300 funções do CNC (<http://webdnc.graco.unb.br>).

Após a conclusão do CAPP, a ordem de trabalho é enviada para o sistema de manufatura remota (telemanufatura), que faz sua programação disponibilizando os recursos necessários para a fabricação da peça (programa NC, ferramentas, *setup* e *blank*).

9.2 Arquitetura Multiagente Para Projeto Colaborativo

A arquitetura concebida e implementada para o sistema multiagentes (MAS) pode ser caracterizada no comportamento dos agentes como sendo Deliberativa, na organização interna como sendo do tipo

Blackboard e na arquitetura do sistema multiagentes como sendo do tipo Federativa utilizando a abordagem do Facilitador (SHEN *et al.*, 2001).

É utilizada a ferramenta computacional JATLite (*Java Agent Template Lite*), Agente Facilitador, para implementação do sistema de desenvolvimento de produto colaborativo. Utiliza-se as facilidades do JATLite para desenvolvimento do sistema de gerenciamento de sessões de projeto colaborativo, onde um agente de interface de projeto disponibiliza seu projeto aos demais agentes participantes da sessão de modelagem de produto.

O sistema de projeto colaborativo utiliza um mecanismo assíncrono de gerenciamento das sessões de modelagem colaborativa, na qual um agente de interface WebCADbyFeatures executa a modelagem da peça e compartilha seu modelo de *features* da peça com os demais agentes de interface via Agente Facilitador (AMR), que faz a intermediação da comunicação entre os agentes via KQML (*middleware*). Os agentes que recebem o modelo podem visualizá-lo e editá-lo, compartilhando novamente com o agente que iniciou o processo de modelagem colaborativa ou com outros agentes, via Agente Facilitador.

9.3 Modelagem da Informação

9.3.1 WebCADbyFeatures: Modelagem Conceitual da Informação

O sistema WebCADbyFeatures, utiliza-se da abordagem síntese por *features* de projeto, usando uma taxonomia de *features* baseada em CAM-I (1986) e no padrão ISO 10303-224 (*STEP - Standard Exchange of Product Model Data*).

O modelo conceitual da informação foi concebido através da abordagem metodológica IDEF1X (*I (Integrated Computer Aided Manufacturing) DEFINITION*). Este modelo se utiliza de uma sintaxe e semântica própria, porém sua representação gráfica é muito próxima do modelo UML e conseqüentemente da linguagem Java, o que foi um dos principais fatores para adesão a este modelo, além de ser uma metodologia completa, não somente uma representação gráfica.

A partir deste modelo IDEF1X foi gerado tanto o modelo físico da base de dados para o sistema de gerenciamento de base de dados MySQL, quanto a biblioteca de classes associadas às *features*, em Java. Este modelo de informação é dividido em domínios associados à base de dados de *features* (*Features* de Forma, *Features* de Tolerância, *Features* de Produção e *Features* de Materiais) e base de dados de tecnologia de usinagem (Biblioteca de Máquinas-ferramenta, Biblioteca de Ferramentas de corte, Biblioteca de Usinabilidade e Biblioteca de Fixação). A base de dados de *features* relaciona-se com o modelo do produto e a base de dados de tecnologia de usinagem relaciona-se com o modelo de recursos.

Utiliza-se a biblioteca de *features* de forma para planejamento de processo baseado em CAM-I, estendendo-a com a inclusão de *feature* concêntrica do tipo *spline* e de outras *features* de Eixo C. As *features* utilizadas na implementação no sistema WebCADbyFeatures foram as rotacionais concêntricas e não-concêntricas (Eixo C) mostradas no diagrama hierárquico da figura 6.13. A utilização da metodologia IDEF1X permitiu uma rápida modelagem em UML, devido a grande semelhança entre esses modelos. A figura 9.2 mostra um exemplo de diagrama de classes (UML), com uma parte da biblioteca de *features* (ÁLVARES, 2003), implementada no sistema WebMachining

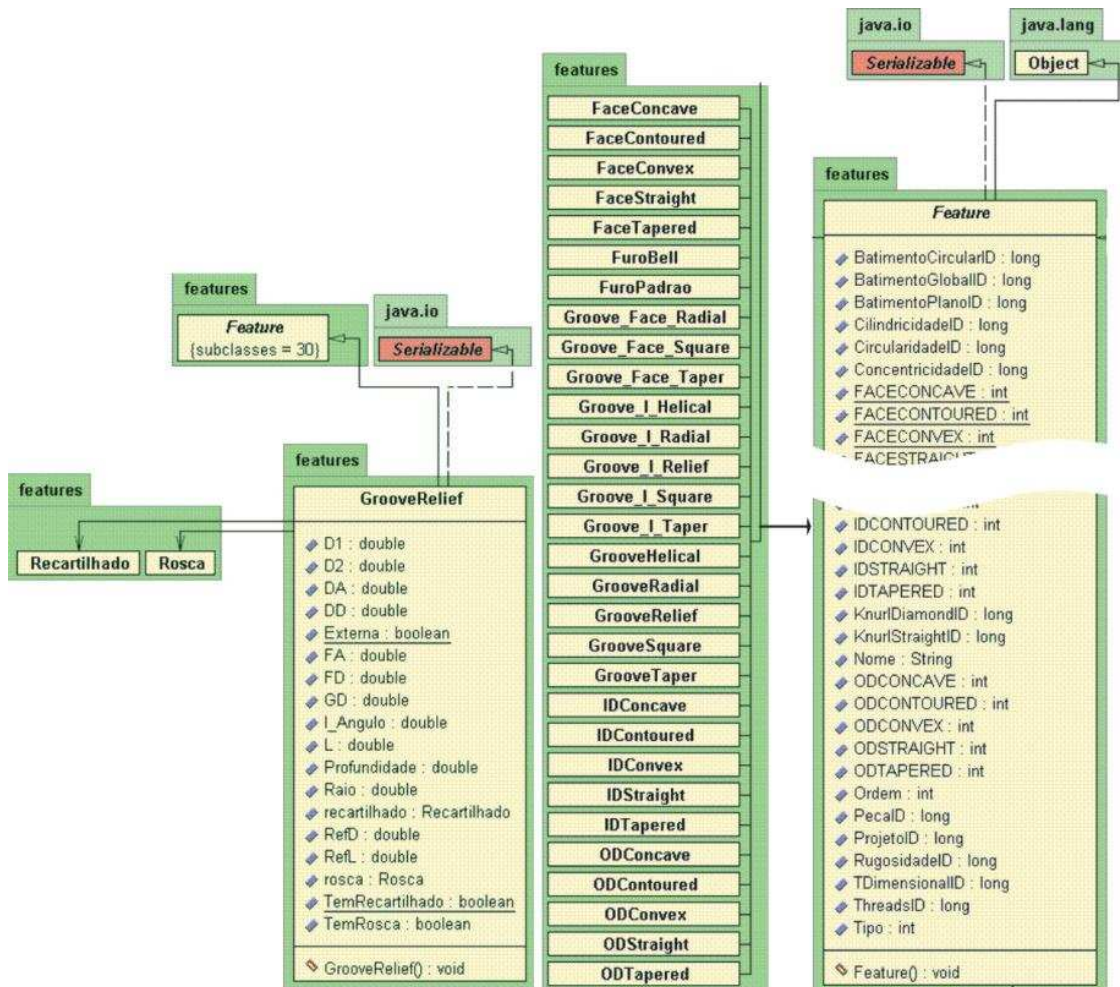


Figura 9.2: Resumo da modelagem UML da biblioteca de *features*.

(<http://WebMachining.AlvaresTech.com>).

9.3.2 WebCADbyFeatures: Modelagem Física - MySQL

Para a implementação dos modelos de dados foi selecionado o servidor de banco de dados relacional MySQL. Também foi desenvolvida uma biblioteca de classes Java com os modelos UML, para uma maior funcionalidade do banco de dados. O banco de dados foi implementado utilizando-se as tabelas INNODB, por serem transacionais, suportarem chaves estrangeiras, alta performance em ambientes multi-usuários e por terem tamanho de tabelas ilimitado (INNODB, 2004).

Optou-se pelo uso de *servlets* para fazer a conexão do applet Java WebCADbyFeatures com o banco de dados MySQL, adotando-se uma abordagem em três camadas (figura 9.3).

9.4 Sistema Computacional WebCADbyFeatures

O programa tem como entrada de dados a modelagem por *features* e outras informações necessárias (figura 6.4), e como saída o modelo de *features* da peça bruta e da peça acabada, o qual serve de entrada para o CAPP, que por sua vez tem como saída a geração do plano de processo linearizado e o

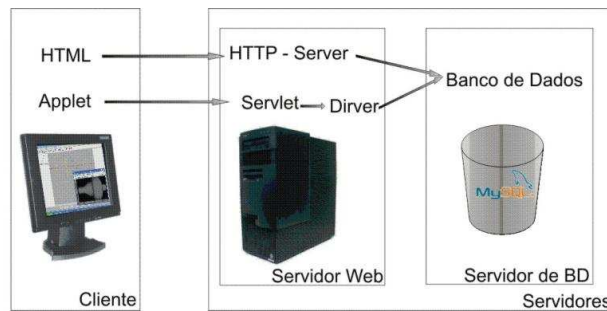


Figura 9.3: As três camadas do acesso ao banco de dados.

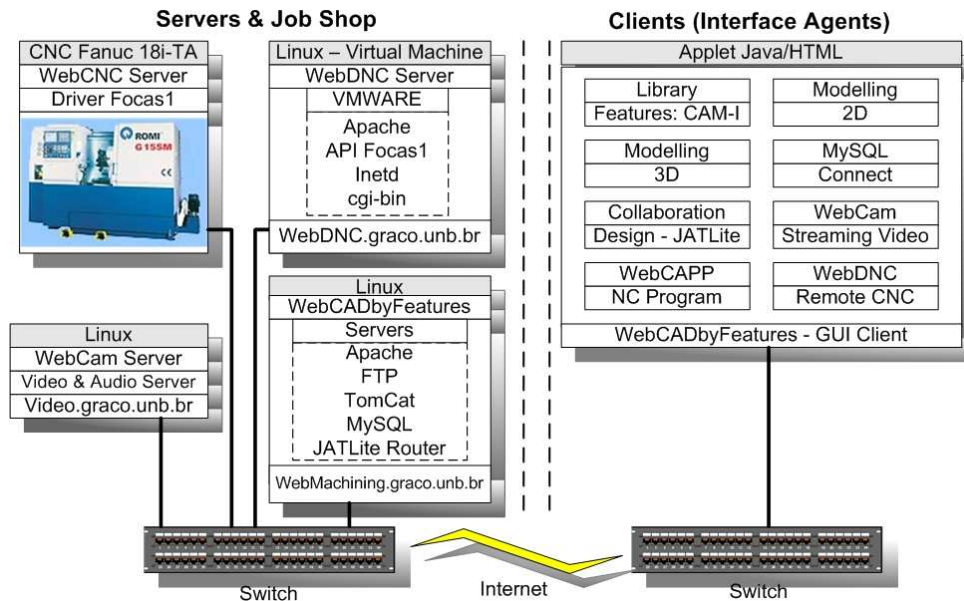


Figura 9.4: Arquitetura computacional WebMachining: módulos do sistema WebCADbyFeatures.

programa NC (Código G) da peça correspondente, conforme apresentado no diagrama IDEF0 (figura 6.4).

A figura 9.4 apresenta a arquitetura do sistema computacional WebCADbyFeatures, que é baseado em um modelo cliente/servidor, com a utilização da linguagem Java de programação em sua codificação.

No lado do cliente, a interface gráfica com o usuário (GUI) é representada por *applets*, sendo estes designados de Agentes de Interface. São implementados dois servidores:

- Servidor VRML baseado em *servlets* (TomCat): usado para geração do modelo 3D da peça em VRML, a partir do modelo de *features* de projeto da peça;
- Servidor Roteador/Facilitador JATLite: permite o gerenciamento das diversas sessões de modelagem de produto colaborativa, fazendo a coordenação da comunicação entre os vários agentes de interface WebCADbyFeatures, bem como, com os demais agentes do sistema, administrando o roteamento das mensagens entre os agentes, segurança do sistema e o registro de agentes. É implementado através do *Agent Message Router* (AMR) da arquitetura JATLite. Os agentes sempre se comunicam uns com os outros via AMR, como por exemplo para enviar um *e-mail*

(protocolo SMPT) ou o arquivo com o modelo de *features* (protocolo FTP).

O WebCADbyFeatures permite a criação e a manipulação do modelo de *features* para a peça bruta e acabada de forma colaborativa, o armazenamento dessas informações em um banco de dados MySQL, a validação do modelo e a visualização do modelo geométrico em 2D e 3D (VRML).

É constituído por uma GUI (*Graphical User Interface*) que atua como um agente de interface no sistema multiagente WebMachining, suportando modelagem colaborativa por *features*. Esta interface CAD possui menus, opções de visualização, mensagens de erro, manipulação de *features*, comunicação com o gerenciador de sessão JATLite para modelagem colaborativa, comunicação com o servidor do Banco de Dados, comunicação com o servidor VRML, monitoração do chão-de-fábrica (WebCAM), teleoperação do centro de torneamento, entre outras funcionalidades (fig. 9.4).

Os principais componentes são: a GUI na forma de *applet* Java, a biblioteca de *features*, interface gráfica 2D, Collaborative Design IPlayer Router Client, componentes de visualização 2D (primitivas gráficas como retas, *splines* e arcos) e componentes de visualização 3D (VRML). As informações referentes às *features* são manipuladas através de um sistema de gerenciamento de banco de dados.

A modelagem da peça inicia-se com o acesso do cliente à página da Web¹ (figura 9.5) onde se encontra o *applet* Java para a utilização do programa CAD. Se o usuário for cadastrado, é feito um acesso ao banco de dados e confere-se o *login* e a senha do usuário, trabalha-se *on-line* com o restante do sistema integrado CAD/CAPP/CAM (WebMachining). Pode-se optar em trabalhar de forma *off-line*, versão *demo*. Para a geração do modelo sólido 3D, optou-se pela geração do código em VRML ao invés do JAVA3D, para tornar o programa mais leve, podendo assim ser carregado mais rapidamente a partir do *browser*.

VRML é uma linguagem padrão de modelagem de mundos e objetos (avatares) virtuais para a Internet. Possui várias funcionalidades, dentre as quais funções específicas de modelagem, como extrusão, além de controle de luzes e câmeras básicas, suporte a conteúdo multimídia como textura, sons e filmes (HATMAN e WERNECK, 1996). Possui algumas entidades geométricas básicas, como cubos, cones e cilindros. Observou-se no decorrer do desenvolvimento do projeto a inviabilidade de compor estas entidades básicas em geometrias mais complexas, para formar as peças. Optou-se, portanto pela geração dos sólidos com o método de extrusão.

A navegação no sistema inicia-se após o cadastramento do usuário via PHP (*Personal Home Page*) a partir da URL <http://WebMachining.AlvaresTech.com>. A seguir o *applet* é chamado, feito seu *download* via *Web*, e automaticamente a máquina Java local inicia o *applet*. O *applet* foi desenvolvido usando o padrão de GUI AWT (*Abstract Windowing Toolkit*) e não SWING. Optou-se por AWT para permitir maior desempenho e compatibilidade com qualquer máquina Java versão 1.1, que é implementada de forma nativa na maioria dos *browsers* (netscape, iexplorer, entre outros), sem necessidade de um *plug-in* específico para uma determinada versão de Java, facilitando o uso pelo usuário do sistema.

A primeira janela oferece as opções para iniciar a modelagem da peça (figura 9.6a). Para um usuáriocomum só é possível criar um novo projeto. O usuário é na realidade um Agente de Interface.

A seguir abre-se uma nova janela (figura 9.6b) que reúne todas as informações de projeto, sendo que estas são armazenadas dentro de uma classe para ser adquirida e transmitida ao banco de dados.

¹<http://WebMachining.AlvaresTech.com>.

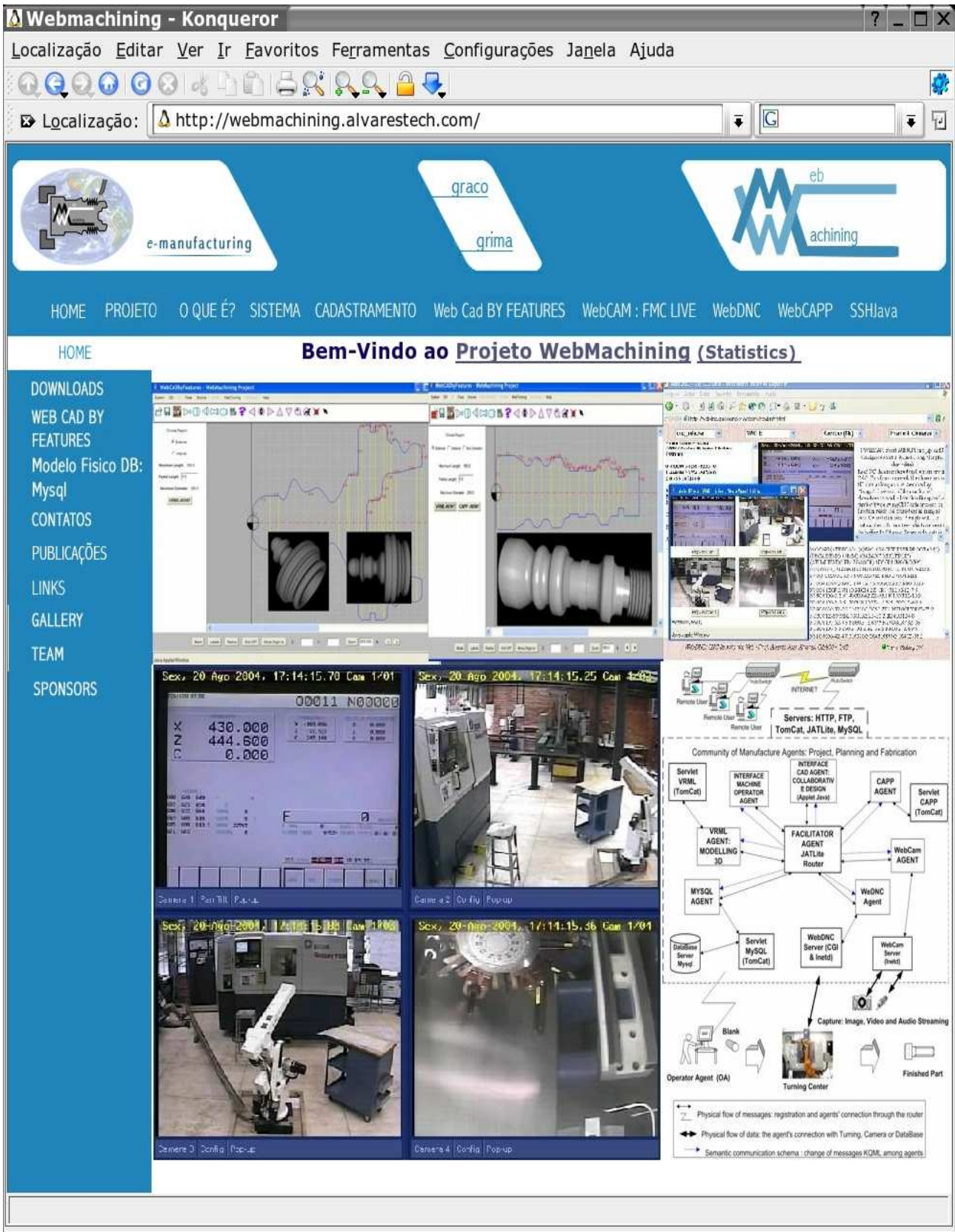


Figura 9.5: Acesso ao sistema WebCADbyFeatures (<http://WebMachining.AlvaresTech.com>).

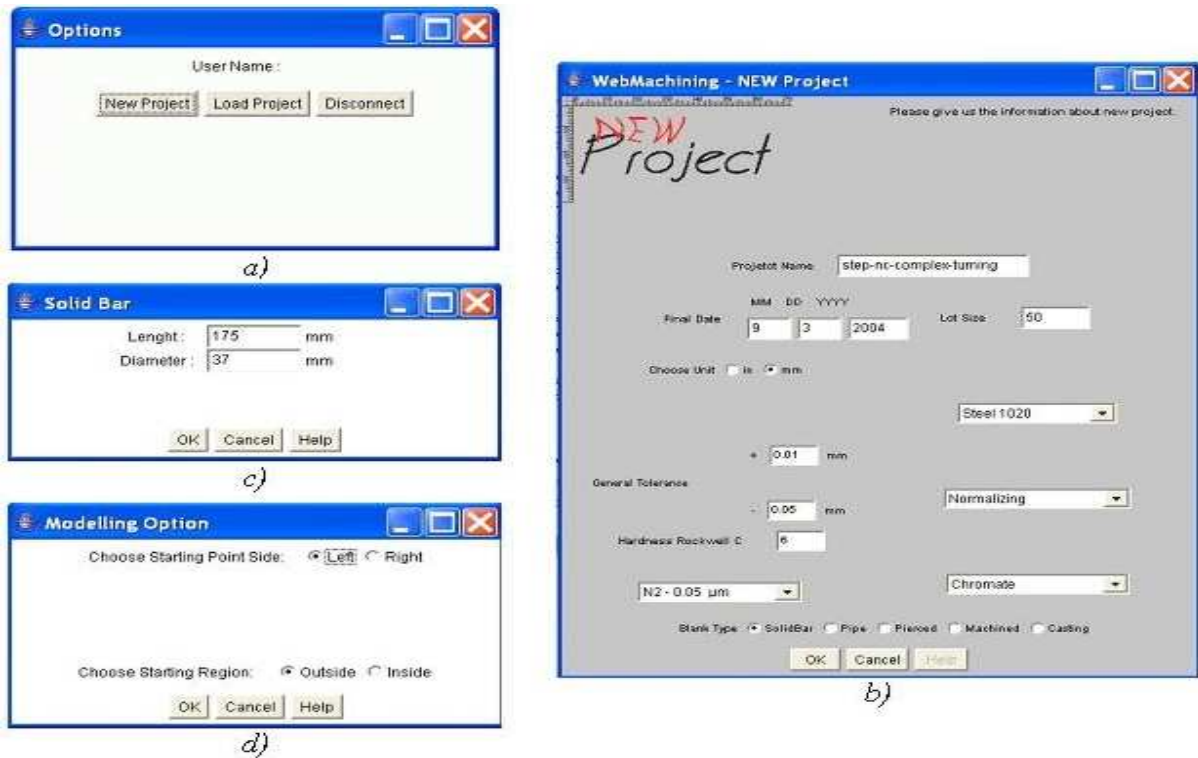


Figura 9.6: Etapas da geração da peça no CAD: a) Janela de Opções (Options); b) Janela de dados de um novo projeto (New Project); c) Janela de dados do blank barra sólida (SolidBar); d) Janela de opções de modelagem (Modelling Options).

Dentre os campos a serem preenchidos, tem-se a opção do tipo de *blank* (*blank type*), que determina a próxima janela a ser mostrada. Caso o usuário não queira alterar nada, os campos são preenchidos com valores padrão, baseados no nome do usuário e data atual. O tipo de *blank* padrão (peça bruta) é o "barra sólida". O sistema orienta o usuário, solicitando ao mesmo as informações relevantes para modelagem da peça e planejamento do processo.

Se for escolhida o *blank* (peça bruta) barra sólida aparece uma nova janela (figura 9.6c) requisitando as informações geométricas mais relevantes de uma barra sólida (*feature* barra sólida), que são o diâmetro e o comprimento. Assim são inseridas e armazenadas as informações importantes para uso futuro pelos demais módulos do sistema WebMachining, em especial nas atividades de planejamento de processo. Informações sobre tolerância dimensional e rugosidade geral do projeto são também solicitadas.

A última janela desta fase preparatória (figura 9.6d), dá ao usuário as opções de modelagem ponto de origem (esquerda ou direita) e se prefere iniciar a modelagem pela região externa ou interna da peça. O padrão é a modelagem esquerda (da esquerda para a direita) e *features* externas, que é a preferência mais usual entre os projetistas.

Prosseguindo na modelagem, tem-se a janela de desenho ou janela principal (figura 9.1), onde a peça desejada é modelada, utilizando-se das *features* de forma disponíveis na biblioteca de *features*. Inicialmente modela-se a peça usando as *features* pelo método de adição, ou seja, usa-se as *features* como blocos de construção da geometria da peça (tipo tijolos/lego), que será representada por um perfil 2D da peça, baseado na taxonomia CAM-I (1986), e estendendo-a.

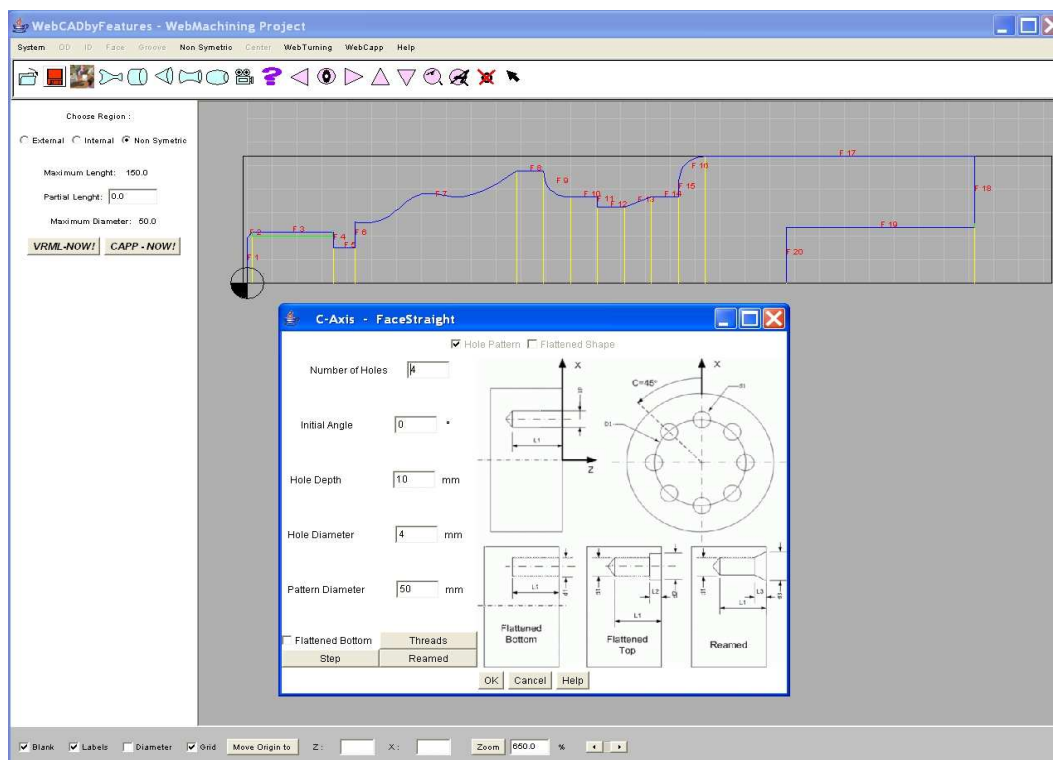


Figura 9.7: Modelagem da *feature* de Eixo C padrão de furos axial.

Após a conclusão desta fase de adição de *features* (*features* concêntricas Região Externa e Região Interna), inicia-se a modelagem da peça por *features* de subtração, ou seja, de volumes a serem removidos da peça modelada pelo método de remoção de volumes de material, estando estas *features* associadas ao Eixo C do centro de torneamento (*features* não-concêntrica), ferramenta rotativa, como *features* de rasgo de chaveta, *slots*, furos, entre outros (operações de fresamento e furação radial e axial).

A figura 9.7 apresenta uma tela do WebCADbyFeatures para modelagem de uma *feature* de Eixo C do tipo padrão de furos axial, que será executada na face direita da peça modelada.

O usuário tem a opção de dimensionar o desenho em 2D (*zoom*), movê-lo na tela e ainda gerar o sólido VRML a qualquer momento que desejar, permitindo assim a modelagem 3D correspondente, a partir do modelo geométrico 2D e de *features*, disponível no modelo de informação da peça.

A visualização em 3D é feita através de um *plug-in* para VRML, previamente instalado pelo usuário, no seu *browser*. Ao acionar o botão "VRML-NOW!", o modelo da peça é enviado ao servidor WebMachining via *servlets*, que a seguir grava o arquivo no servidor e o envia para o *browser* do cliente via mecanismo *post* do HTML, que faz a chamada do *plug-in* VRML disponível, permitindo a navegação no mundo virtual criado para a peça.

Tem-se a opção para salvamento local do modelo geométrico em 2D e 3D (extensão *wrl*) e de *features* (extensão *ftf*), desde que se altere a política de segurança da máquina Java local, permitindo a leitura e escrita de arquivos na máquina do usuário. A máquina Java vem configurada de forma segura, impedindo que *applets* tenham acesso aos recursos locais da máquina. Para gravar localmente é necessário degradar a política de segurança da máquina Java local.

No menu WebTurning, o usuário encontra opções para a visualização do chão-de-fábrica do

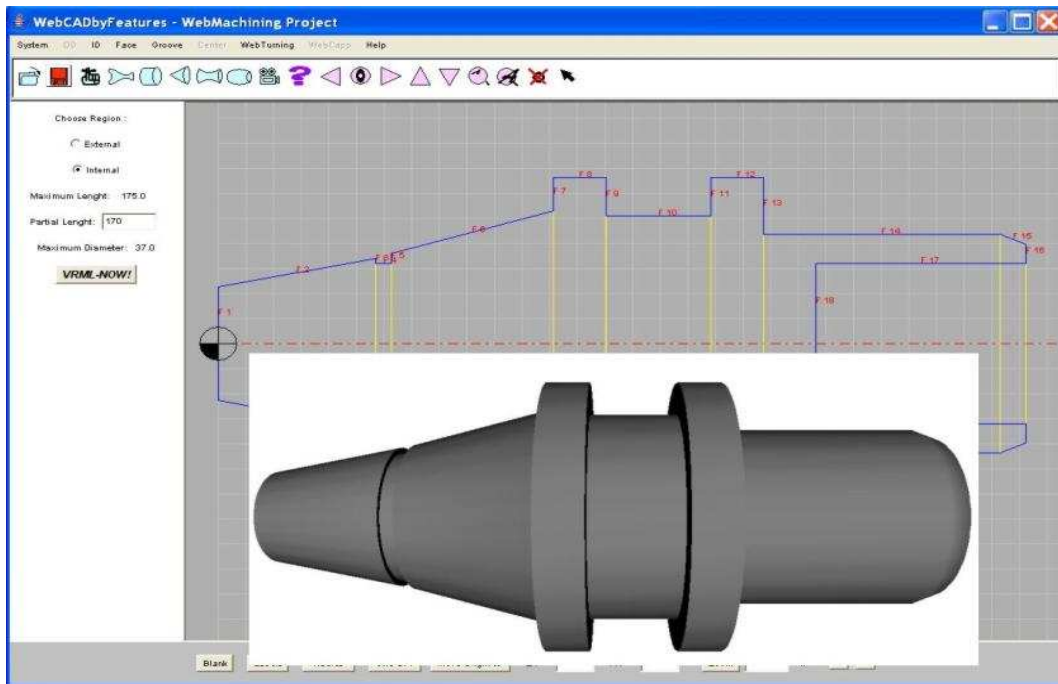


Figura 9.8: Interface WebCADbyFeatures: 2D e VRML.

GRACO em tempo real. Já o menu WebCAPP e o botão "CAPP_NOW!" dão acesso as funções de CAPP. O acesso ao banco de dados só é possível para usuários cadastrados, pelo menu "System" na opção de salvamento remoto. Na figura 9.8 tem-se outra peça exemplo complexa em 2D e o sólido, 3D, em VRML, correspondente.

9.5 Modelagem Colaborativa

Inicialmente é necessário acessar a interface do cliente denominada "WebCADbyFeatures Collaborative Design IPLayer Router Client", através do terceiro ícone da barra principal (figura 9.7), denominado "Connect Collaborative Design WebCADbyFeatures", mostrado na figura 9.9. Este *applet* é formado por diversos painéis, que disponibilizam as funcionalidades necessárias para permitir o gerenciamento e a comunicação para desenvolvimento da modelagem colaborativa.

Faz-se o registro no *Router* através do painel "Register" do agente de interface, fornecendo o "Agent Name = Alvares", "Password = alvares" e "Email = alvares@AlvaresTech.com" (figura 9.9). Os demais dados relativos ao nome do *Router* (Facilitador - AMR) e das portas TCP, já estão preenchidos. A seguir conecta-se o agente registrado anteriormente ao servidor/roteador JATLite (AMR), através do painel "Request" (figura 9.10). Os demais painéis "Compose", "FTP" e "Reserve" são usados para comunicação com os demais agentes que participarão da sessão de modelagem colaborativa.

No lado direito da GUI (figura 9.10) são disponibilizadas informações sobre a conexão com o Router. É utilizado o KQML como linguagem de comunicação entre os agentes. Algumas diretivas utilizadas são: *sender*, *content*, *receiver*, *performative* entre dezenas de outras associadas ao KQML.

A figura 9.11 apresenta um exemplo de modelagem colaborativa, onde o Agente de Interface "Alvares" compartilha seu modelo de *features* de produto com o Agente de Interface "jones". O Agente

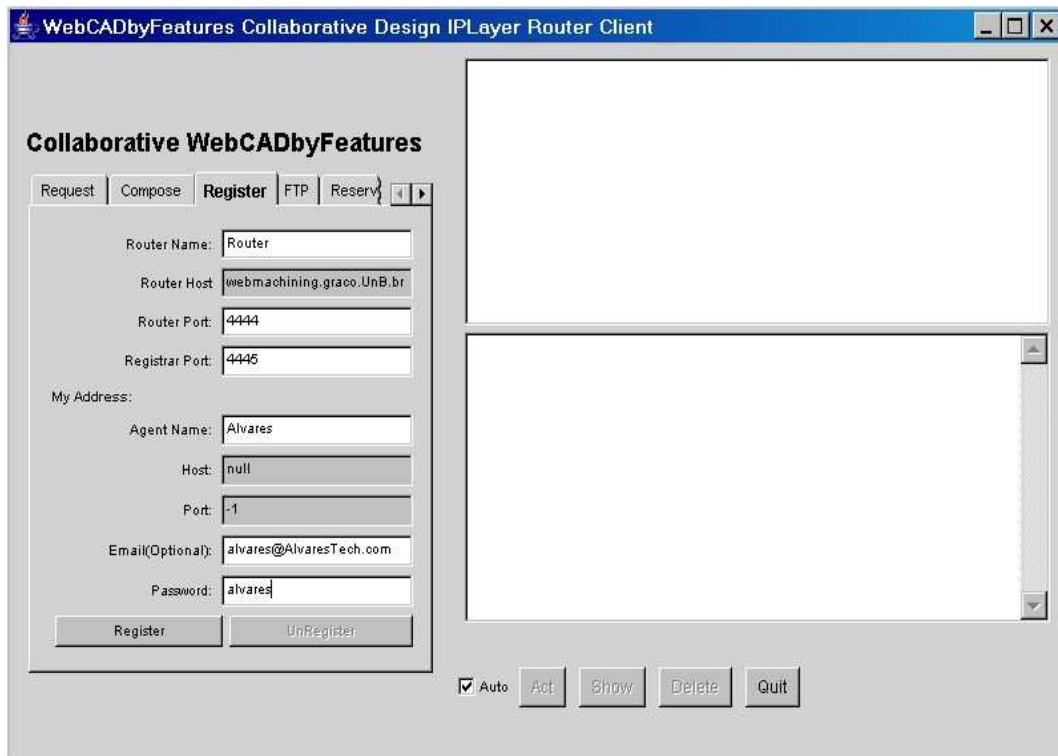


Figura 9.9: Interface do cliente para projeto colaborativo: "WebCADbyFeatures Collaborative Design IPLayre Router Client", apresentando o painel "Register".

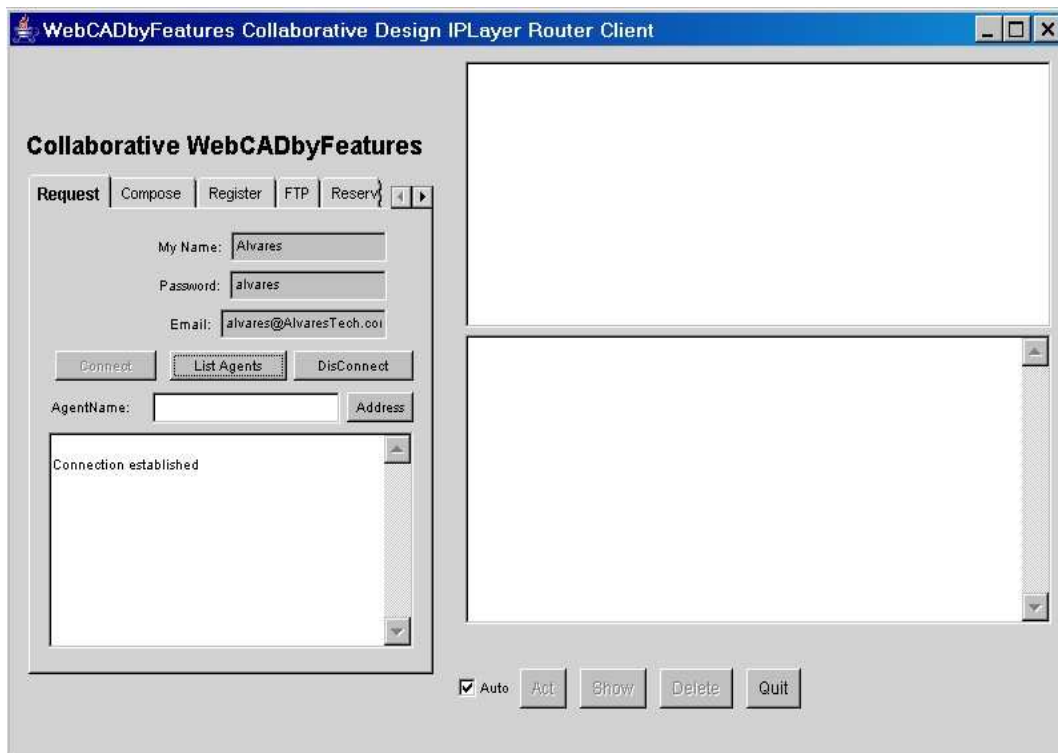


Figura 9.10: Interface do cliente para projeto colaborativo: Request do Agent de Interface "Alvares" no servidor Router do ambiente JATLite.

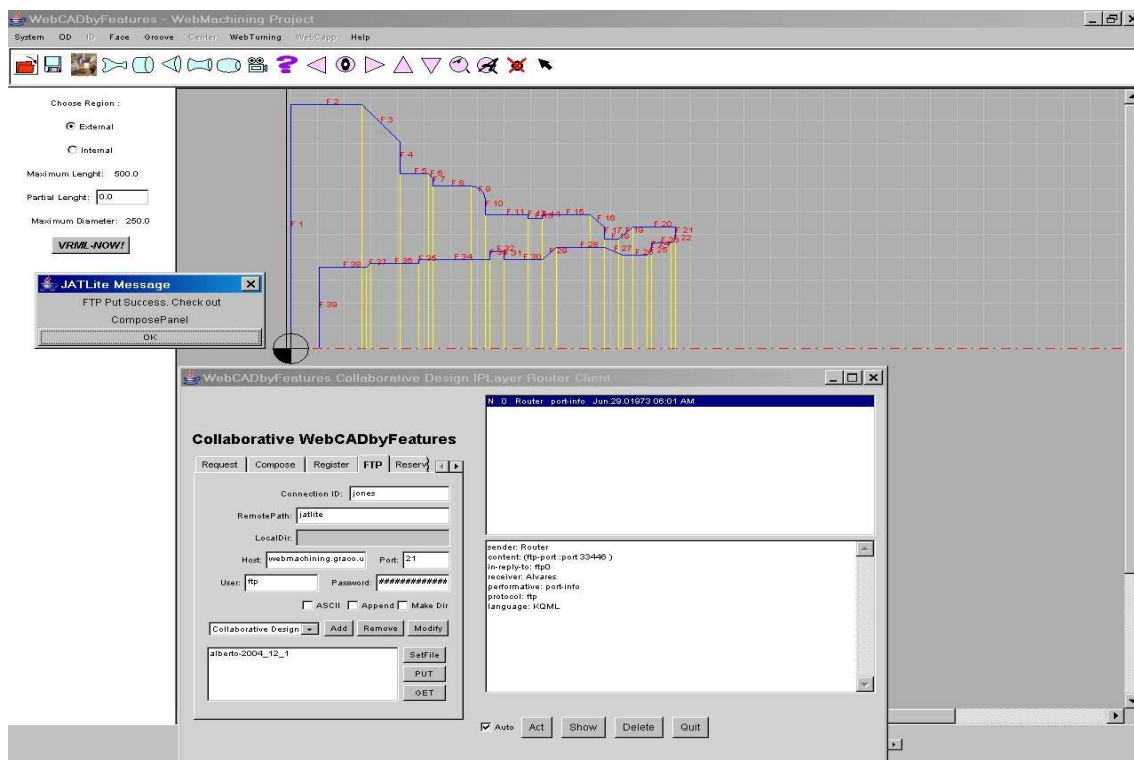


Figura 9.11: Modelagem colaborativa entre os agentes "Alvares" e "jones". Agente "Alvares" envia modelo por *features* para agente "jones".

"Alvares" envia por FTP o seu arquivo de projeto, denominado "alvares-2004_12_1", utilizando o Agente Router (gerenciador de sessão), para o Agente "jones", que irá recebê-lo diretamente em sua GUI, e a seguir fará a análise e as edições que achar conveniente.

A seguir o Agente "jones" envia o arquivo com as suas edições, nomeado pelo sistema de "jones_20-04_12_1" ou outro nome qualquer, para o Agente "Alvares", que irá recebê-lo em sua interface de projeto, com as edições realizadas pelo Agente "jones" (figura 9.12).

Os dois agentes podem conversar diretamente via *Router*, trocando mensagens através do painel "*Compose*", onde define-se uma ontologia, no caso associada a terminologia de desenvolvimento de produto, denominada de "*Project*", e envia-se o "*Content*" desta mensagem utilizando a linguagem KQML, com suas diretivas "*sender: jones*", "*receiver: Alvares*" e "*performative: tell*". Por exemplo o Agente "Alvares" recebe mensagem do Agente "jones", sobre as edições realizadas no projeto. Pode-se enviar *e-mail* via *Router*, caso haja necessidade de comunicar-se com outro agente.

9.6 Discussão

O sistema de projeto colaborativo WebCADbyFeatures utiliza mecanismo assíncrono de gerenciamento de sessões de modelagem colaborativa associado à uma arquitetura multiagentes para compartilhamento dos modelos de *features*, diferentemente da arquitetura WebSpiff (BIDARRA *et al.*, 2001) que usa abordagem síncrona, utilizando um servidor que gerencia as sessões de modelagem de produto colaborativa em execução. O servidor WebSpiff mantém o modelo de produto centralizado no servidor; permitindo assim a atualização sincronizada de todas as sessões de modelagem em curso.

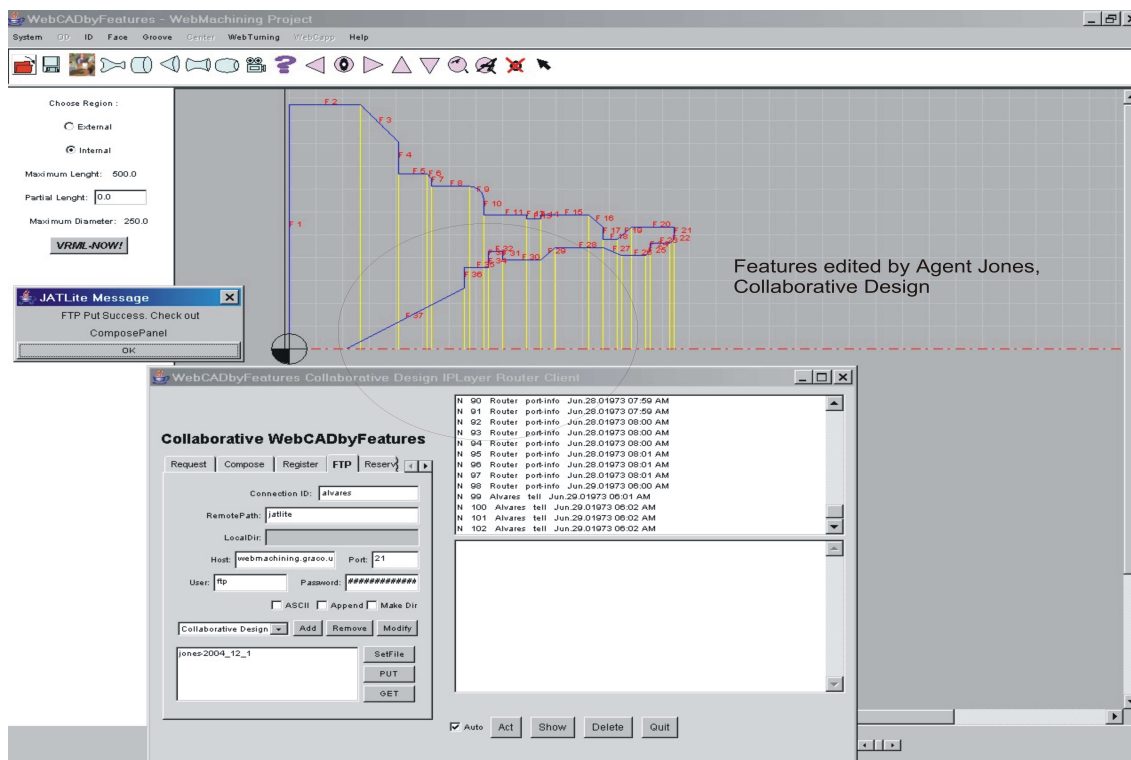


Figura 9.12: Modelagem colaborativa entre os agentes "Alvares" e "jones". Agente "Alvares" recebe o modelo por *features* do agente "jones", após efetuar as alterações no modelo de *features*.

Não é utilizado a figura de agentes no WebSpiff. WebCADByFeatures mantém os modelos de *features* descentralizados, e usa a figura do Agente Facilitador para compartilhar os modelos para os demais agentes de interface, via FTP, HTML ou *e-mail*.

No modelo de *features* da peça acabada, cada *feature* de forma modelada pode conter informações sobre *features* de tolerâncias (dimensão, rugosidade, forma, posição, orientação, batimento e localização) e *features* de referências de projeto. As peças bruta (*blank*) e acabada, que compõem um projeto de um agente de interface têm associadas também *features* de tratamento térmico, *features* de tratamento superficial, *features* de material da peça e *features* de dados produção (tamanho do lote, data de entrega, usuário, senha, entre outros). Este modelo de informação é bastante completo e está implementado via banco de dados MySQL.

A arquitetura WebCADbyFeatures pode ser definida como um sistema de projeto colaborativo orientada à *Web* baseada em *features*, usando mecanismo assíncrono de gerenciamento de sessões de modelagem associado a um sistema multiagentes e um banco de dados MySQL, sendo multi-usuário e distribuída, bem como concorrente e assíncrona.

Em um contexto de modelagem colaborativa, a complexidade do cliente é principalmente determinada pelas facilidades de modelagem e interatividade implementadas no cliente, enquanto que o carregamento da rede é determinado principalmente pelo tipo e tamanho do modelo de dados sendo transferido de/para os clientes.

WebCADbyFeatures adota uma solução de compromisso, concebida entre os extremos, *thin clients* e *fat clients*, disponibilizando funcionalidades no lado do cliente e no lado do servidor. No lado do servidor tem-se a GUI que disponibiliza todas as facilidades de modelagem em 2D de uma peça. No

lado do servidor, é realizado o gerenciamento das sessões de modelagem colaborativa via Agente Facilitador (JATLite) e a geração do modelo da peça 3D em VRML, através de *servlets*.

WebCADbyFeatures oferece total facilidade de interação e modelagem local, mantendo seu próprio modelo de *features* local. A comunicação com o Agente Facilitador é então requerida quando houver necessidade de sincronizar as modificações de dados do modelo local com os outros clientes. Neste ambiente colaborativo, onde clientes podem concorrentemente modificar o modelo de dados local, a prevenção de inconsistências de dados entre diferentes clientes torna-se um problema crucial, sendo equacionado através do sistema multiagentes, via Agente Facilitador e KQML. Todos os clientes devem ter o mesmo modelo de *features*. Quando um usuário altera o modelo, todos os clientes são atualizados via Agente Facilitador, se o Agente de Interface desejar. Para que o processo tenha consistência as mensagens de atualização em KQML são entregues na mesma ordem para todos os clientes, quando um Agente de Interface requer o compartilhamento de seu modelo.

O carregamento da rede, associado à largura de banda e ao ambiente computacional heterogêneo também devem ser levados em consideração no desenvolvimento de uma ferramenta de projeto colaborativa. No caso do WebCADByFeatures utiliza-se o protocolo TCP/IP, em especial as tecnologias *Web* para efetiva comunicação durante o ciclo de desenvolvimento de produto (CAD, CAPP, CAM), usando uma arquitetura cliente-servidor. Assim, o sistema depende da largura de banda disponível entre o cliente e o servidor, para uma utilização confortável e rápida das GUIs.

No lado do cliente, o *applet* que é recebido via *Web* a partir do servidor WebMachining tem o tamanho de 2,5 MBytes. Em uma conexão Internet com 128 KBytes/s, é necessário cerca de 30 segundos para fazer o *download* e iniciar a máquina virtual Java no cliente, não tendo portanto um tempo significativo de espera por parte do cliente, diferentemente do que ocorre com o sistema WebCAD 2000 do Cybercut, que tem um tempo de espera da ordem de dois minutos.

O compartilhamento do modelo de *features* com os demais agentes de interface do sistema WebMachining é feito via *download* (FTP) do modelo de *features*, e seu armazenamento em um diretório público no servidor de FTP WebMachining (<ftp://WeMachining.AlvaresTech.com/jatlite>). Na sequência, o agente que participa da modelagem é comunicado sobre a disponibilização do novo modelo de *features* e o Agente Facilitador faz a intermediação da comunicação entre os agentes do sistema multiagente, realizando o *upload* do modelo de *features* do primeiro agente e carregando automaticamente na GUI do cliente, que poderá visualizá-lo em 2D, bem como em 3D, em seu *applet* Java, editá-lo, salvá-lo localmente e no servidor, bem como compartilhar com outros agentes que participam da sessão de modelagem colaborativa e de forma assíncrona. Há a possibilidade de disponibilizar o modelo de *features* via e-mail, comunicando a URL em que se encontra o novo modelo, caso um dos Agentes de Interface não esteja *on-line* durante a sessão de modelagem colaborativa.

O procedimento de *download* e *upload* de arquivos com os modelos de *features* é muito rápido, pois os arquivos são muito pequenos. Por exemplo um arquivo típico (ftr) com 40 *features* de forma e todos seus atributos, tem cerca de 12 kBytes, o que é insignificante para o compartilhamento de arquivos via Internet de banda larga. Os tamanhos de arquivos associados aos modelos de *features*, armazenados pelo sistema WebCADbyFeatures, podem ser verificados na URL <ftp://WebMachining.AlvaresTech.com/jatlite>. O mesmo raciocínio se aplica aos arquivos VRML (wrl), que tem cerca de 14 kBytes para o arquivo de modelo de *features* correspondente no exemplo acima. Os arquivos VRML estão disponíveis na URL <ftp://WebMachining.AlvaresTech.com/vrml>.

Toda a interface com o sistema se inicia a partir de um navegador *Web* (*browser*²), sendo necessário a instalação de dois *plug-ins* na máquina cliente, VRML e máquina Java compatível com AWT (Java 1.1). Como *applets* são seguros por natureza, impedem a gravação de arquivos na máquina local. Caso o usuário deseje a gravação de arquivos na sua máquina é necessário alterar a política de segurança da máquina Java, modificando ou criando, por exemplo, o arquivo de configuração “.*java.policy*” e incluindo a diretiva: *grant {permission java.security.AllPermission;};*

Caso seja usado algum *firewall* pessoal na máquina cliente ou na rede local do usuário, é necessário liberar o acesso para algumas portas usadas pelo TCP (*sockets*), como a porta 8080 para a conexão com o servidor Tomcat, portas 4444 e 4445 para conexão com o servidor JatLite, porta 80 para conexão com o servidor Apache e porta 21 para conexão com o FTP.

É difícil comparar o sistema WebCADbyFeatures com outras propostas e sistemas computacionais acadêmicos, pois não se tem referências bibliográficas de nenhum outro sistema computacional semelhante ao WebCADbyFeatures, cabendo destacar que o mesmo é parte de um ambiente de desenvolvimento de produto colaborativo totalmente baseado na *Web* e disponível para uso. A quase totalidade dos sistemas relatados na literatura não estão *on-line*. Deve-se destacar duas exceções WebSpiff e Cybercut, que podem ser acessados e inicializados, entretanto param de funcionar após algum tempo de uso, mostrando a existência de vários problemas na arquitetura e implementação, sendo que muitos módulos e funcionalidade destes sistemas computacionais não funcionam ou estão inacessíveis.

Uma virtude do WebCADbyFeatures em comparação com os sistemas *on-line* WebSpiff e Cybercut é a sua facilidade de uso, não necessitando de maiores conhecimentos técnicos por parte do projetista. Um usuário pouco experimentado pode facilmente utilizar o sistema, pois o mesmo foi concebido para ser facilmente utilizável não necessitando de instalação ou configuração além dos *plug-ins*, orientado por menus, com uma modelagem lógica baseada em objetos, tendo uma interface intuitiva levando a um rápido desenvolvimento de produto. Uma peça relativamente complexa com 40 *features* por exemplo, pode ser modelada em menos de dez minutos, incluindo a instanciação de *features* de forma, de tolerância, de material, de tratamento superficial, de tratamento térmico, de referências de projetos e dados de produção. A partir da GUI do WebCADbyFeatures pode-se acessar todos os demais módulos do sistemas (VRML, WebCAPP (*servlets* e *applets*), WebTurning, WebDNC e WebCam (vídeo e áudio)).

Por fim, para tornar as sessões de modelagem colaborativa mais atrativa, recomenda-se fortemente a utilização do software SKYPE (<http://www.skype.com>) como ferramenta de comunicação baseada em VOIP (voz sobre IP), por ser uma ferramenta multi-plataforma (Windows, Linux, Mac e Pocket PC), bastante eficiente e gratuita, ao contrário do MSN Messenger da Microsoft, que não tem no momento a mesma eficiência além de ser proprietário, não sendo disponibilizado para outras plataformas computacionais, como Linux. Usando esta ferramenta de comunicação de voz sobre IP, pode-se aproximar ainda mais os projetistas, facilitando a sua comunicação e o desenvolvimento de produto.

²Como o Internet Explorer é o *browser* mais utilizado, deve-se instalar uma máquina Java diretamente da Sun, pois a Microsoft não fornece mais suporte para máquina Java nativa, nas novas versões do Windows. Quando se instala o Windows XP não se tem *plug-in* para Java.

Capítulo 10

Planejamento do Processo Não-Linear: WebCAPP

Este capítulo descreve o uso e detalhes da implementação computacional do sistema generativo de planejamento de processo auxiliado por computador denominado de WebCAPP, voltado para usinagem de peças rotacionais constituídas de *features* concêntricas e não-concêntricas, em um centro de torneamento Galaxy 15M da Romi. O sistema WebCAPP gera o macroplanejamento e microplanejamento do processo de fabricação da peça modelada pelo sistema WebCADbyFeatures, baseado nas definições e nomenclatura STEP-NC. As saídas geradas pelo WebCAPP estão associadas a um plano de processo não-linear representado por um grafo E/OU, plano de processo linearizado e programa NC em código G. O sistema é baseado em uma arquitetura cliente-servidor utilizando tecnologia *Web*, *servlets* e *applets*, sendo inicializado através da GUI WebCADbyFeatures, após a modelagem colaborativa da peça rotacional.

10.1 Introdução

As atividades do CAPP iniciam-se com a normalização das *features* modeladas pelo projetista no ambiente WebCADbyFeatures. Esta normalização consiste em identificar as *features* e representá-las de uma única forma (formato canônico), evitando redundâncias e variações em sua representação, facilitando assim o mapeamento de *features* para a visão de usinagem.

Todas as *features* da peça exemplo da figura 10.1 podem ser representadas como *ODTapered*, mas de acordo com a metodologia utilizada f1 e f4 são *FaceStraight*, f2 é *ODStraight* e somente f3 é do tipo *ODTapered*.

O procedimento de normalização se justifica devido a grande flexibilidade dada ao projetista no ambiente WebCADbyFeatures. O usuário pode realizar a modelagem por *features* de síntese de projeto de várias maneiras, dificultando o desenvolvimento de um CAPP rápido e robusto, pois esta flexibilidade na modelagem e representação das *features* de forma (fig. 10.1) leva a uma série de casos, *loops* e exceções, dificultando também o processo de detecção de falhas do sistema. Assim, optou-se por esta fase de normalização, a qual deixa os dados em um formato padronizado para o sistema WebCAPP.

Uma vez compatibilizado os dados da visão de projeto, inicia-se a primeira fase do CAPP, decom-

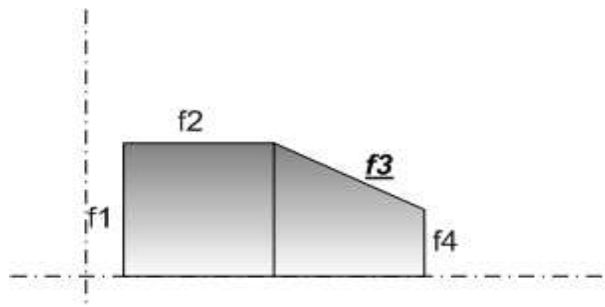


Figura 10.1: Normalização das *features* de projeto.

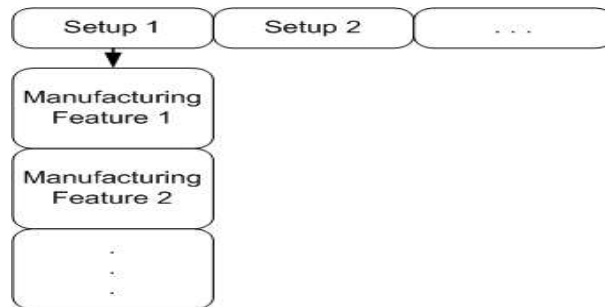


Figura 10.2: Decomposição orientada à geometria.

posição orientada ao *setup*, que é responsável por determinar as possíveis *features* de fixação e linhas delimitadoras, com alternativas.

Na fase decomposição orientada à geometria definem-se os blocos geométricos funcionais que serão usinados em cada *setup*. Essas estruturas funcionais são as *features* de usinagem e de torneamento armazenadas em um vetor, que por sua vez está dentro de um vetor de *setups*, conforme mostrado na figura 10.2.

De acordo com as diversas características, um conjunto de *features* pode ser interpretado como *feature* de usinagem de face, que possui como característica básica ter uma grande movimentação no eixo X em comparação com a movimentação no eixo Z; *feature* de usinagem (geral) que possui características monotônicas (diâmetros somente crescentes ou somente decrescentes) e *feature* de usinagem especial, na qual os diâmetros crescem e decrescem, por exemplo.

Por último tem-se a decomposição orientada à operação que consiste em montar a informação adquirida numa estrutura de *Workingsteps* e *Workplans*, tal como o previsto pelo Step-NC. Nesta fase são montados os *Workingsteps*, determinando uma operação de usinagem para cada *feature* de usinagem identificada, levando em conta as alternativas de operações, criando sub-geometrias se necessário. Nesta fase também se determinam os planos de segurança e aproximação, obtendo junto com os anteriores, todos os parâmetros necessários para se desenhar uma boa estratégia de movimentação de ferramenta para o próximo passo.

Geração do Código G

A geração do Código G inicia-se com a leitura de cada *Setup* (*Setup1* e *Setup2*), onde é traçada uma estratégia para cada *feature* de usinagem com as informações agregadas durante todas as fases

anteriores, após a linearização do *Workplan* com alternativas. Assim é gerado um arquivo texto que pode ser carregado na máquina-ferramenta via WebTurning ou enviado para geração do plano de processo via XML, que gera um arquivo XML para o usuário, apresentando o plano de processo linearizado com o código G ao final do arquivo.

10.2 Implementação Computacional do WebCAPP

O objetivo principal do planejamento do processo é a geração do microplanejamento do processo que deverá conter um plano de processo linearizado e detalhado, bem como a geração do programa NC para este plano de processo. O CAPP possui duas interfaces que podem ser acessadas pelo usuário:

- via *servlets*: gera o plano de processo linearizado na forma tabular e o programa NC associado, apresentado em formato XML;
- via *applets*: gera os resultados intermediários associados as dez atividades de planejamento de processo, ou seja, da atividade de “decomposição orientada à geometria, *setup* e operação” ao programa NC em código G (RS 274).

Assim é possível acompanhar todas as atividades previstas no planejamento de processo através do *applet* WebCAPP, ou simplesmente opta-se pela geração do plano de processo linearizado e do código G via *servlets*.

10.2.1 Via Servlet

A figura 10.3 apresenta as saídas geradas pelo WebCAPP via *servlets*, designada por WebCAPPServlet, mostrando o plano de processo linearizado indicando as operações de usinagem, ferramenta de corte e condições de usinagem para cada *setup*. Também é apresentado o código G gerado pelo WebCAPP.

A figura 10.4 apresenta a simulação do programa NC executada no centro de torneamento e obtida através do sistema de captura de imagens do ambiente WebTurning, durante o *try-out* da peça, diretamente do *display* do CNC.

10.2.2 Via Applet

A figura 10.5 apresenta a impressão formatada para o “Console Java” dos resultados da atividade de decomposição orientada ao *setup*, geometria e operação geradas pelo WebCAPP versão *applet*. Pode-se verificar detalhadamente os resultados dos métodos e algoritmos aplicados nesta atividade, tendo como saída a classificação da peça e o tipo de fixação necessária, a quantidade de *setups*, as possibilidades de linhas delimitadoras (LD), os agrupamentos de *features* de usinagem e torneamento para cada *setup*, e os *Workingsteps* associando as *features* de usinagem (volumes de usinagem) com as operações de usinagem possíveis, gerando assim as alternativas de usinagem.

A figura 10.6a apresenta a lista de *Workingsteps* gerado pelo WebCAPPApplet. A figura 10.6b apresenta o grafo E/OU, apresentando as alternativas de operações que poderão ser executadas sobre as *features* de usinagem e torneamento, mostradas na figura 10.6a .

A figura 10.7 apresenta parte do programa NC gerado pelo WebCAPPApplet para uma peça.



Figura 10.3: Plano de processo linearizado e programa NC gerado pelo WebCAPPServlet em XML.

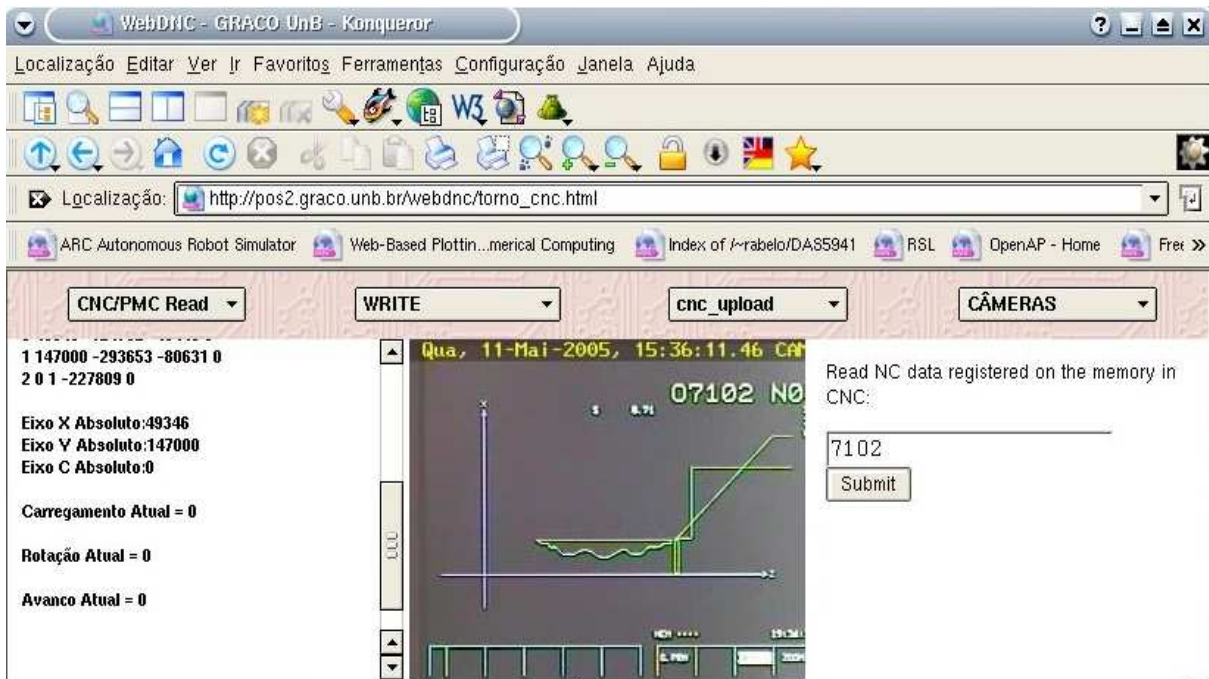


Figura 10.4: Simulação do programa NC capturado pelo sistema de processamento de imagens Web-Cam do sistema WebDNC.

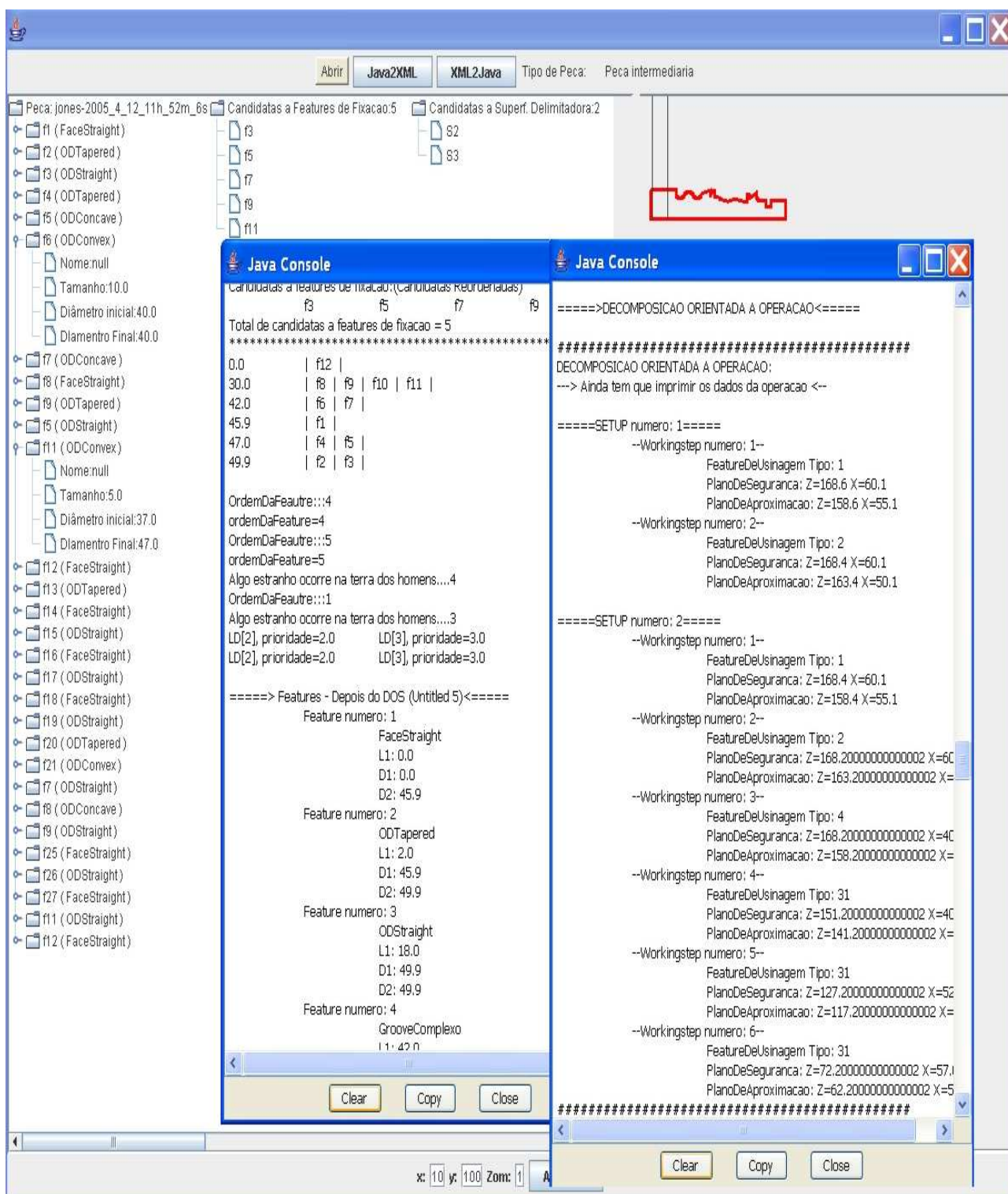


Figura 10.5: Resultado da atividade de decomposição orientada ao *setup*, geometria e operação gerado pelo sistema WebCAPPApplet, sendo a peça constituída por 29 *features*.

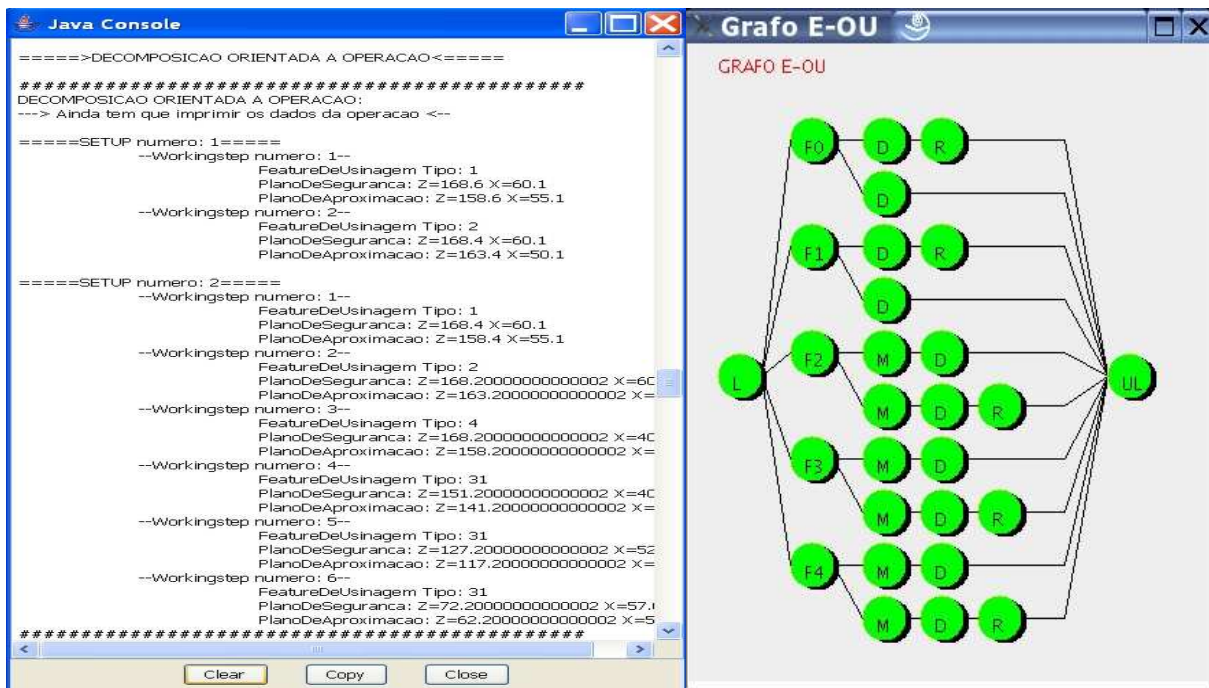


Figura 10.6: a) Lista de Workingsteps; b) grafo E/OU.



Figura 10.7: Programa NC gerado pelo sistema WebCAPPapplet.

10.3 Discussão

O CAPP foi implementado parcialmente, tendo várias atividades previstas e descritas nos capítulos anteriores ainda por implementar ou melhorar. As principais atividades idealizadas foram concluídas, como por exemplo o método de decomposição baseado em *setup*, geometria e operações de usinagem, que faz o mapeamento da maioria das *features* presentes em peças rotacionais simétricas e assimétricas. O método também reconhece *features* como *splines* e *features* de Eixo C, estendendo a taxonomia CAM-I.

Nesta versão implementada são reconhecidas todas as *features* concêntricas para região externa e interna. Foram implementadas algumas *features* de eixo C, como furos e padrões de furos não-concêntricos e rasgo de chave. As demais *features* não-concêntricas serão implementadas em versões futuras do WebCAPP.

As famílias de peças reconhecidas, ou seja, a forma padrão da geometria das peças tratadas pelo reconhecimento de *features* é bastante abrangente, contudo algumas geometrias ainda não são reconhecidas pelo método de decomposição. Os métodos reconhecem os relacionamentos geométricos e não-geométricos de *features*, em especial as *features* de tolerâncias e de referências, mostrando-se bastante eficiente na sua análise, e de certa forma mais abrangente por incluir o reconhecimento de *features* do tipo *spline* e *features* não-concêntricas.

A inclusão de *features splines*, *features* de tolerâncias, *features* de referências e o tratamento de regiões internas e de *features* de eixo C, dão uma maior amplitude ao CAPP, que pode tratar uma quantidade de *features* bem superior aos sistemas relatados na literatura (VARVAKIS, 1991) e de uma forma mais eficiente em termos de GUI com o usuário e uso de tecnologia *Web*.

O tempo de execução das atividades de CAPP, que não dependem do acesso ao banco de dados da Sandvik via Internet¹ ocorrem em menos de 5 segundos em um PC Pentium IV de 3,2 GHz em plataforma Linux, para a peça exemplo (capítulo 8), com 39 *features*.

O uso do *software* MatLab para executar o procedimento de otimização das condições de usinagem e a linearização do plano de processos para a peça exemplo (capítulo 8) leva cerca de dois minutos, pois tem-se vários tempos associados ao procedimento como: conexão via Internet para inicializar a aplicação de GA *standalone* do MatLab no servidor WebMachining; montagem dos dados de entrada para o algoritmo de otimização; escrever dados de entrada GA para arquivo texto (txt); chamar executável MatLab; carregamento do arquivo de entrada de dados pelo executável MatLab; executar o algoritmo genético; formatar saída; gravação saída formatada em arquivo texto que será lido pelas demais classes (módulos) do *software* WebCAPP. Uma forma de melhorar o desempenho do WebCAPP seria trabalhar com arquitetura paralela e distribuída de computadores (*cluster* de PCs), onde várias atividades poderiam ser executadas em paralelo.

O código G gerado e executado no *try-out*² mostra um índice de acerto muito grande no código G gerado, necessitando de poucos ajustes por parte do operador no chão-de-fábrica. Normalmente estes

¹A atividade de otimização das condições de usinagem através de GA usando o software MatLab depende do acesso ao banco de dados da Sandvik.

²A fase inicial de *try-out* consiste em uma operação do CNC em modo de teste, sem a movimentação dos acionamentos, para simulação do programa NC na própria máquina-ferramenta. A seguir executa-se o programa NC, bloco a bloco, em modo *dry-run*, onde os eixos X, Z e C são acionados e a placa e as ferramentas acionadas ficam estacionárias. O *try-out* finaliza-se com a usinagem da peça usando o Nylon (Technyl 6) como material de usinagem.

ajustes estão associados às condições de usinagem, e não à trajetória programada de ferramentas de corte, que exigem pequenas modificações, e normalmente focadas no *setup* da peça bruta (*Setup1*) e semi-acabada (*Setup2*) na máquina, definição de sobrematerial e determinação do Zero Peça, que são procedimentos manuais e dependem das instruções que o plano de fixações (*Setups*) define para o operador realizar junto à máquina. As vezes alguns ajustes são necessários por parte do operador neste plano de *setups*.

O material das peças usinadas no *try-out* foi *nylon*, pra evitar o desgaste das ferramentas. Algumas peças foram usinadas a seguir com latão e aço 1020, pois estes materiais estavam disponíveis no almoxarifado do laboratório. Foram necessários ajustes nas condições de usinagem recomendadas pelo sistema de GA, baseado em dados da Sandvik, da ordem de 15% nos valores de velocidades de corte e avanços.

Devido à ausência e disponibilidade de *softwares* CAPP generativos (comerciais e acadêmicos) para geração de código G, só foi possível realizar uma comparação com pacotes CAD/CAM tradicionais, no caso o *SmartCam Turning* da *Camax* para plataforma Solaris. A peça exemplo (capítulo 8) foi modelada por vários alunos da disciplina Tecnologias de Comando Numérico do curso de Engenharia Mecatrônica da UnB, usando o *SmartCam*, o *WebMachining* e através de programação manual. Os alunos não conheciam nenhum dos softwares, mas tinham conhecimento de CAD, tecnologia de usinagem, planejamento de processos e programação NC.

O ciclo de desenvolvimento da peça constituída pelas atividades de projeto detalhado, planejamento do processo e a geração do código G demorou uma média de 5 horas, usando o ambiente *SmartCam*, salientando tratar-se de usuários (alunos) sem experiência no uso do *SmartCam*. Por outro lado, usando o sistema *WebMachining*, os alunos levaram em média 20 minutos. Por fim, a programação manual da peça levou uma média de 8 horas, incluindo a simulação no CNC, a partir do macroplanejamento definido pelo *WebCAPP*. Mostra-se assim a grande facilidade de uso do sistema *WebMachining*, pois a interface CAD é muito intuitiva e o CAPP gera o código G de forma automática, sem interferência do usuário.

O sistema *WebMachining* tem potencial para ser integrado a um CNC industrial (baseado em PC e SO Windows), onde o *WebCADbyFeatures* seria a interface com o operador da máquina, permitindo a programação da peça orientada ao chão-de-fábrica, melhorando significativamente a GUI com o usuário. A aderência do *WebCAPP* ao STEP-NC também é uma fator de destaque, pois no futuro o sistema poderá ser migrado para CNCs baseados em STEP-NC. Pode também ser utilizado como ambiente de programação de máquinas para a indústria, principalmente que trabalhe por encomenda, projetando e fabricando peças diferentes a todo o momento.

Existem várias limitações na implementação, pois o sistema é um protótipo, sendo que o objetivo do seu desenvolvimento foi no sentido de comprovar a exequibilidade da metodologia, métodos e algoritmos, validando-os, bem como demonstrar a facilidade de uso, geração de boas soluções de planejamento de processo não-linear, sua linearização e a geração do código G, e por fim, a fabricação da peça. Obviamente não se pode comparar os recursos gráficos disponíveis no sistema *WeMachining* com sistemas comerciais, pois os sistema não foi implementado como *software* comercial.

Quando comparado a um CAD/CAM comercial, como o *SmartCam*, fica demonstrada a sua superioridade em termos de concepção e de facilidade de uso e tempo de geração do código G. O mesmo ocorre quando comparado com vários sistema acadêmicos relatados na revisão de literatura, que não

são tão abrangentes e normalmente não trabalham de forma distribuída e baseado em tecnologia *Web*, não podendo nem ao menos testá-los por não estarem disponíveis, tanto o executável quanto o código fonte.

Entre as limitações desta implementação para geração de planos de processos não-lineares, têm-se o universo de peças tratadas, onde várias *features* não-concêntricas não foram incluídas nesta versão, bem como uma melhor formatação na saída de dados intermediários do CAPP, como representação por grafo E/OU e saídas dos algoritmos genéticos via *applet*, que serão ainda contempladas nas versões futuras. Isto ocorreu devido ao tempo disponível para conclusão da tese, que completou quatro anos em setembro de 2005.

Capítulo 11

Teleoperação Centro de Torneamento: WebTurning

Este capítulo descreve os detalhes da implementação computacional do sistema de manufatura remota (telemanufatura) voltado para o torneamento de peças rotacionais em um centro de torneamento Galaxy 15M da Romi, denominado WebTurning.

11.1 Introdução

O desenvolvimento de metodologias e sistemas que permitam a fabricação de peças à distância é importante em um contexto de *e-Mfg*, pois, desta forma, o usuário/cliente e a empresa que fabrica as peças podem manter um contato a distância, mas de forma direta. Há também a possibilidade de utilização de sistemas desta natureza para Ensino a Distância, pois os equipamentos e acessórios utilizados para fabricação, que são de custo elevado, poderão ficar acessíveis a estudantes de instituições que não possuam tais equipamentos (laboratórios remotos). O sistema para teleoperação do centro de torneamento Galaxy 15M é uma das unidades de uma Célula Flexível de Manufatura para fabricação de peças rotacionais, constituída também por um robô industrial, um robô móvel, um micrômetro laser, coordenador da FMC e paletes de peças brutas e acabadas. As unidades da FMC podem ser visualizados na URL <http://video.graco.unb.br>.

O sistema WebTurning tem como objetivo a implementação de uma interface de teleoperação, utilizando os recursos do ambiente WWW, para supervisionar, monitorar e comandar remotamente um centro de torneamento da Romi, modelo Galaxy 15M com CNC Fanuc 18i-ta, tendo a Internet como via de controle e um *browser* como GUI.

Este sistema pode ser acessado via URL <http://WebDNC.graco.unb.br>, sendo baseado em uma arquitetura cliente-servidor (tecnologia *Web*) utilizando metodologia descrita em ÁLVARES e ROMARIZ (2002), sendo constituído por dois módulos principais:

- Servidores: representados pelos programas localizados em uma estação de trabalho (plataforma Linux) responsável pela supervisão e controle do centro de torneamento (WebDNC) e pela monitoração do chão-de-fábrica através da captura de vídeo e áudio (WebCam);
- Clientes: representados por *applets* Java e páginas HTML no lado do cliente.

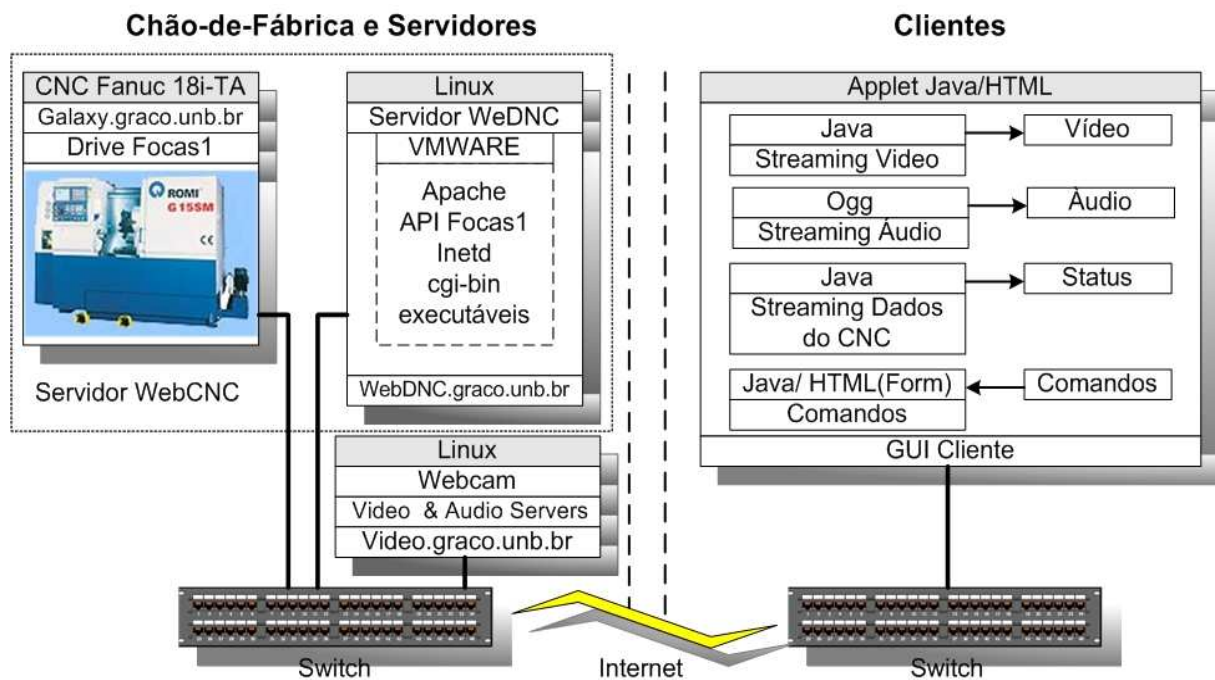


Figura 11.1: Arquitetura detalhada: módulos do sistema WebTurning.

11.2 Arquitetura WebTurning

O sistema WebTurning é constituído por quatro módulos básicos, três servidores (1, 2 e 3) e um cliente (4), sendo sua arquitetura cliente-servidor apresentada na figura 11.1:

1. servidores de *streaming* de vídeo e de *streaming* de áudio (WebCam);
2. servidor FOCAS1 (*Fanuc Open CNC API Specifications*) localizado no centro de torneamento, sendo representado pelo CNC Fanuc 18i-ta;
3. servidor de teleoperação WebDNC que atua como uma camada intermediária entre o CNC e o cliente usando mecanismos de acesso via *Web*, como CGI e *inetd*;
4. interface gráfica (GUI) em Java e HTML.

Os servidores WebDNC e FOCAS1 compõem a atividade WebCNC do modelo IDEF0 (fig. 6.12).

Este comando remoto é possível através das funções disponibilizadas pelo CNC 18i-ta Fanuc, via protocolo ethernet/FOCAS1 (*Fanuc Open CNC API Specifications*). A partir desta API e *driver FOCAS/ethernet* instalado no CNC, acessado via *socket* TCP/IP usando rede de comunicação ethernet, é possível executar cerca de 300 funções associadas ao controle do CNC, PMC/CLP (Controlador Lógico Programável) e DNC (Comando Numérico Distribuído). As figuras 11.2 e 11.3 apresentam as facilidades de comunicação presentes no CNC Fanuc 18i-ta, onde tem-se placa de rede ethernet, bibliotecas, API e protocolo de aplicação FOCAS1/DNC1 e protocolos TCP/IP.

O servidor WebDNC disponibiliza serviços de comando, execução de programas, *download* e *upload* de programas, tratamento de erros e demais funções associadas ao protocolo de comunicação disponibilizado pelo fabricante, trabalhando de modo bidirecional, recebendo comandos através da

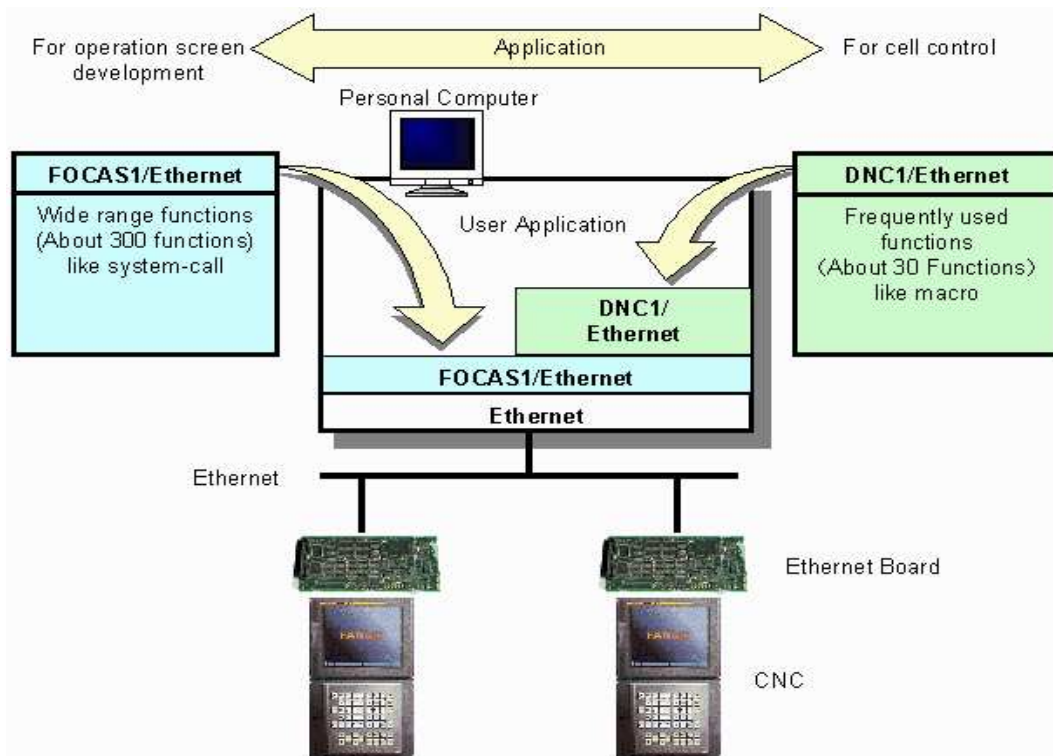


Figura 11.2: Hardware: placa ethernet, protocolo e API FOCAS1/DNC1 (Fanuc, 2003).

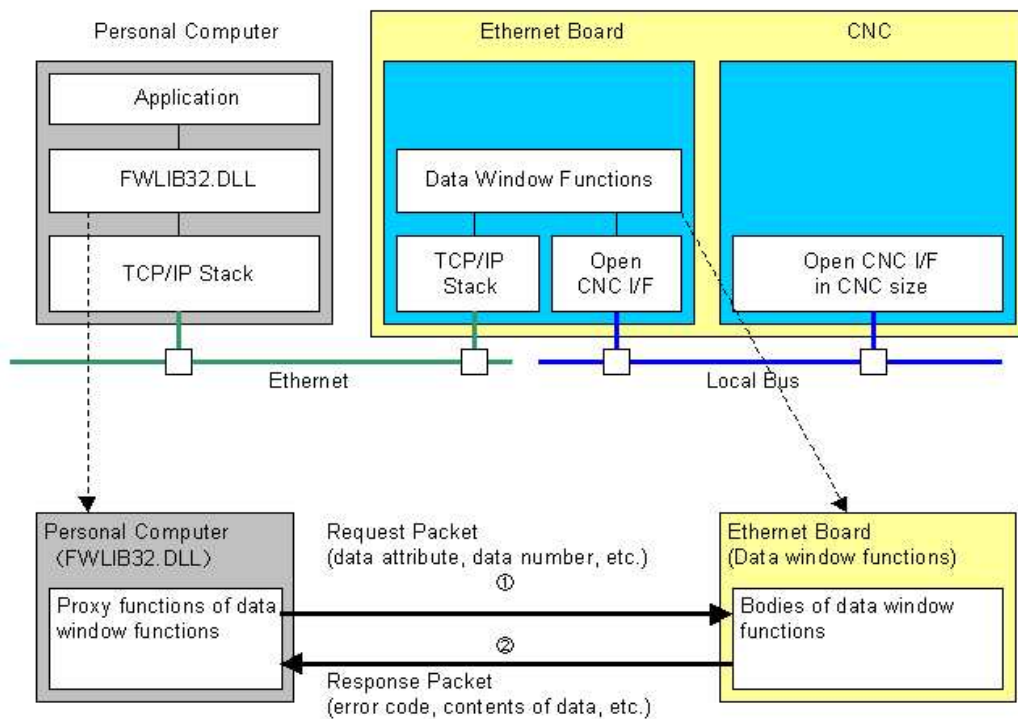


Figura 11.3: Biblioteca DLL FOCAS1/DNC1:fluxo de dados (Fanuc, 2003).

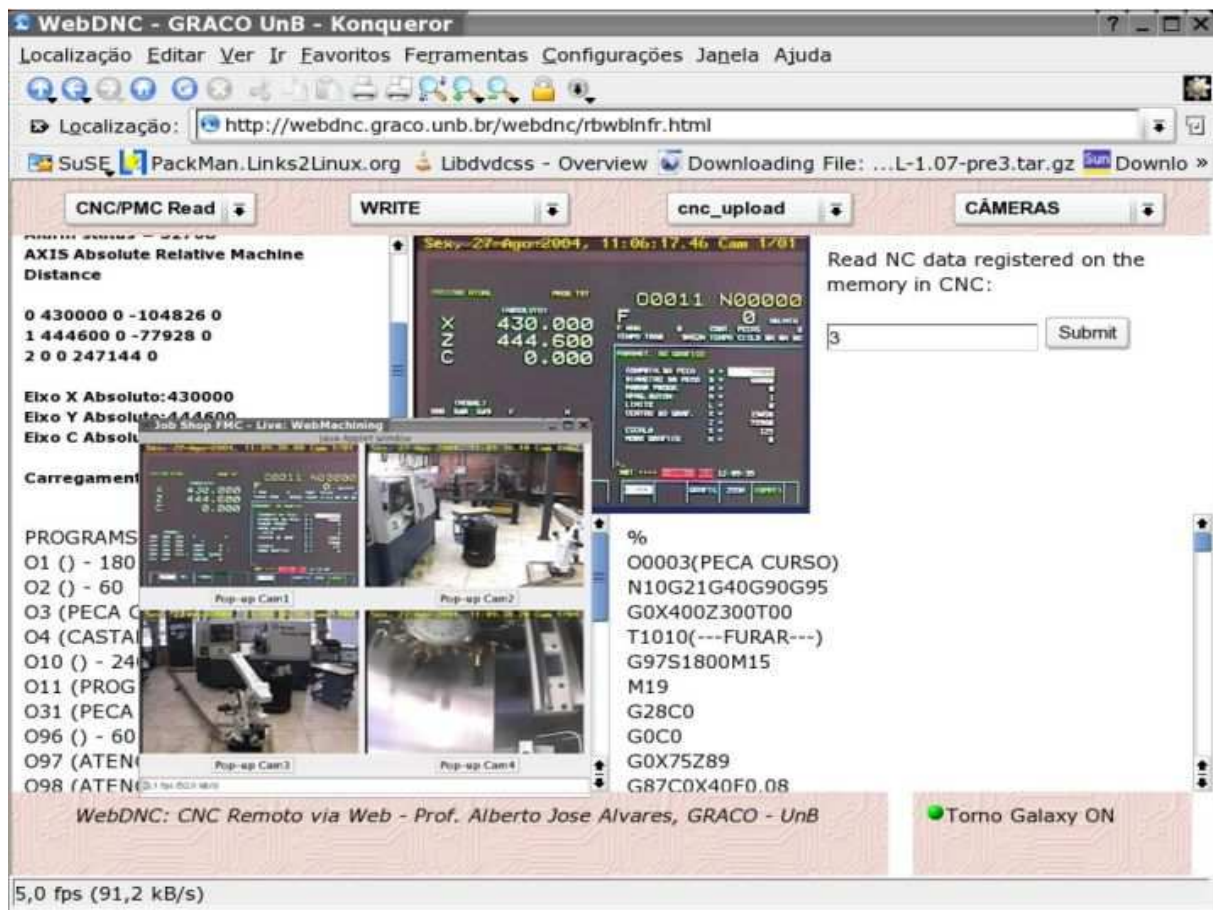


Figura 11.4: WebDNC: teleoperação e monitoramento remoto do centro de torneamento Galaxy 15M (<http://WebDNC.graco.unb.br>).

Internet e enviando dados de status do centro de torneamento via FOCAS1/DNC1. O WebDNC trabalha em uma camada intermediária, entre os clientes (PC) e o servidor FOCAS1 (centro de torneamento). O servidor de vídeo WebCam é responsável pela captura de vídeo e áudio e sua distribuição via *broadcasting* através do protocolo TCP/IP (Internet).

11.3 Interface com o Usuário WebDNC: GUI Cliente-Servidor

Para facilitar a utilização do WebTurning foi desenvolvido a interface gráfica WebDNC baseada em Java e HTML (figura 11.4), a partir da qual é possível controlar as funções relativas ao CNC, PMC (CLP) e DNC, sendo implementada usando tecnologia *Web*. A GUI, *front-end* do usuário, consiste de vários *applets* e formulários HTML, que são carregados pelo *browser* (<http://WebDNC.AlvaresTech.com>).

A figura 11.4 apresenta a tela principal de acesso ao sistema WebTurning, onde podem ser visualizados, nos *frames* superiores, os menus onde o usuário seleciona as funções FOCAS1. Na segunda linha de *frames* as três janelas são: janela de *status*, janela de realimentação gráfica e janela de função. Na terceira linha estão as janelas de envio e recebimento de mensagens. Através dos menus pode-se acessar os serviços remotos para teleoperação correspondentes às funções da biblioteca FOCAS1/Ethernet. Há um menu de opções que permite ativar a realimentação gráfica. A tela de *status*

permite que o estado do Galaxy 15M apareça na tela (posição, modo de operação e configurações principais). A tela de envio (caixa de envio) apresenta os dados a serem enviados, podendo ser corrigido ou editado antes do envio. A tela de recebimento mostra os dados recebidos pelo Galaxy 15M.

11.4 Funções FOCAS1/Ethernet Para Teleoperação do Centro de Torneamento

As funções especificadas pela biblioteca FOCAS1/ethernet (11.2 e 11.3) foram divididas em três grupos:

- Funções de leitura: permitem a leitura de uma série de dados e configurações do Galaxy 15M, como por exemplo as condições iniciais de operação e os dados de configuração e memória do CNC/PMC;
- Funções de escrita: permitem a alteração de configurações do CNC e do PMC, a movimentação dos eixos do centro de torneamento através de entrada de dados em modo MDI (*Manual Data Input*), por exemplo.
- Funções de programa: permitem realizar operações associadas ao *download* e *upload* de programas NC, *setup* e dados de ferramentas, bem como apagar programas que estejam na memória do CNC.

A figura 11.5 apresenta um exemplo de algumas funções implementadas. Utilizou-se para o desenvolvimento do sistema as linguagens de programação C, HTML, CGI, Java e JavaScript. O cliente acessa o sistema e tem a sua disposição uma tela de comando dividida em *frames* correspondentes as funções de teleoperação (figura 11.4), bem como as várias informações referentes ao centro de torneamento. Cada função possui uma página HTML correspondente, que descreve o funcionamento da função e pede para que o usuário defina os parâmetros da função.

A leitura destes parâmetros é feita por um CGI ou servidor específico via *inetd*. O cliente conecta-se ao servidor WebDNC, e este estabelece uma comunicação via *socket* com o CNC, segundo o protocolo FOCAS1/DNC1, enviando via rede ethernet e TCP/IP os *streaming* de dados e recebendo suas respectivas respostas ou mensagens de erro, que são apresentadas no formato HTML e *streaming* de dados, visualizadas na tela de comando e em *applets*.

11.5 Servidor WebDNC

A conexão Internet é implementada através de *sockets* entre os módulos WebDNC e módulo cliente. O servidor WebDNC é baseado em servidores HTTP/inetd/CGI, instalados em um computador pessoal plataforma Linux (PC Linux), permitindo o acesso por parte do usuário remoto via Web (html e java), que executa a função desejada da API do FOCAS1. Ao executar o programa no servidor local, este conecta-se ao servidor FOCAS1 no CNC, enviando a estrutura de dados necessária para executar a função no CNC. Após a execução no CNC, o *status* da função é enviada para o cliente remoto. Cada

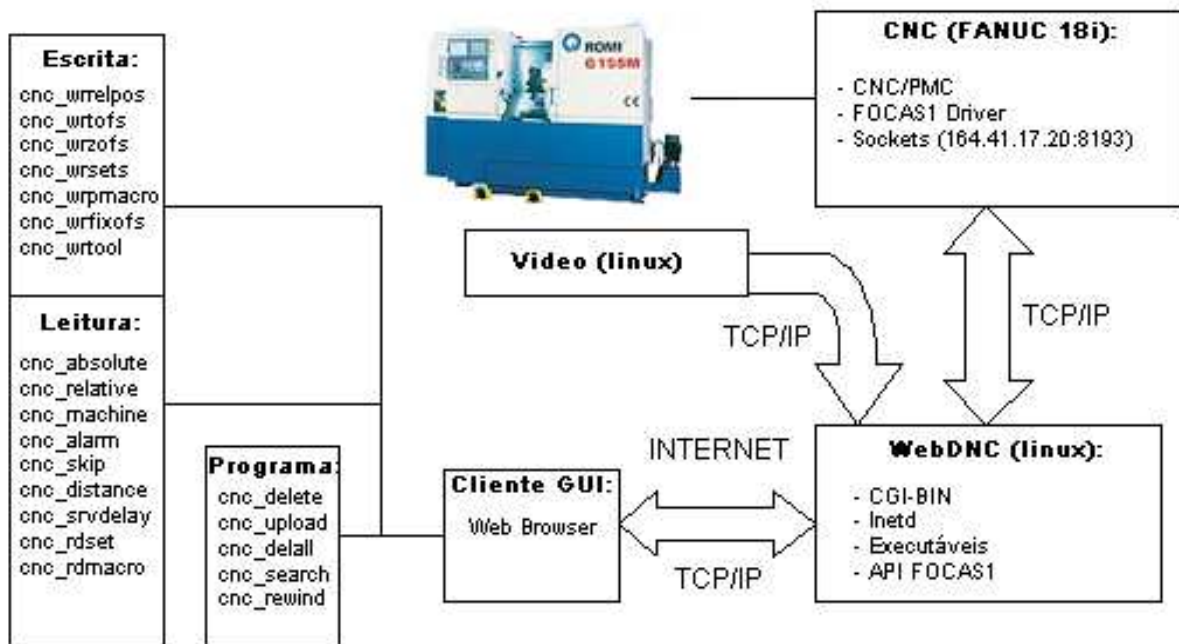


Figura 11.5: Arquitetura, fluxo de dados e algumas funções disponibilizadas pelo WebTurning.

função oferecida pelo protocolo FOCAS1 é disponibilizada para o cliente *Web* via mecanismos do tipo: CGI, chamada função via *inetd* ou chamada de função diretamente via *socket*.

Servidor FOCAS1 - CNC

O Servidor FOCAS1/ethernet está localizado no CNC da Fanuc 18i-ta do centro de torneamento Galaxy 15M. Disponibiliza uma conexão por rede local de comunicação ethernet via protocolo TCP/IP, usando *sockets* DNC1/FOCAS1 configurado para atender solicitações para o IP 164.41.17.20, na porta 8193. A partir deste *socket* tem-se acesso ao *driver* da Fanuc, que implementa as funções do protocolo FOCAS1/DNC. O cliente através de um *browser* conecta-se ao servidor WebDNC e executa a função desejada, que via *socket* envia uma estrutura de dados para o CNC, implementando as funções disponibilizadas pelo protocolo FOCAS1. Este servidor é proprietário da FANUC, sendo apenas habilitado e configurado no CNC. A API de desenvolvimento FOCAS1 está disponível apenas em plataforma Windows, fornecida pela FANUC por cerca de setecentos dólares, sendo ainda necessário habilitar o *driver* FOCAS1 no CNC FANUC, bem como realizar a configuração dos *sockets* no CNC.

O servidor WebDNC (no PC), que acessa o servidor WebCNC (no CNC) foi desenvolvido em Microsoft Visual C++ em plataforma Windows e instalado em uma máquina virtual Windows, em um PC com sistema operacional Linux, usando o servidor VMware (<http://www.vmware.com>). Assim os servidores WebTurning são executados em plataforma Linux, apesar da API FOCAS1 ser disponibilizada apenas para plataforma Windows.

Desta forma cada uma das funções tem o seu programa correspondente. Cada programa CGI via API FOCAS1, possui módulos que realizam as seguintes tarefas: leitura dos dados enviados por formulário ou *applet* Java; montagem do *streaming* de dados a ser enviado; inicialização do *socket* (IP e porta TCP do CNC: 164.41.17.20, 8193); estabelecimento da comunicação via estrutura de

dados FOCAS1; envio e recebimento de *streaming* de dados, fechamento do *socket*; verificação do *streaming* de dados recebido; montagem da página HTML e *applet* que será apresentada ao usuário com a resposta. A figura 11.5 apresenta alguns comandos implementados e aceitos pelo servidor WebTurning.

11.6 Servidores de Vídeo e Áudio: WebCam

O servidor WebCam (figura 11.6) é responsável pela captura de vídeo e áudio, e o envio de imagens e som via Internet. As imagens são capturadas por uma placa de captura de imagens baseada no *chip* Conexant B878, convertidas em *jpg* e enviadas para o cliente na forma de *streaming*, via *socket* TCP/IP. O servidor de *streaming* de áudio é responsável pela captura de áudio e a distribuição do *streaming* de áudio via Internet. O áudio do processo de usinagem é capturado por um microfone instalado na máquina e conectado à placa de som de um computador pessoal. Os servidores de vídeo e áudio estão associados ao módulo WebCam do sistema WebTurning, sendo implementados em plataforma Linux.

O servidor de vídeo disponibiliza ao cliente *Web* uma conexão via *sockets* ao sistema WebCam, para captura de imagens (servidor e cliente), permitindo a captura de até dezesseis entradas de vídeo, conversão das imagens capturadas para formato *jpeg*, detecção de movimento e gravação das imagens em *jpeg*; e por fim o envio ao cliente do *streaming* de imagens *jpeg* (30 frames/segundo para cada processador de imagem por placa) para animação no lado do cliente via *applet* Java, ou mesmo usando o mecanismo de animação *server-push* presente no Netscape e alguns *browsers*, com exceção do Iexplorer. O servidor de vídeo está disponível em <http://video.graco.unb.br>.

O servidor de vídeo também permite sua configuração via *socket*, podendo ativar detecção de movimento, ajustar qualidade da imagem *jpeg*, definir máscara a ser aplicada nas imagens visando definição de áreas da imagem para detecção de movimento, ajuste de brilho, contraste, entre outras opções de ajuste. Pode-se incluir mais câmeras para monitoração do chão-de-fábrica, devido a modularidade do sistema WebCam. Nesta implementação foram disponibilizadas quatro câmeras, para possibilitar uma imersão do cliente no chão-de-fábrica.

O servidor de *streaming* de áudio utiliza dois programas: Ices e Icecast (<http://www.icecast.org>), permitindo a captura de áudio de uma entrada da placa de som do PC e a conversão para o formato Ogg Vorbis (*ogg*), realizado pelo servidor Ices. A distribuição do *streaming* de áudio via TCP/IP é realizado pelo servidor Icecast. O servidor pode ser acessado pela URL <http://WebMachining.AlvaresTech.com:8000/audioturning.ogg>, sendo necessário um *plugin* para tocar os arquivos em formato Ogg Vorbis (<http://www.winamp.com>).

11.7 Discussão

O sistema desenvolvido mostrou-se bastante versátil e robusto, permitindo a fabricação de peças remotamente via Internet, através do uso de conceitos de *e-manufacturing* e telemanufatura, sendo o ambiente de teleoperação parte de um sistema integrado CAD/CAPP/CAM.

As facilidades de comunicação disponíveis no CNC como a placa ethernet, o protocolo TCP/IP

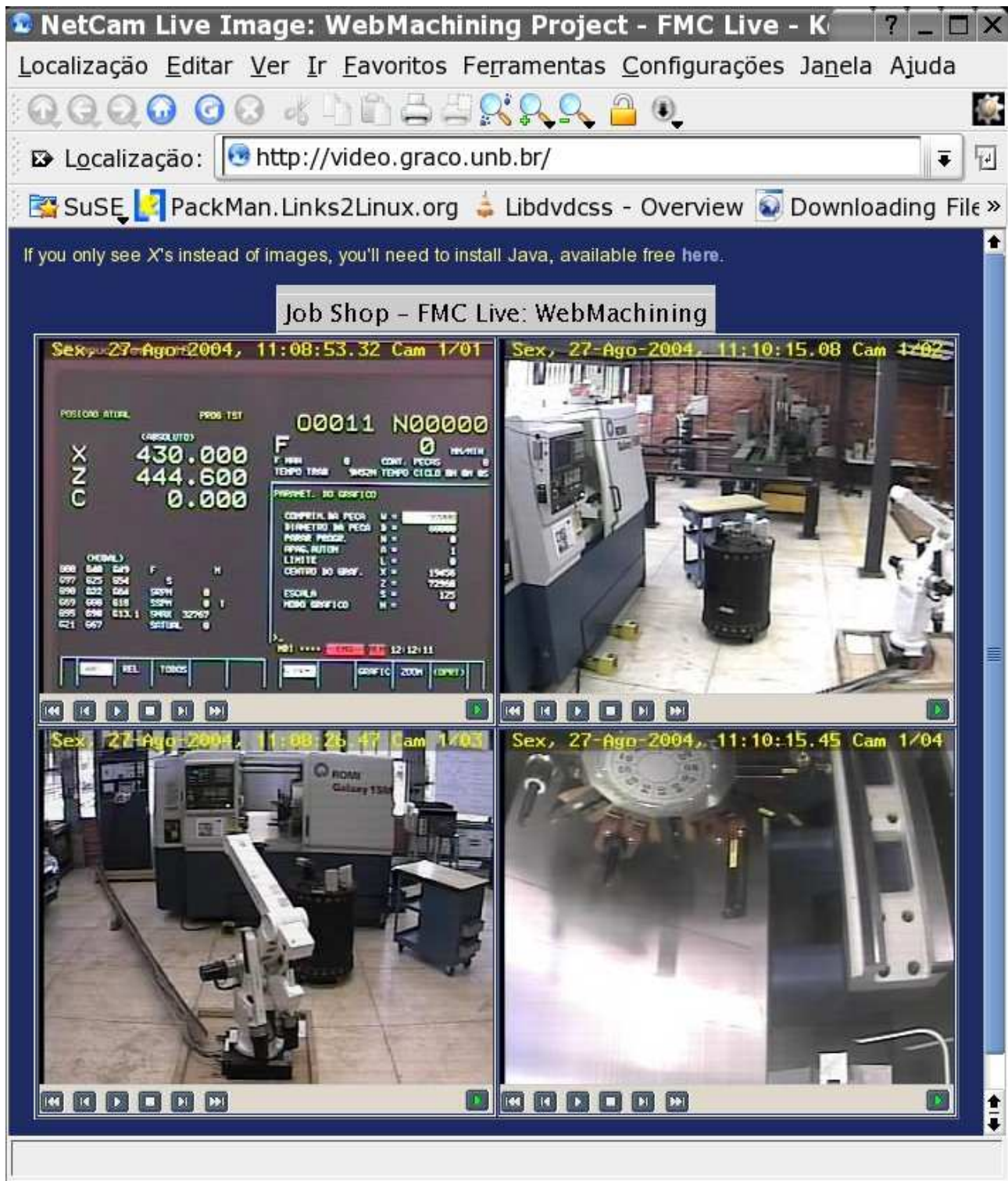


Figura 11.6: Sistema de monitoração por imagem WebCam (NetCam), constituído por quatro applets, que monitoram a tela do CNC, o interior da máquina e o exterior.

e os protocolos de aplicação FOCAS1 e DNC1, foram fundamentais para a implementação do sistema de teleoperação via *Web*. A API Focas1/DNC1 tem algumas limitações para acessar diretamente as funções associadas ao CLP, como por exemplo os botões associados à *Start Cycle*, Emergência, modos de operação (MDI, Automático e Edição), *Test Program*, entre outros. Para acessar estas funções, que são fundamentais para permitir total teleoperação do centro de torneamento Galaxy 15M, foi necessário a alteração do *Ladder* do CLP, realizado pela engenharia de aplicação das Indústrias Romi, fabricante do Galaxy 15M. Estas alterações se fizeram necessárias também para implementação da FMC, para permitir a integração lógica entre o centro de torneamento Galaxy 15M, o Robô ASEA IRB6, o micrômetro laser Mitutoyo, garra pneumática Schunk e o robô móvel Nomad XR4000 (AGV).

A Fanuc possui vários *softwares* para teleoperação do CNC 18iTa, como o *CNC Screen* e o BOP (*Basic Operation Package*), sendo todos desenvolvidos como parte da arquitetura *Cimplicity*, proprietários e de custo elevado. Nenhum dos programas de teleoperação e para DNC da FANUC permitem a teleoperação via *Web* (baseada em *browser*) e também não são multiplataforma, só trabalhando em arquitetura Windows.

A largura de banda e os atrasos inerentes ao protocolo TCP/IP (ÁLVARES *et al.*, 2000 e 1999) impõem uma forte restrição aos sistemas de teleoperação via Internet. Para resolver este problema é necessário dotar o sistema de teleoperação, no servidor junto ao CNC, de mecanismos que possibilitem a tomada de decisão em situações críticas, sem depender do lado do cliente, no caso do usuário/operador. A captura de imagens e *streaming* de vídeo *on-line* é fundamental para permitir a imersão do operador no sistema, possibilitando maior segurança no envio de comandos. A captura e *streaming* de áudio *on-line* para o cliente não é essencial, como o vídeo, mas possibilita ao operador remoto maior imersão e segurança durante o procedimento de teleoperação.

Como temos um atraso inerente ao TCP/IP, deve-se ter muito cuidado nas ações de comando executadas remotamente. Para se ter uma idéia de tempo real, o menor tempo de ciclo de varredura de funções de PLC no centro de torneamento Galaxy é de 13 ms. O TCP/IP não permite uma aplicação em “tempo real” para tempos de ciclo desta ordem de grandeza, sendo necessário tomar medidas de segurança quando da implementação de um sistema de teleoperação via TCP/IP, *Web*.

É necessário, portanto dotar o sistema de alguma inteligência para resolver conflitos que poderão ocorrer durante o processo de teleoperação. Para tornar o sistema de teleoperação mais seguro, deve-se utilizar uma arquitetura de controle do tipo supervisorio, onde as ações de controle são executadas localmente, a partir da solicitação de um comando remoto feita pelo cliente. Assim o sistema WebTurning é classificado como “*controle supervisorio com predomínio do controle realizado pelo operador humano*”.

As imagens em formato *jpg* com qualidade ajustada em 75% e resolução de 320x240 *pixels* geram arquivos com cerca de 6 kBytes que são enviados para o cliente. Considerando que a taxa máxima de imagens recebidas por cada *applet* é de 7 quadros/s e como têm-se quatro *applets* realizando as animações, a GUI do cliente pode receber no máximo 28 quadros/s, necessitando de uma largura de banda para total aproveitamento dos quadros de no mínimo 168 kBytes/s. Uma animação com 3 quadros/s é bastante aceitável para monitoração visual do processo de usinagem, necessitando assim de uma largura de banda de 72 KBytes/s. O sistema de vídeo permite o ajuste da quantidade de quadros recebidos, bem como se adequa à largura de banda disponível para a conexão. Já o arquivo

de áudio é enviado através do formato Ogg, sendo amostrado no servidor a uma taxa de 22.050 kHz, necessitando de uma largura de banda de 26 kbit/s. Assim para o sistema WebCam ter uma boa qualidade de vídeo e áudio necessita-se de uma largura de banda de no mínimo 76 KBytes/s, o que é plenamente satisfeito com os atuais serviços comerciais de Internet banda larga.

Para executar operações de telecomando no CNC é necessário uma pequena largura de banda, demandando menos de 2 kBytes/s, pois os formulários HTML e comandos em Java enviados para os servidores *Web*, *servlet* e *Inetd*, são da ordem 1 kByte/s. As ações de comandos são executadas localmente a partir de chamadas remotas realizadas por programas escritos em linguagem C, com cerca de 10 KBytes cada, que acessam o servidor FOCAS1 localizado no CNC. O servidor WebDNC está localizado na mesma rede ethernet que o CNC está conectando, tendo uma largura de banda disponível de 10 Mbits/s. Assim os comandos de teleoperação são executados rapidamente, e sem maiores problemas, pois não se utiliza o protocolo TCP/IP para ações de controle em “tempo real”, que demandem tempos de respostas abaixo de 7s, sendo este o atraso das solicitações de teleoperação em rede Internet do sistema WebDNC.

Em aplicações em rede Intranet, sob condições especiais, é possível executar controle em “tempo real”, da ordem de décimos de segundos. Já em rede Internet, onde o tempo de acesso é crítico, controle em “tempo real” é inviável, sendo mais racional trabalhar com arquiteturas de supervisão e de telecomandos, deixando as ações de controle no servidor junto ao processo.

Como ocorre com o a GUI WebCADbyFeatures, a interface com o sistema WebTurning se inicia a partir de um navegador *Web*, sendo necessário a instalação de dois *plug-ins* na máquina cliente, sendo o primeiro uma máquina virtual Java versão 1.1 e o segundo um tocador (*player*) de arquivos de áudio no formato Ogg Vorbis.

Caso seja usado algum *firewall* pessoal na máquina cliente ou na rede local do usuário, é necessário liberar o acesso para algumas portas usadas pelo TCP (*sockets*), como a porta 8000, necessária para a conexão de *streaming* de áudio e a porta 80 (normalmente liberada, HTTP) para *streaming* de vídeo. A política de segurança da máquina Java do usuário também deve ser modificada, para permitir a gravação de arquivos de vídeo em formato AVI no lado do cliente, a partir do *applet*.

Capítulo 12

Estudos de Casos: Peças Exemplos

São apresentados dez estudos de casos contendo todo o ciclo de vida de desenvolvimento de produto contemplado pela metodologia e sistema computacional WebMachining: projeto colaborativo, planejamento de processo e usinagem da peça. Os modelos de *features*, filmagens da usinagem, fotos e figuras estão disponíveis na URL <ftp://WebMachining.AlvaresTech.com/gallery/avi-estudosdecasos>.

12.1 Introdução

A tabela 12.1 apresenta uma compilação das *features* de forma presentes nas dez peças modeladas, sendo:

1. Seis peças estilizadas representando um jogo de xadrez (peão, torre, cavalo, bispo, rainha e rei) modeladas com *features* OD, ID, *splines*, faces, canal, rasgos radial (Eixo C) e furos axial (Eixo C). O *blank* é um tarugo de *nylon* de 50 mm. As peças são usinadas em um único *setup*;
2. Peça sete (tolerância) apresenta *features* de tolerância de concentricidade e de referência (relacionamentos não-geométricos) e *features* OD, ID, faces, canal, rosca métrica, de transição e cones. O *blank* é um tubo de *nylon* com diâmetro interno de 37 mm e externo de 75 mm. A peça é usinada em dois *setups*;
3. Peça oito (romi) é a peça adotada pelas Indústrias Romi em seus cursos de treinamento de operação e programação do Galaxy 15M, sendo constituída por vários tipos de *features* internas, externas e Eixo C, como: OD, ID, faces, cone, arco, canal, rosca métrica, rasgos radial e furos Axial (Eixo C). Os *blanks* são tubos de *nylon* e latão com diâmetro interno de 37 mm e externo de 75 mm. A peça é usinada em dois *setups*;
4. Peça nove, denominada por peça complexa é constituída por *features* OD, ID, arco, *spline*, canal, rosca, face, rasgo (Eixo C) e padrão de furos (Eixo C). O *blank* é um tarugo de *nylon* de 50 mm. A peça é usinada em dois *setups*;
5. Peça dez representa uma superfície “hiperbólica” (espelho hiperbólico) modelada por uma *spline*, OD, canal e face. O *blank* é um tarugo de *nylon* de 50 mm. A peça é usinada em um único *setup*.

Tabela 12.1: *Features* presentes nas peças estudadas.

	Face	OD	ID	Rosca	<i>Spline</i>	Arco	Cone	Rasgo Radial	Furo Axial	N°. <i>Features</i>
Peão	5	5	-	-	1	3	1	-	-	15
Torre	6	5	1	-	1	1	1	4	-	19
Cavalo	6	6	-	-	1	2	1	-	-	16
Bispo	6	7	-	-	1	2	1	-	-	17
Rainha	5	6	-	-	1	5	1	0	5	23
Rei	6	6	-	-	1	-	1	2	0	16
Tolerância	9	10	5	1	-	3	8	-	-	35
Romi	4	7	2	1	-	8	4	3	3	32
Complexa	6	7	1	1	1	2	2	3	3	26
Espelho	4	4	-	-	1	-	-	-	-	9

Tabela 12.2: Ferramentas Sandvik disponíveis no Galaxy 15M.

N. Torre	Porta-Ferramenta	Inserto	Operação
T0101	L166.5FA-2020-16	VBMT110312-PF4015	rosquear externo
T0202	broca - acionada	aço rápido	furar (6 mm)
T0303	LF123g20-2020B	N123G200300003-GM4025	sangrar (circular - 4 mm)
T0404	R416.2-0200C 3-31	LCMX030308-53 1020	furar (20 mm)
T0505	SVVBN-2020K11	VBMT1604 08-MM2025	tornear externo - neutra
T0606	R166.4kF-20F16	VBMT110312-PF4015	rosquear interno
T0707	SVJBL-2020K-16	VBMT1604 08-MM2025	tornear externo
T0808	DWLNL-2020-k06	WNMG060408-PM4015	tornear externo
T0909	A16R-SDUPL 07-R	DPMT070204-PM4015	tornear interno
T1010	fresa - acionada	aço rápido	fresar (12 mm)
T1111	N176.39-2020-10	RCMT0602M0 - 4025	sangrar (circular - 12 mm)
T1212	DDJNL-2020-K15	DNMG150608QM235	tornear externo

A tabela 12.2 apresenta as ferramentas Sandvik disponíveis na torre de ferramentas do centro de torneamento Galaxy 15M, que foram utilizadas na usinagem das peças.

12.2 Peças Jogo de Xadrez

12.2.1 WebCADbyFeatures

O projeto da torre é apresentado na figura 12.1, destacando a instanciação de uma *feature* de Eixo C, no caso rasgo radial, sendo as informações da *feature* de Eixo C incluídas a partir de um menu associado à uma *feature* OD. Este rasgo será gerado pelo processo de fresamento com uma ferramenta de 12 mm. O modelo VRML gerado pelo sistema WebCADbyFeatures é apresentado na figura 12.1.

O projeto do bispo é apresentado na figura 12.2, destacando a instanciação de uma *feature* concêntrica do tipo OD convexo (arco), onde os parâmetros associados à esta *feature* são incluídos a partir de um menu. Também é apresentado o modelo VRML gerado pelo sistema WebCADbyFeatures.

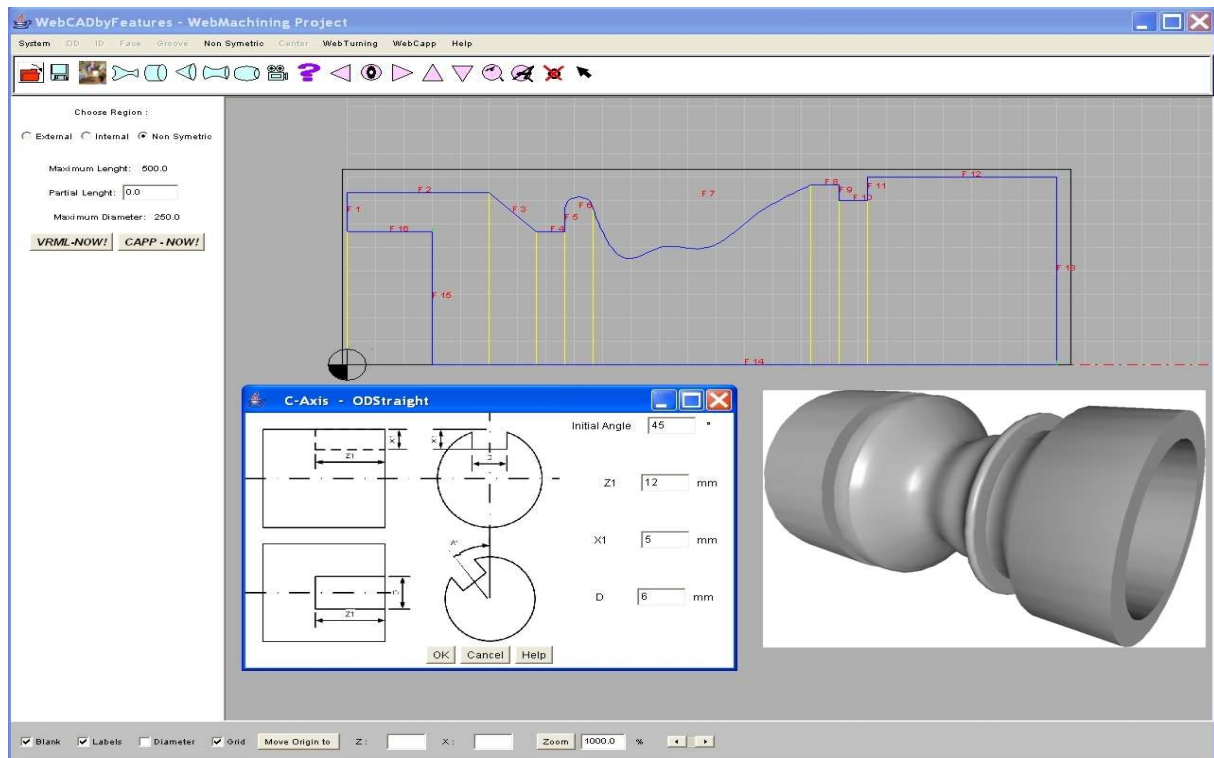


Figura 12.1: Modelagem torre utilizando *feature* Eixo C e o VRML associado às *features* concêntricas.

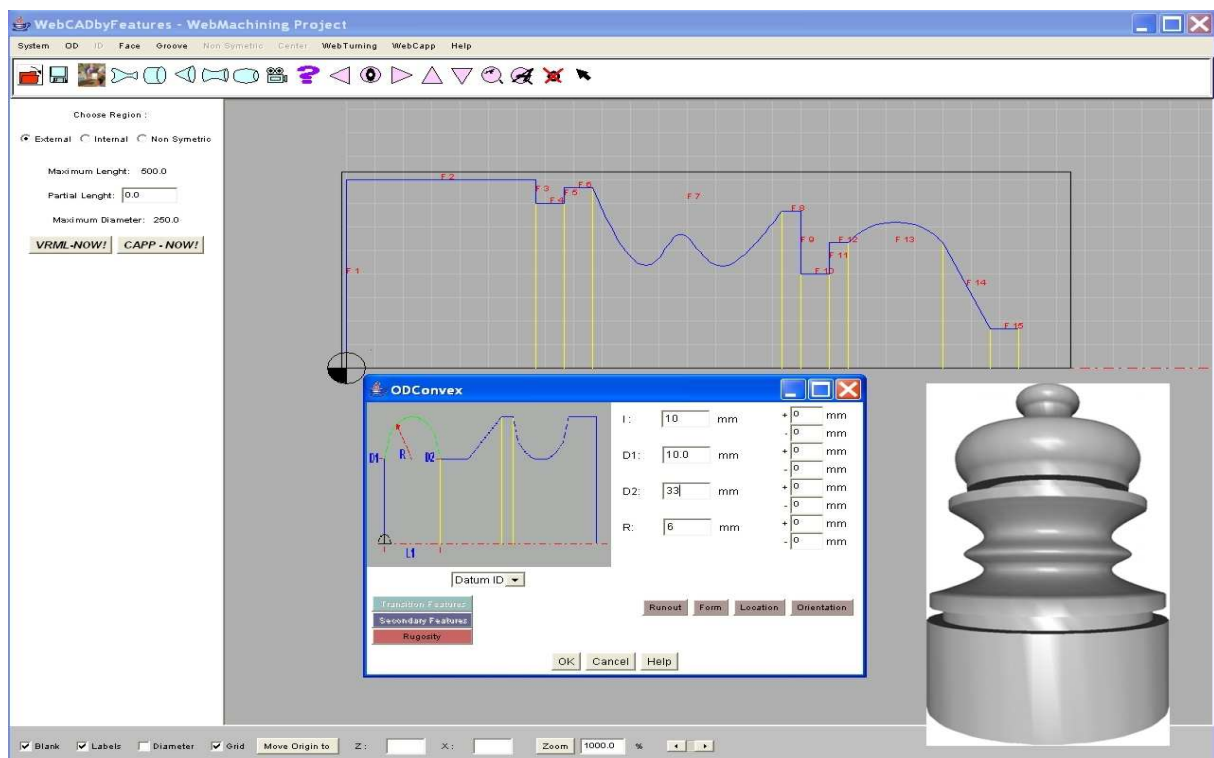


Figura 12.2: Modelagem bispo utilizando *feature* ODconvex e o VRML associado às *features* concêntricas.

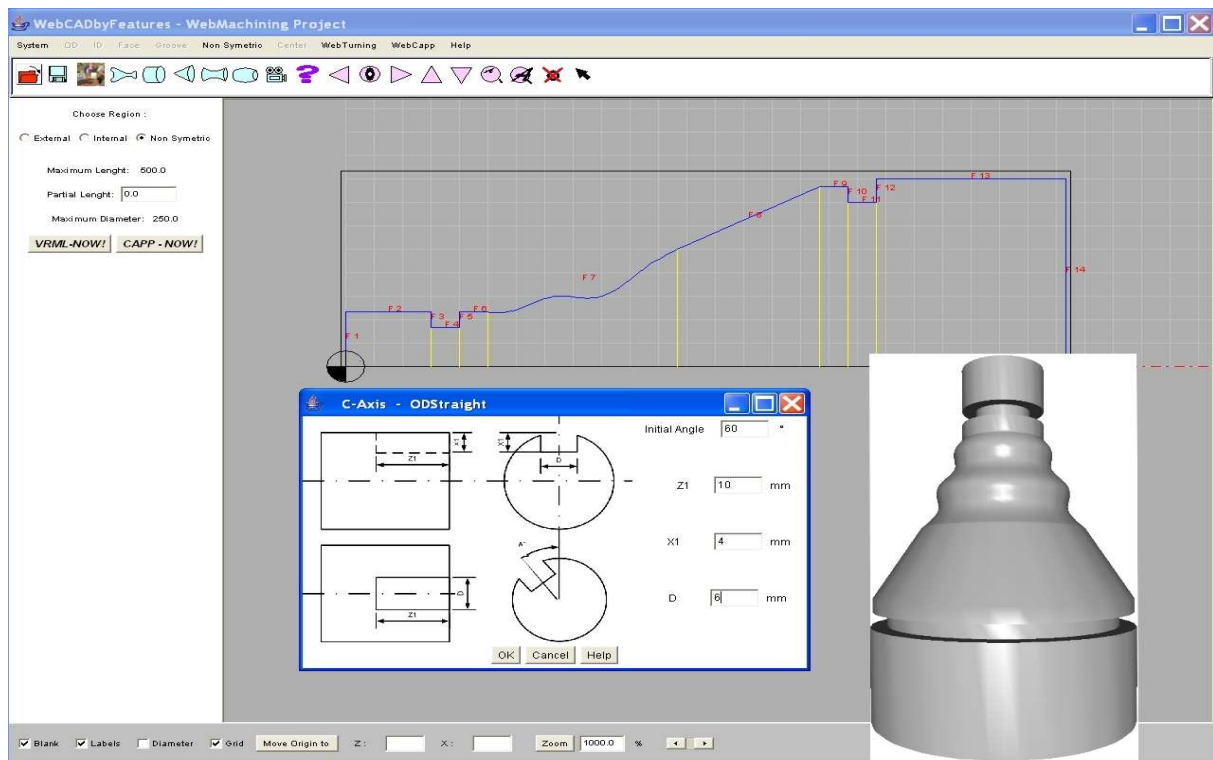


Figura 12.3: Modelagem do cavalo utilizando *feature* Eixo C e o VRML associado às *features* concêntricas.

O projeto do cavalo é apresentado na figura 12.3, destacando também a instanciação de uma *feature* de Eixo C. É apresentado o modelo VRML gerado pelo sistema WebCADbyFeatures.

O projeto do peão é apresentado na figura 12.4, destacando a instanciação de uma *feature spline*, e ao lado o modelo VRML gerado pelo sistema WebCADbyFeatures.

O projeto da rainha é apresentado na figura 12.5, sendo apresentado duas visões do modelo VRML gerado pelo sistema WebCADbyFeatures.

O projeto do rei e o modelo VRML gerados pelo sistema WebCADbyFeatures são apresentados na figura 12.6.

12.2.2 WebCAPP¹

O sistema WebCAPP é chamado via *servlets* ou *applet* gerando o plano de processo da usinagem da peça a partir do modelo de *features* de projeto, bem como o programa NC para o centro de torneamento Galaxy 15M.

O procedimento se inicia com a normalização do modelo de *features* da peça modelado pelo agente de interface WebCADbyFeatures. A figura 12.7 apresenta a saída do WebCAPP para este procedimento associado à peça torre. Nota-se o reconhecimento de *features* do tipo *groove* complexa e *spline virtual*, apresentando os pontos cartesianos de todas as *features* mapeadas.

A seguir tem início o procedimento de mapeamento de *features* baseado na decomposição orientada ao *setup*. A figura 12.8 apresenta a saída do WebCAPP para este método de decomposição

¹Serão apresentados alguns detalhes das saídas associadas aos algoritmos de algumas atividades do WebCAPP apenas para a peça torre.

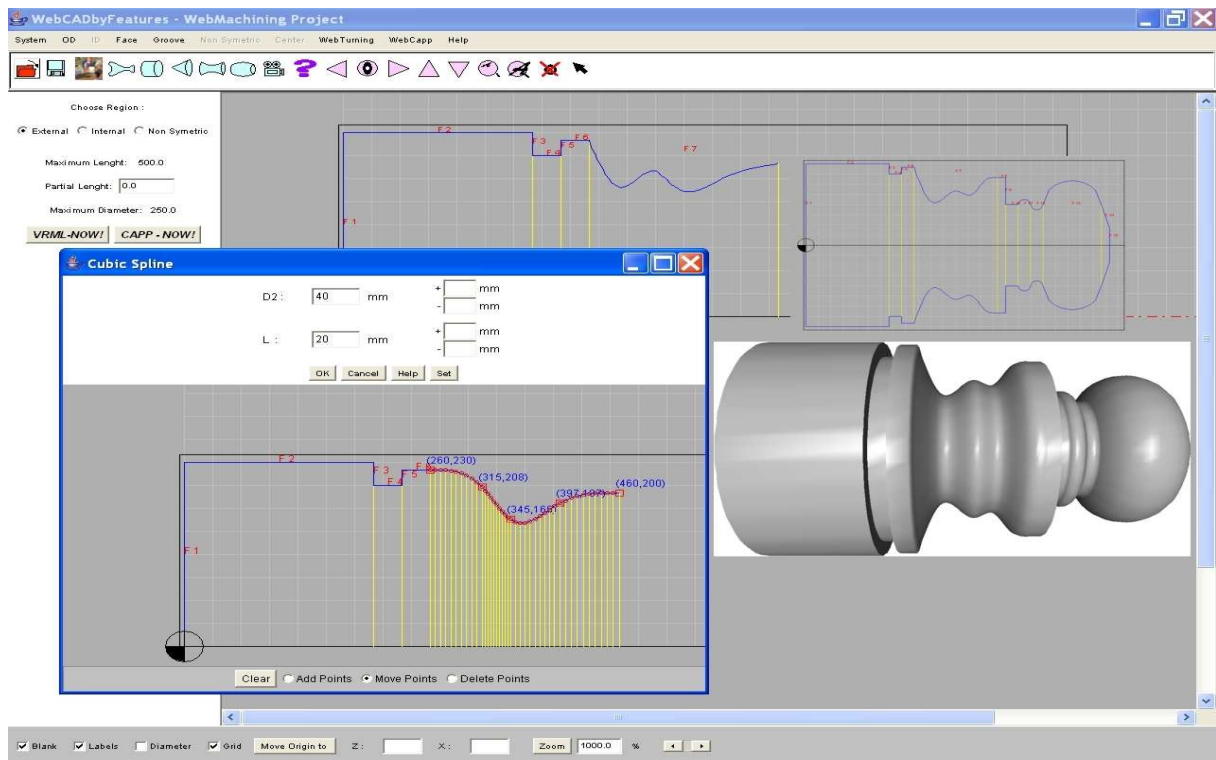


Figura 12.4: Modelagem do peão utilizando *feature spline* e o VRML associado às *features* concêntricas.

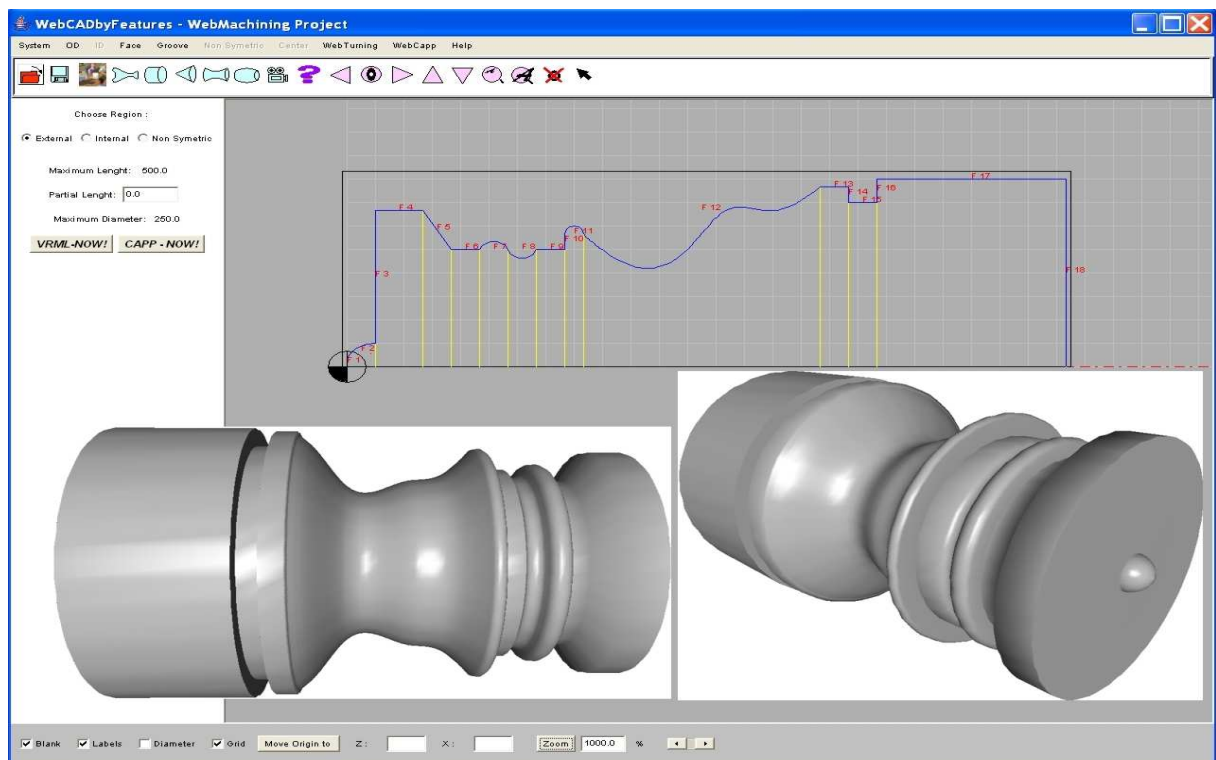


Figura 12.5: Modelagem rainha e duas visões do VRML associado às *features* concêntricas.

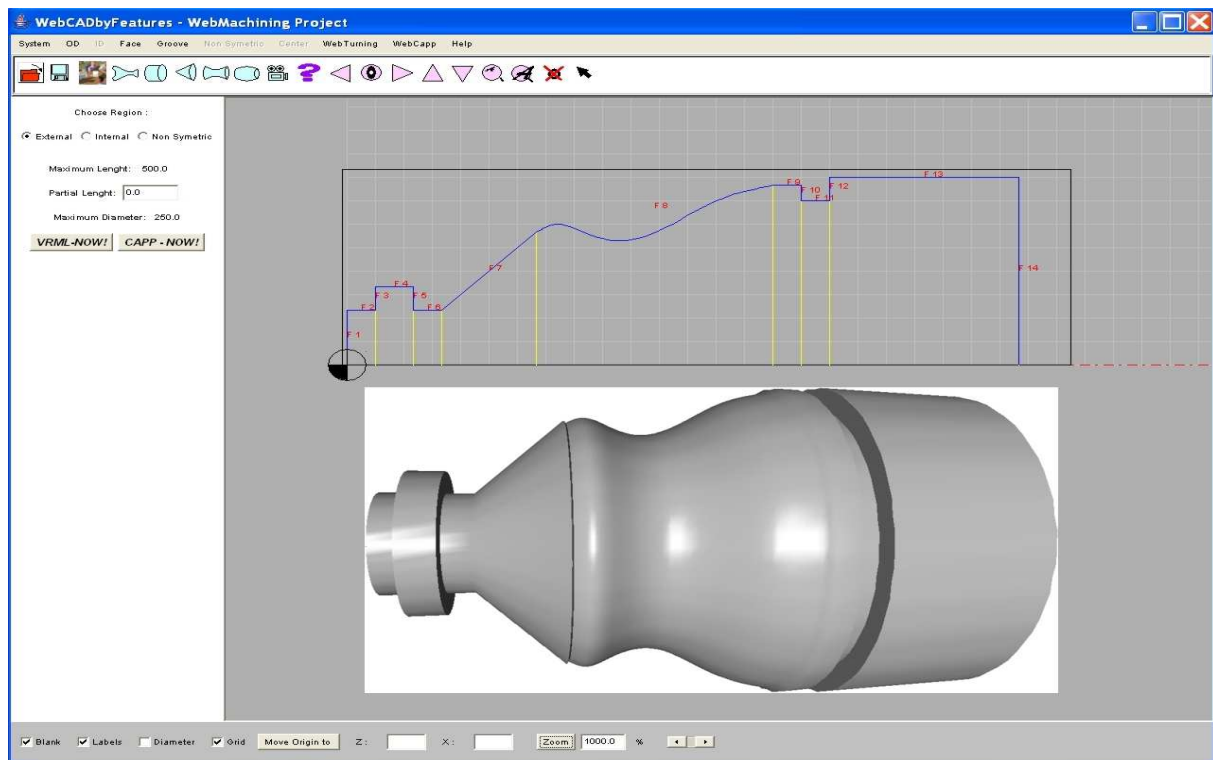


Figura 12.6: Modelagem do rei e VRML associado às *features* concêntricas.

associado à peça torre. São identificadas as *features* de fixação externas e internas, as superfícies delimitadoras (linhas delimitadoras), bem como a presença de *features* do tipo Eixo C, no caso quatro *keyways*.

O procedimento de mapeamento de *features* baseado na decomposição orientada à geometria é executado a seguir. A figura 12.9 apresenta a saída do WebCAPP para este método de decomposição associado à peça torre. São identificadas as *features* de fixação para cada *setup*, o tipo de *blank* existente no estoque e as *features* de usinagem e de torneamento que serão usinadas, em cada *setup*.

A seguir é realizado o procedimento de mapeamento de *features* baseado na decomposição orientada à operação de usinagem, determinando assim os *workingsteps* e um *workplan* não-linear. A figura 12.10 apresenta os *workingsteps* linearizados com todas as informações tecnológicas necessárias para o plano de processo associado à peça torre. São apresentados os dados associados às operações de usinagem para cada *feature* de usinagem, incluindo os planos de segurança e de aproximação, condições de usinagem, fluido de corte, ferramenta selecionada e seu número na torre de ferramentas (magazine) do Galaxy 15M e número do corretor de ferramenta associado, entre outras.

Na sequência o programa NC é gerado para a peça. A figura 12.11 apresenta o programa NC gerado para a peça torre. A figura 12.12 apresenta a simulação gráfica dos programas NC gerados pelo WebCAPP para as peças torre e rei. A figura 12.13 apresenta as simulações gráfica para as peças bispo, cavalo, peão e rainha. Estas simulações foram realizadas no próprio CNC do Galaxy 15M e visualizadas em um PC através do *software* CNCScreen da Fanuc.

```

FEATURE numero: 1
  Tipo: FaceStraight
  L1: 0.0
  D1: 35.0
  D2: 45.0
FEATURE numero: 2
  Tipo: ODStraight
  Tem Rosca: false
  L1: 15.0
  D1: 45.0
  D2: 45.0
FEATURE numero: 3
  Tipo: GrooveComplexo
  Features presentes:
    FEATURE numero: 1
      Tipo: ODTapered
      L1: 5.0
      D1: 45.0
      D2: 34.0
    FEATURE numero: 2
      Tipo: ODStraight
      Tem Rosca: false
      L1: 3.0
      D1: 34.0
      D2: 34.0
    FEATURE numero: 3
      Tipo: FaceStraight
      L1: 0.0
      D1: 34.0
      D2: 40.0
    FEATURE numero: 4
      Tipo: ODConvex
      L1: 3.0
      D1: 40.0
      D2: 40.0
    FEATURE numero: 5
      Tipo: Nula
      L1: 0.0
      D1: 40.0
      D2: 46.0
  L1: 11.0
  D1: 45.0
  D2: 46.0
FEATURE numero: 4
  Tipo: OSplineVirtual
  Z: 0.0 X: 40.0
  Z: 0.20000076293945312 X: 38.83993148803711
  Z: 0.1999988555908203 X: 37.6895637512207
  Z: 0.20000076293945312 X: 36.55860137939453
  Z: 0.1999988555908203 X: 35.45673751831055
  Z: 0.20000076293945312 X: 34.393680572509766
  Z: 0.20000076293945312 X: 33.379127502441406
  Z: 0.1999988555908203 X: 32.42278289794922
  Z: 0.20000076293945312 X: 31.534347534179688
  Z: 0.1999988555908203 X: 30.72351837158203
  Z: 0.20000076293945312 X: 30.0
  Z: 0.5 X: 27.7222843170166
  Z: 0.5 X: 27.291900634765625
  Z: 0.5 X: 27.226634979248047
  Z: 0.5 X: 27.446081161499023
  Z: 0.5 X: 27.869842529296875
  Z: 0.5 X: 28.417516708374023
  Z: 0.5 X: 29.00870132446289
  Z: 0.5 X: 29.562997817993164
  Z: 0.5 X: 30.0
  Z: 0.5 X: 30.259082794189453
  Z: 0.5 X: 30.358684539794922
  Z: 0.5 X: 30.337026596069336
  Z: 0.5 X: 30.232315063476562
  Z: 0.5 X: 30.0827693939209
  Z: 0.5 X: 29.926605224609375
  Z: 0.5 X: 29.80202865600586
  Z: 0.5 X: 29.74725914001465
  Z: 0.5 X: 29.800512313842773
  Z: 0.5 X: 30.0
  Z: 0.5 X: 30.373489379882812
  Z: 0.5 X: 30.906972885131836
  Z: 0.5 X: 30.906972885131836
  Z: 0.5 X: 31.57599449157715
  Z: 0.5 X: 32.356101989746094
  Z: 0.5 X: 33.22283172607422
  Z: 0.5 X: 34.15174102783203
  Z: 0.5 X: 35.11836624145508
  Z: 0.5 X: 36.0982551574707
  Z: 0.5 X: 37.06694793701172
  Z: 0.5 X: 38.0
  Z: 0.5999984741210938 X: 39.04665756225586
  Z: 0.6000022888183594 X: 40.01542282104492
  Z: 0.5999984741210938 X: 40.91495132446289
  Z: 0.6000022888183594 X: 41.7538948059082
  Z: 0.5999984741210938 X: 42.540916442871094
  Z: 0.5999984741210938 X: 43.2846565246582
  Z: 0.6000022888183594 X: 43.99378967285156
  Z: 0.5999984741210938 X: 44.67695236206055
  Z: 0.6000022888183594 X: 45.342803955078125
  Z: 0.5999984741210938 X: 46.0
  L1: 23.0
  D1: 46.0
  D2: 46.0
FEATURE numero: 5
  Tipo: Nula
  L1: 0.0
  D1: 46.0
  D2: 47.0
FEATURE numero: 6
  Tipo: ODStraight
  Tem Rosca: false
  L1: 3.0
  D1: 47.0
  D2: 47.0
FEATURE numero: 7
  Tipo: GrooveComplexo
  Features presentes:
    FEATURE numero: 1
      Tipo: FaceStraight
      L1: 0.0
      D1: 47.0
      D2: 42.0
    FEATURE numero: 2
      Tipo: ODStraight
      Tem Rosca: false
      L1: 3.0
      D1: 42.0
      D2: 42.0
    FEATURE numero: 3
      Tipo: FaceStraight
      L1: 0.0
      D1: 42.0
      D2: 48.0
  L1: 3.0
  D1: 47.0
  D2: 48.0
FEATURE numero: 8
  Tipo: ODStraight
  Tem Rosca: false
  L1: 20.0
  D1: 48.0
  D2: 48.0
FEATURE numero: 9
  Tipo: FaceStraight
  L1: 0.0
  D1: 48.0
  D2: 0.0
FEATURE numero: 10
  Tipo: IDStraight
  L1: 66.0
  D1: 0.0
  D2: 0.0
FEATURE numero: 11
  Tipo: FaceStraight
  L1: 0.0
  D1: 0.0
  D2: 35.0
FEATURE numero: 12
  Tipo: IDStraight
  L1: 9.0
  D1: 35.0
  D2: 35.0

```

Figura 12.7: Procedimento de normalização realizado pelo WebCAPP identificando as *features* de projeto que serão mapeadas.


```
#####
DECOMPOSICAO ORIENTADA A SETUP:

Features de Fixacao possiveis:
  Feature numero: 10
  Feature numero: 12
  Feature numero: 2
  Feature numero: 8

Superficies Delimitadoras Externas
  Superficies Delimitadoras Externas Possiveis:
    Feature numero: 7
    Feature numero: 8
  Superficie Delimitadora Externa Utilizada: 8

Superficies Delimitadoras Internas
  Superficies Delimitadoras Internas Possiveis:
    Feature numero: 10
  Superficie Delimitadora Interna Utilizada: 10
#####
#NAME?
#NAME?
#NAME?
48.0
Existem 4 features no EIXO C:
Inicio: 0
Fim: 8

KeyWay 2
KeyWay 2
KeyWay 2
KeyWay 2
```

Figura 12.8: Decomposição orientada ao *setup* com a identificação das *features* de fixação, superfícies delimitadoras e *features* de Eixo C.

12.2.3 WebTurning

As figuras 12.14, 12.15 e 12.16 apresentam as GUIs WebDNC que mostram as interfaces de teleoperação para as peças xadrez, fixações das peças, programa NC, telas do CNC, *status* do CNC, programas disponíveis na memória do CNC, entre outras informações pertinentes à interface de teleoperação. As seis peças fabricadas em *nylon* são apresentadas na figura 12.17. A figura 12.18 apresenta detalhes da torre e da rainha.

12.3 Peça com Tolerâncias

12.3.1 WebCADbyFeatures

O projeto da peça com duas *features* de tolerâncias de cilindricidade e duas *features* de referência (referência “A” associada à *feature* OD-F18 e referência “M” associada à *feature* ID-F35) é apresentado na figura 12.19, sendo denominado peça com tolerâncias geométricas. Também é apresentado o modelo VRML gerado pelo sistema WebCADbyFeatures. A peça apresenta *features* concêntricas de região externa e região interna, incluindo rosca, ODs, IDs, arcos, *features* de transição, cones e canais, totalizando 35 *features*. O *blank* é vazado, ou seja, é do tipo tubo.

12.3.2 WebCAPP

O procedimento de mapeamento de *features* gera um plano de processo com dois *setups*, sendo que no *Setup1* ocorre a usinagem das *features* externas F1 a F5 e das *features* internas F32 a F35, incluindo a *feature* de referência M e a *feature* de tolerância de cilindricidade associada. No *Setup2* ocorre a usinagem das *features* externas F6 a F24 e das *features* internas F25 a F31, incluindo a *feature* de referência A associada à *feature* de tolerância de cilindricidade.

```

#####
DECOMPOSICAO ORIENTADA A GEOMETRIA:

Blank Necessario:
    D1: 0.0
    D2: 50.0
    L1: 79.0

Blank Utilizado (Existente no estoque):
    D1: 0.0
    D2: 50.0
    L1: 79.0

Blank Setup1:
    D1: 33.0
    D2: 50.0
    L1: 7.0

SETUP numero: 1
FEATURE DE USINAGEM numero: 1
Features:
    Tipo: 0
        L1: 7.0
        D1: 33.0
        D2: 33.0
    Tipo: 0
        L1: 0.0
        D1: 33.0
        D2: 0.0
FEATURE DE USINAGEM numero: 2
    Tipo: Face
    Origem: Z= 0.0 X= 33.0
    Features:
        Tipo: 0
            L1: 0.0
            D1: 33.0
            D2: 35.0
        Tipo: 6
            L1: 0.0
            D1: 35.0
            D2: 45.0
        Tipo: 0
            L1: 0.0
            D1: 45.0
            D2: 50.0
        Tipo: 0
            L1: -2.0
            D1: 50.0
            D2: 50.0
FEATURE DE USINAGEM numero: 3
    Tipo: Bloco Externo
    Origem: Z= 0.0 X= 45.0
    Features:
        Tipo: 1
            L1: 15.0
            D1: 45.0
            D2: 45.0
        Tipo: 0
            L1: 11.0
            D1: 45.0
            D2: 46.0
        Tipo: 32
            D1: 46.0
            D2: 47.0
        Tipo: 1
            L1: 3.0
            D1: 47.0
            D2: 47.0
        Tipo: 0
            L1: 3.0
            D1: 47.0
            D2: 48.0
        Tipo: 1
            L1: 20.0
            D1: 48.0
            D2: 48.0
        Tipo: 0
            L1: 0.0
            D1: 48.0
            D2: 50.0

FEATURE DE USINAGEM numero: 4
    Tipo: GrooveComplexo Externo
    Origem: Z= 15.0 X= 45.0
    Features:
        Tipo: 31
            Features incluidas:
                FEATURE numero: 1
                    Tipo: ODTapered
                    L1: 5.0
                    D1: 45.0
                    D2: 34.0
                FEATURE numero: 2
                    Tipo: ODStraight
                    Tem Rosca: fe
                    L1: 3.0
                    D1: 34.0
                    D2: 34.0
                FEATURE numero: 3
                    Tipo: FaceStraight
                    L1: 0.0
                    D1: 34.0
                    D2: 40.0
                FEATURE numero: 4
                    Tipo: ODConvex
                    L1: 3.0
                    D1: 40.0
                    D2: 40.0
                FEATURE numero: 5
                    Tipo: Nula
                    L1: 0.0
                    D1: 40.0
                    D2: 46.0
            L1: 11.0
            D1: 45.0
            D2: 46.0

FEATURE DE USINAGEM numero: 5
    Tipo: GrooveComplexo Externo
    Origem: Z= 52.0 X= 47.0
    Features:
        Tipo: 31
            Features incluidas:
                FEATURE numero: 1
                    Tipo: FaceStraight
                    L1: 0.0
                    D1: 47.0
                    D2: 42.0
                FEATURE numero: 2
                    Tipo: ODStraight
                    Tem Rosca: fe
                    L1: 3.0
                    D1: 42.0
                    D2: 42.0
                FEATURE numero: 3
                    Tipo: FaceStraig
                    L1: 0.0
                    D1: 42.0
                    D2: 48.0
            L1: 3.0
            D1: 47.0
            D2: 48.0

FEATURE DE USINAGEM numero: 6
    Tipo: Spline Externa
    Origem: Z= 26.0 X= 40.0
    Features:
        Tipo: 32
            L1: 23.0
            D1: 46.0
            D2: 46.0

FEATURE DE USINAGEM numero: 7
    Tipo: Bloco Interno
    Origem: Z= 0.0 X= 35.0
    Features:
        Tipo: 11
            L1: 9.0
            D1: 35.0
            D2: 35.0
        Tipo: 0
            L1: 0.0
            D1: 35.0
            D2: 33.0

FEATURE DE USINAGEM numero: 8
    Tipo: Rasgo de Chaveta
    Origem: Z= 0.0 X= 45.0
    Features:
        Tipo: 9
            Tipo: Rasgo de Chaveta
            Origem: Z= 0.0 X= 45.0
            Features:
                Tipo: 10
                    Tipo: Rasgo de Chaveta
                    Origem: Z= 0.0 X= 45.0
                    Features:
                        Tipo: 11
                            Tipo: Rasgo de Chaveta
                            Origem: Z= 0.0 X= 45.0
                            Features:
                                #####
                                pro: 2
                                FEATURE DE USINAGEM numero: 1
                                    Tipo: Face
                                    Origem: Z= 0.0 X= 0.0
                                    Features:
                                        Tipo: 6
                                            L1: 0.0
                                            D1: 0.0
                                            D2: 48.0
                                        Tipo: 0
                                            L1: 0.0
                                            D1: 48.0
                                            D2: 50.0
                                        Tipo: 0
                                            L1: -2.0
                                            D1: 50.0
                                            D2: 50.0
                                FEATURE DE USINAGEM numero: 2
                                    Tipo: Bloco Interno
                                    Origem: Z= 0.0 X= 0.0
                                    Features:
                                        Tipo: 0
                                            L1: 0.0
                                            D1: 48.0
                                            D2: 50.0

```

Figura 12.9: Decomposição orientada à geometria.

DECOMPOSICAO ORIENTADA A OPERACAO:

--> Ainda tem que imprimir os dados da operacao <--

SETUP numero: 1

WORKINGSTEP numero: 1
 FeatureDeUsinagem Tipo: 13
 PlanoDeSeguranca: Z= 87.0 X= 0.0
 PlanoDeAproximacao: Z= 82.0 X= 0.0
 Operacoes de Usinagem:
 Tipo: Ainda nao foi tratado
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Numero da Magazine: 4
 Numero de Compensacao: 4

WORKINGSTEP numero: 2
 FeatureDeUsinagem Tipo: 1
 PlanoDeSeguranca: Z= 89.0 X= 60.0
 Tipo: Faceamento
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: DESBASTE GERAL EXTERN
 Numero da Magazine: 7
 Numero de Compensacao: 7

WORKINGSTEP numero: 3
 FeatureDeUsinagem Tipo: 2
 PlanoDeSeguranca: Z= 87.0 X= 60.0
 PlanoDeAproximacao: Z= 82.0 X= 50.0
 Operacoes de Usinagem:
 Tipo: Torneamento Externo
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: DESBASTE GERAL EXTERN
 Numero da Magazine: 7
 Numero de Compensacao: 7

WORKINGSTEP numero: 4
 FeatureDeUsinagem Tipo: 6
 PlanoDeSeguranca: Z= 72.0 X= 55.0
 PlanoDeAproximacao: Z= 62.0 X= 50.0
 Operacoes de Usinagem:
 Tipo: Sangramento Externo
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: SANGRAMENTO EXTERNO
 Numero da Magazine: 3
 Numero de Compensacao: 3

WORKINGSTEP numero: 5
 FeatureDeUsinagem Tipo: 6
 PlanoDeSeguranca: Z= 35.0 X= 57.0
 PlanoDeAproximacao: Z= 25.0 X= 52.0
 Operacoes de Usinagem:
 Tipo: Sangramento Externo
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: SANGRAMENTO E
 Numero da Magazine: 3
 Numero de Compensacao: 3

WORKINGSTEP numero: 6
 FeatureDeUsinagem Tipo: 11
 PlanoDeSeguranca: Z= 61.0 X= 56.0
 PlanoDeAproximacao: Z= 56.0 X= 46.0
 Operacoes de Usinagem:
 Tipo: Ainda nao foi tratado
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: SPLINE INTERNO
 Numero da Magazine: 11
 Numero de Compensacao: 11

WORKINGSTEP numero: 7

FeatureDeUsinagem Tipo: 3
 PlanoDeSeguranca: Z= 87.0 X= 23.0
 PlanoDeAproximacao: Z= 82.0 X= 33.0
 Operacoes de Usinagem:
 Tipo: Torneamento Interno
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: DESBASTE GERAL INTERN
 Numero da Magazine: 9
 Numero de Compensacao: 9

WORKINGSTEP numero: 8

FeatureDeUsinagem Tipo: 21
 PlanoDeSeguranca: Z= 87.0 X= 55.0
 PlanoDeAproximacao: Z= 82.0 X= 45.0
 Operacoes de Usinagem:
 Tipo: Ainda nao foi tratado
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: FRESAMENTO EXTERNO
 Numero da Magazine: 10
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: FRESAMENTO EXTERNO
 Numero da Magazine: 10
 Numero de Compensacao: 10

WORKINGSTEP numero: 10

FeatureDeUsinagem Tipo: 21
 PlanoDeSeguranca: Z= 87.0 X= 55.0
 PlanoDeAproximacao: Z= 82.0 X= 45.0
 Operacoes de Usinagem:
 Tipo: Ainda nao foi tratado
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: FRESAMENTO EXTERNO
 Numero da Magazine: 10
 Numero de Compensacao: 10

WORKINGSTEP numero: 11

FeatureDeUsinagem Tipo: 21
 PlanoDeSeguranca: Z= 87.0 X= 55.0
 PlanoDeAproximacao: Z= 82.0 X= 45.0
 Operacoes de Usinagem:
 Tipo: Ainda nao foi tratado
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: FRESAMENTO EXTERNO
 Numero da Magazine: 10
 Numero de Compensacao: 10

tero: 2

WORKINGSTEP numero: 1

FeatureDeUsinagem Tipo: 1
 PlanoDeSeguranca: Z= 89.0 X= 60.0
 PlanoDeAproximacao: Z= 79.0 X= 55.0
 Operacoes de Usinagem:
 Tipo: Faceamento
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: DESBASTE GERAL EXTERNO
 Numero da Magazine: 7
 Numero de Compensacao: 7

WORKINGSTEP numero: 2

FeatureDeUsinagem Tipo: 3
 PlanoDeSeguranca: Z= 87.0 X= 40.0
 PlanoDeAproximacao: Z= 82.0 X= 50.0
 Operacoes de Usinagem:
 Tipo: Torneamento Interno
 Tecnologia De Usinagem:
 Numero de passes: 3
 Fluido de corte: true
 Ap: 2.0
 F: 0.2
 Vc: 300.0
 Ferramentas Possiveis:
 Descricao: DESBASTE GERAL INTERNO
 Numero da Magazine: 9
 Numero de Compensacao: 9

Figura 12.10: Decomposição orientada à operação.

```

O2300(TORRE)
N10G21G40G90G95
N20M36
N30M37
N40(INICIO SETUP1)
N60G53
N70G40
N80G0X390Z350T00
N90T0404(Tool FURACAO)
N100G54
N110G90
N120G97S800M15
N130(Operacao - CICLO FIXO G74 FURACAO)
N140X0Z87
N150X0Z82
N160G74R2
N170G74Z70Q15000F0.15
N170(WORKINGSTEP - FEATURE DE USINAGEM Fu2)
N180G53
N190G0X390Z350T00
N200T0707(Tool DESBASTE GERAL EXTERNO)
N210G54
N220G96S300
N230G92S4000
N240M4
N250M8
N260(Operacao - CICLO FIXO G75 FACEAMENTO)
N270G0X60Z89
N280G0X55Z78.998
N290G75X28Z77P800000Q2000R1F0.2
N300(WORKINGSTEP - FEATURE DE TORNEAMENTO Ft3)
N310G0X50
N320(Operacao - CICLO FIXO G71-TIPO1 DESBASTE E G70 ACABAMENTO)
N330G0X60Z87
N340G0X50Z82
N350G71U1R2
N360G71P370Q470U0.2W0.2F0.2
N370G0X45
N380G42
N390G1W-5
N400G1W-15
N410X46W-11
N420W-23
N430X47
N440W-3
N450X48W-3
N460W-20
N470X50
N480G70P370Q470
N490(WORKINGSTEP - FEATURE DE TORNEAMENTO Ft4)
N500G53
N510G0X390Z350T00
N520T0303(Tool SANGRAMENTO EXTERNO)
N530G54
N540G96S300
N550G92S4000
N560M4
N570M8
N580(Operacao - CICLO FIXO G72 GROOVE COMPLEXO E G70 ACABAMENTO)
N590G0X55Z72
N600G0X50Z62
N610G72W0.6R0
N620G72P630Q700U0W0F0.2
N630G0Z51
N640G41
N650G1X45
N660G1X40
N670G2X40W3R1.5
N680G1X34
N690W3
N700X45W5
N710G70P630Q700
N720(WORKINGSTEP - FEATURE DE TORNEAMENTO Ft5)
N730G0X50
N740(Operacao - CICLO FIXO G72 GROOVE COMPLEXO E G70 ACABAMENTO)
N750G0X57Z35
N760G0X52Z25
N770G72W0.6R0
N780G72P790Q840U0W0F0.2
N790G0Z22
N800G41
N810G1X47
N820G1X42

N830W3
N840X47
N850G70P790Q840
N870G53
N880G0X390Z350T00
N890T1111(Tool SPLINE EXTERNO)
N900G54
N910G96S300
N920G92S4000
N930M4
N940M8
N950(Operacao - CICLO FIXO G71-TIPO2 DESBASTE E G70 ACABAMENTO)
N960G0X56Z61
N970G0X46Z56
N980G71U0.6R2

N990G71P1000Q1520U0.2W0F0.
N1000G0X40Z56
N1010G42
N1020G1W-5
N1030G1X38.84W-0.2
N1040X37.69W-0.2
N1050X36.559W-0.2
N1060X35.457W-0.2
N1070X34.394W-0.2
N1080X33.379W-0.2
N1090X32.423W-0.2
N1100X31.534W-0.2
N1110X30.724W-0.2
N1120X30W-0.2
N1130X28.598W-0.5
N1140X27.722W-0.5
N1150X27.292W-0.5
N1160X27.227W-0.5
N1170X27.446W-0.5
N1180X27.87W-0.5
N1190X28.418W-0.5
N1200X29.009W-0.5
N1210X29.563W-0.5
N1220X30W-0.5
N1230X30.259W-0.5
N1240X30.359W-0.5
N1250X30.337W-0.5
N1260X30.232W-0.5
N1270X30.083W-0.5
N1280X29.927W-0.5
N1290X29.802W-0.5
N1300X29.747W-0.5
N1310X29.801W-0.5
N1320X30W-0.5
N1330X30.373W-0.5
N1340X30.907W-0.5
N1350X31.576W-0.5
N1360X32.356W-0.5
N1370X33.223W-0.5
N1380X34.152W-0.5
N1390X35.118W-0.5
N1400X36.098W-0.5
N1410X37.067W-0.5
N1420X38W-0.5
N1430X39.047W-0.6
N1440X40.015W-0.6
N1450X40.915W-0.6
N1460X41.754W-0.6
N1470X42.541W-0.6

N2110(Operacao - CICLO MANUAL RASGO DE CHAVETA)
N2120G97S800M15
N2130M19
N2140G0X55Z87
N2150G28C0
N2160G0X34Z82C180
N2170G01Z67F0.16
N2180G0Z82
N2190M17
N2190(FIM SETUP1)
N2200(TROCA DE SETUP)
N2210G53

```

Figura 12.11: Programa NC gerado para a peça torre.

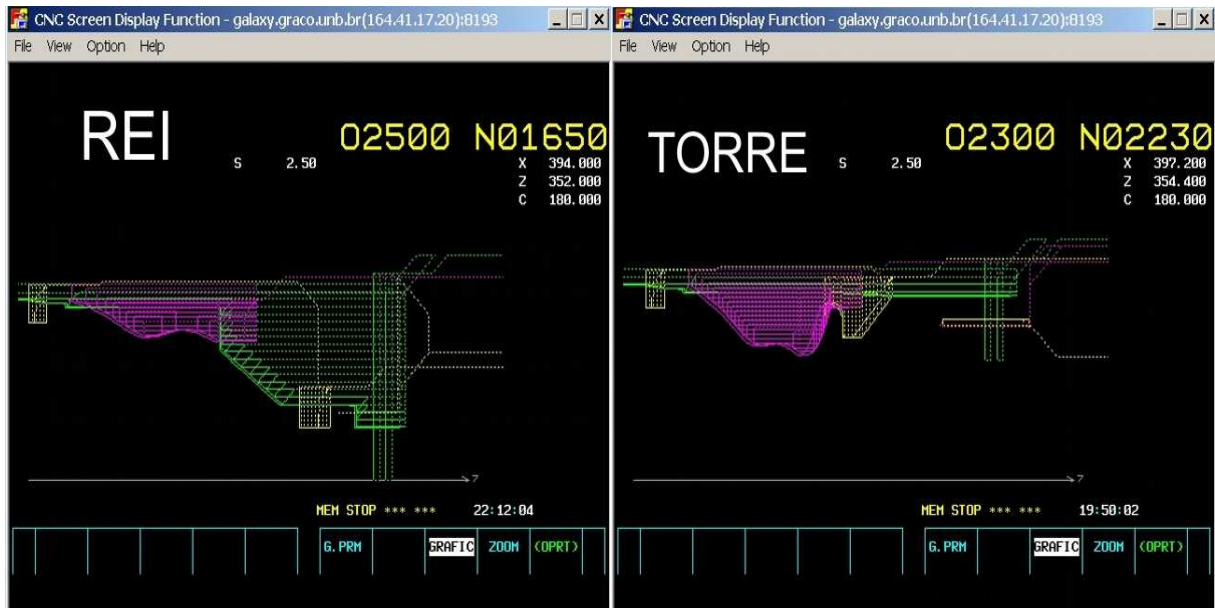


Figura 12.12: Simulação dos programas NC no CNC do Galaxy 15M para as peças rei e torre.

A figura 12.20 apresenta a simulação gráfica do programas NC gerado pelo WebCAPP para o *Setup1* e *Setup2*.

12.3.3 WebTurning

A figura 12.21 apresenta a GUI WebDNC que mostra uma tela com a monitoração do processo de usinagem da peça, fixação da peça no *Setup1* e *Setup2* e simulações. As figuras 12.22 e 12.23 apresentam as *features* de região externa e região interna, respectivamente, da peça usinada.

12.4 Peça Romi

12.4.1 WebCADbyFeatures

O projeto da peça romi é apresentada na figura 12.24. A geometria desta peça foi utilizada pelas Indústrias Romi no treinamento de operação do centro de torneamento Galaxy 15M ministrado em Brasília. A peça possui *features* concêntricas internas e externas, bem como rasgos radiais e furos axiais associados a Eixo C. Também são apresentados os modelo VRMLs gerados pelo sistema WebCADbyFeatures. O *blank* é vazado, ou seja, é do tipo tubo, sendo utilizado dois materiais na usinagem: latão e nylon.

12.4.2 WebCAPP

O procedimento de mapeamento de *features* gera um plano de processo com dois *setups*, também, sendo que no *Setup1* ocorre a usinagem das *features* externas F1 a F4 e das *features* internas F27 a F29. No *Setup2* ocorre a usinagem das *features* externas F4 a F23 e das *features* internas F24 a F26.

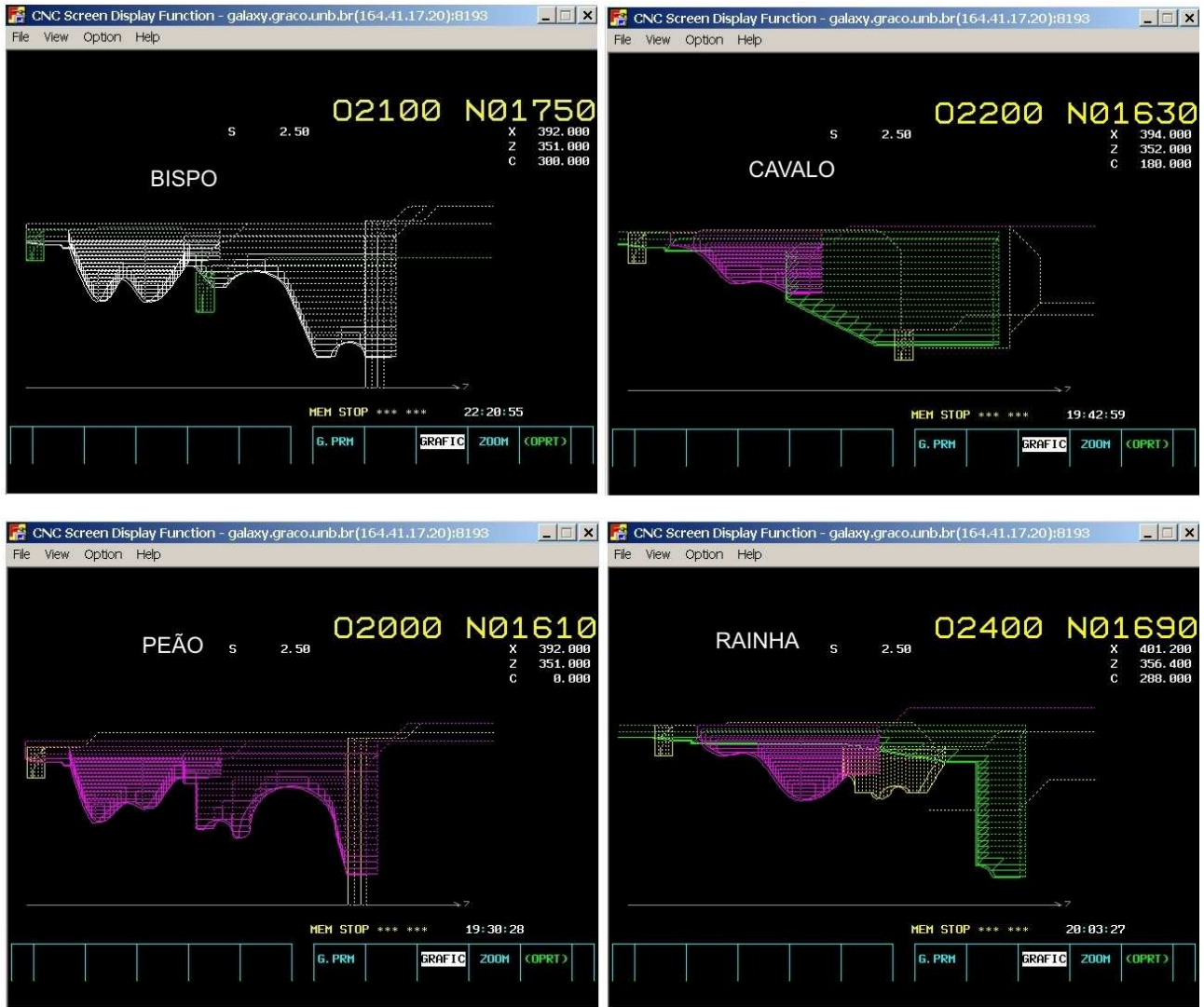


Figura 12.13: Simulação dos programas NC no CNC do Galaxy 15M para as peças bispo, cavalo, peão e rainha.

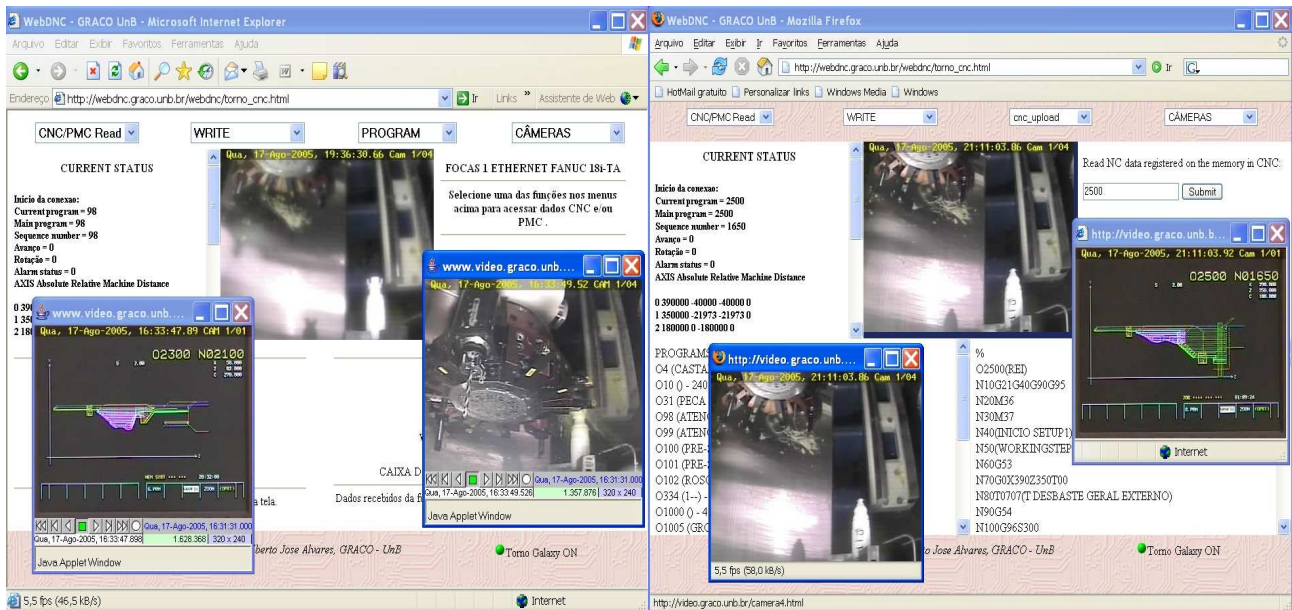


Figura 12.14: GUI WebDNC mostrando a fabricação da torre e realizando o *upload* do arquivo NC da peça rei para o Galaxy 15M. Também são apresentadas simulações e imagens da usinagem.

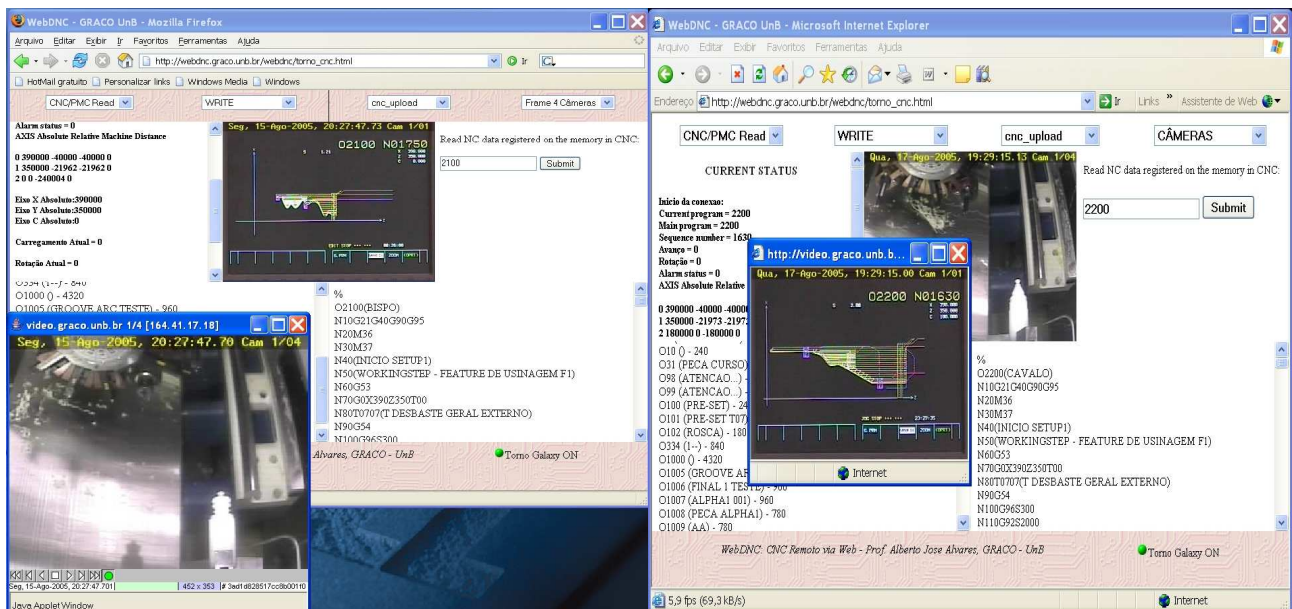


Figura 12.15: GUI WebDNC mostrando a usinagem do bispo e do cavalo.

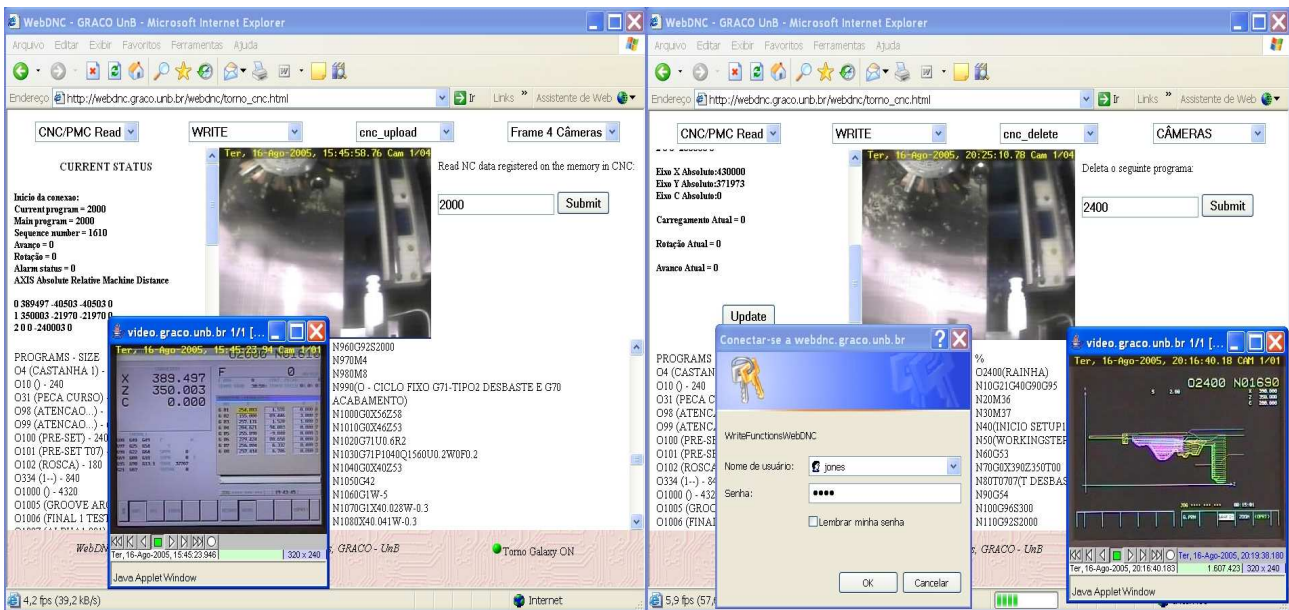


Figura 12.16: GUI WebDNC mostrando a usinagem do peão e da rainha e telas associadas ao *download* e *upload* de programas NC.



Figura 12.17: Peças do jogo de xadrez usinadas pelo Galaxy 15M.



Figura 12.18: Detalhes das peças torre e rainha.

A figura 12.25 apresenta a simulação gráfica do programas NC gerado pelo WebCAPP para o *Setup1* e *Setup2*.

12.4.3 WebTurning

A figura 12.26 apresenta a GUI WebDNC mostrando uma tela de monitoração do processo de usinagem da peça. As outras telas apresentam o *status* do CNC, programas disponíveis na memória do CNC, entre outras informações pertinentes à interface de teleoperação. As figuras 12.27 e 12.28 apresentam detalhes das *features* de região externa e região interna, da peça romi, usinadas em *nylon* e latão.

12.5 Peça Complexa

12.5.1 WebCADbyFeatures

O projeto da peça complexa constituída por *features* concêntricas e não-concêntricas é apresentado na figura 12.29, destacando-se um menu associado à *features* de Eixo C, padrão de furos Axial, definido sobre a *feature* de face F18.

A figura 12.30 apresenta os modelos VRML (*wireframe* e isométrico) gerado pelo sistema WebCADbyFeatures. A peça apresenta *features* concêntricas de região externa e região interna (rosca, ODs, IDs, canais, furo, arcos, *spline*, *features* de transição e cones) e *features* de Eixo C (rasgo radial e padrão de furos axial) totalizando 26 *features*, 20 concêntricas e 6 Eixo C. O *blank* é do tipo tarugo.

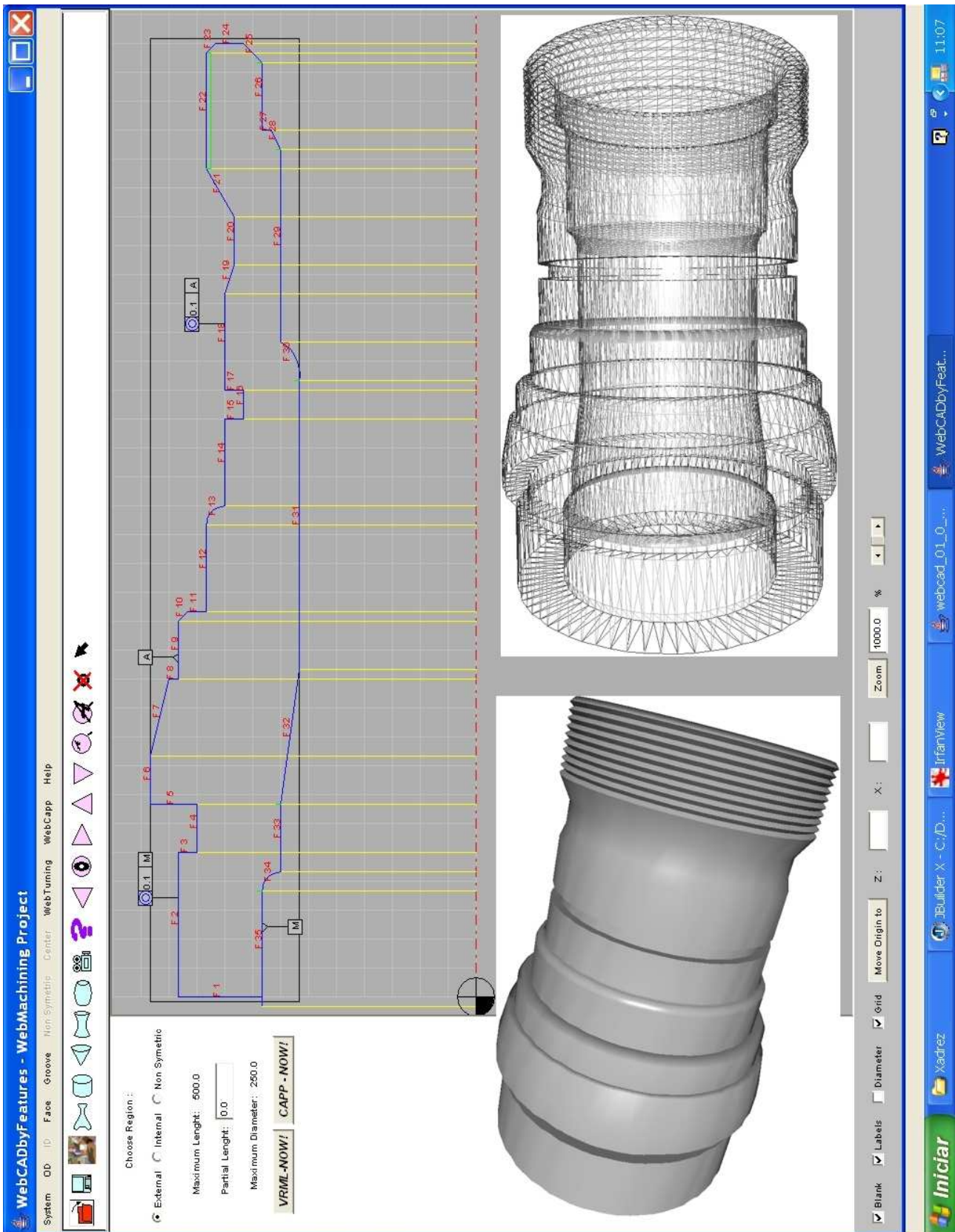


Figura 12.19: Modelagem de peça com *features* de tolerâncias de cilíndricidade e *features* de referências A e M e os VRMLs associados.

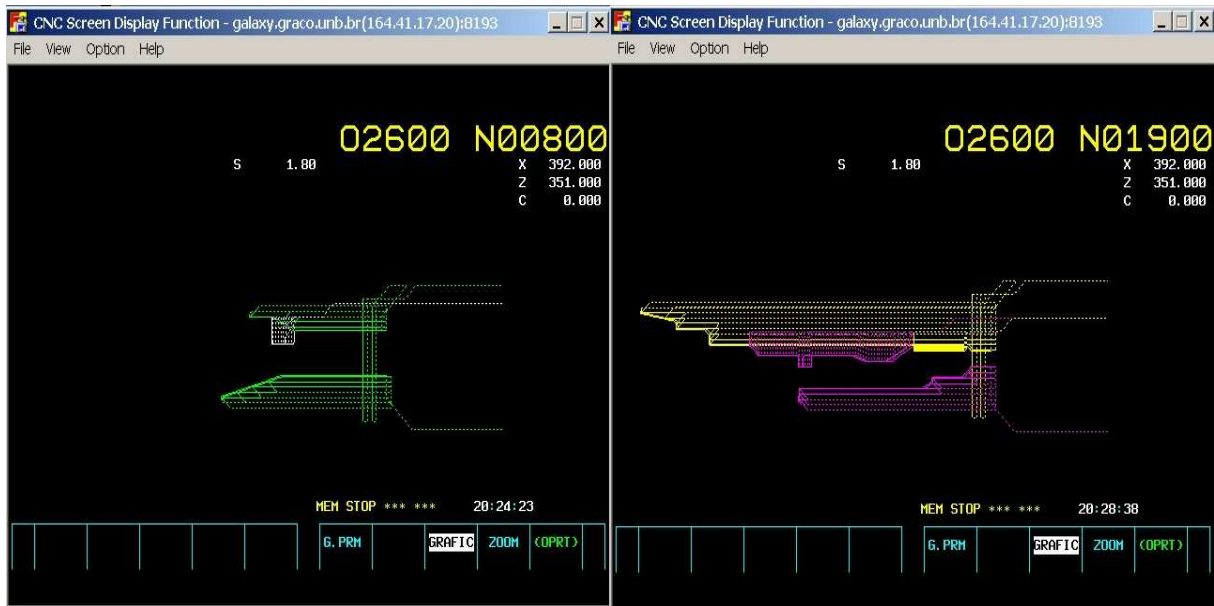


Figura 12.20: Simulação dos programas NC no CNC do Galaxy 15M para a peça tolerância no *Setup1* e *Setup2*.

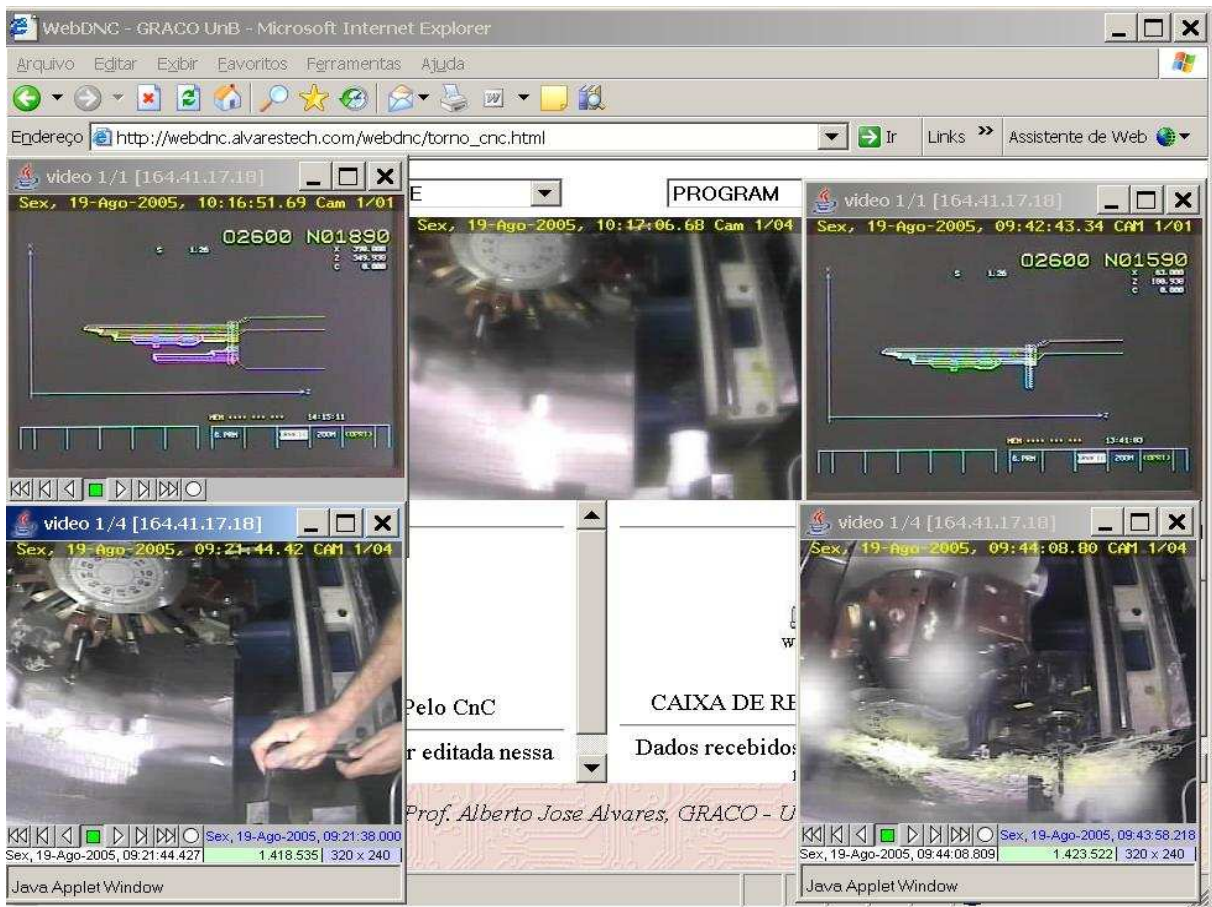


Figura 12.21: GUI WebDNC realizando a monitoração da usinagem da peça tolerância.



Figura 12.22: *Features* externas da peça tolerância.



Figura 12.23: *Features* internas da peça tolerância, lado esquerdo e lado direito.

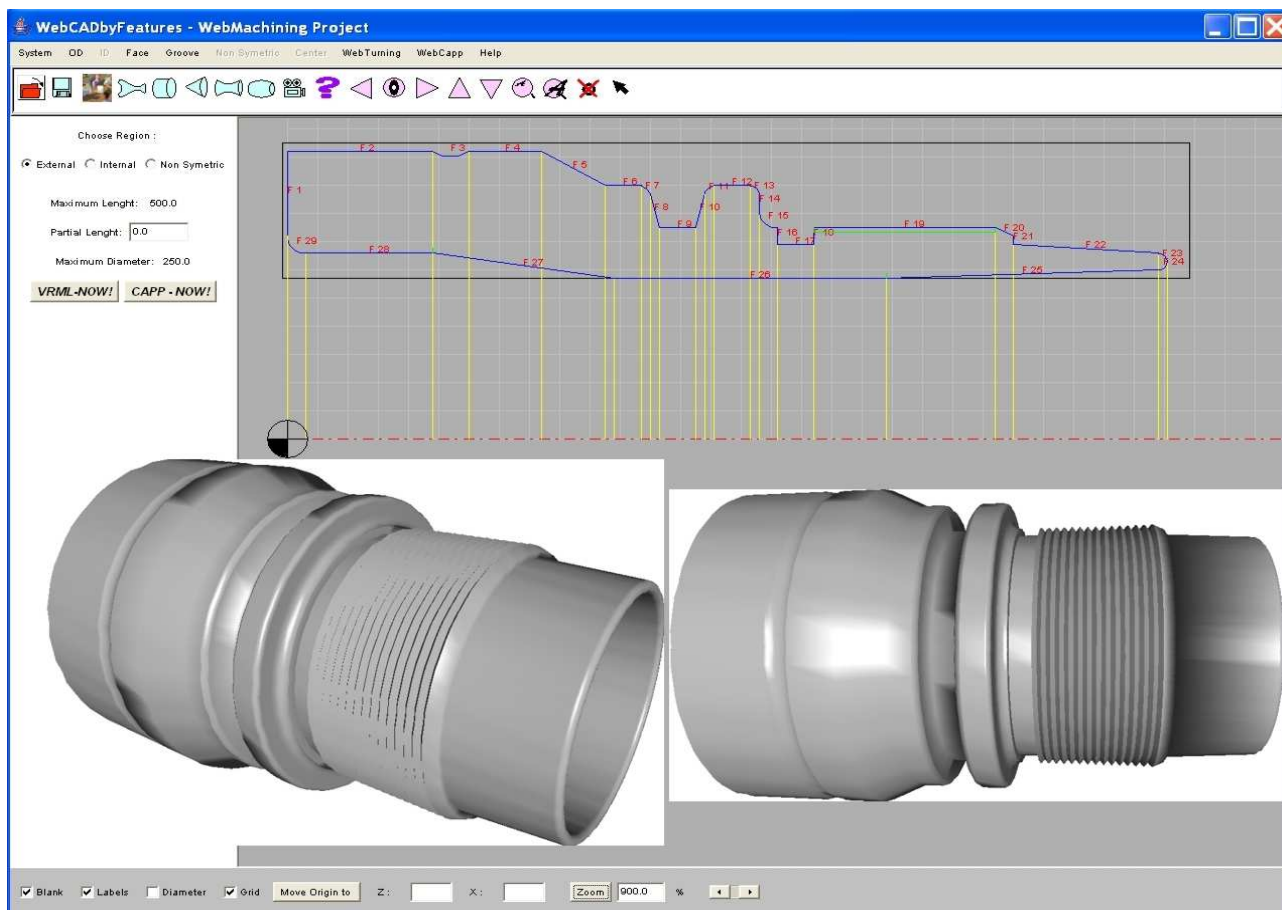


Figura 12.24: Modelagem da peça romi constituída por *features* concêntricas e não-concêntricas, e os modelos VRMLs associados destacando-se a rosca métrica e os canais.



Figura 12.25: Simulação dos programas NC no CNC do Galaxy 15M para a peça romi no *Setup1* e *Setup2*.

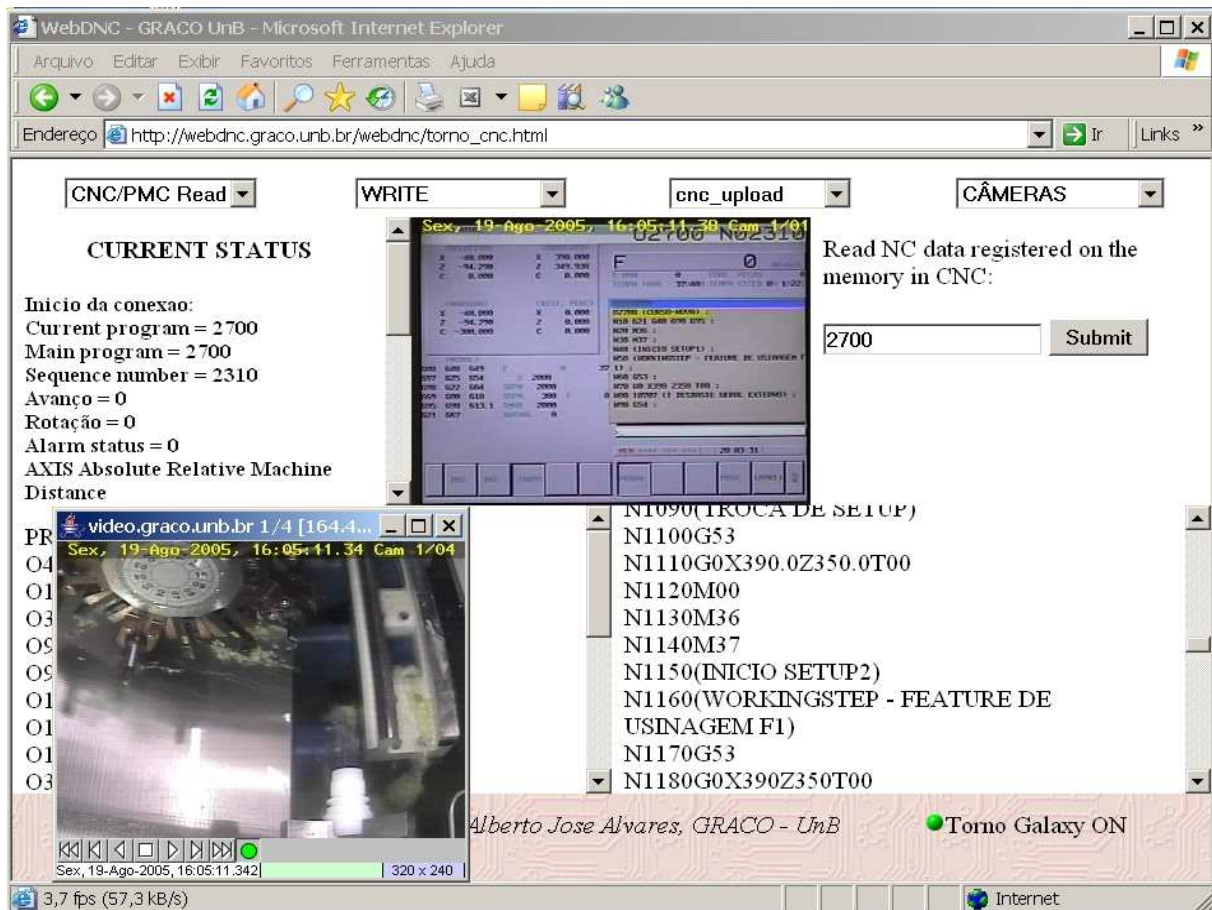


Figura 12.26: GUI WebDNC realizando a monitoração da usinagem da peça romi.



Figura 12.27: Detalhes *features* internas e externas da peça romi.



Figura 12.28: *Features* externas (lado esquerdo e direito) da peça romi.

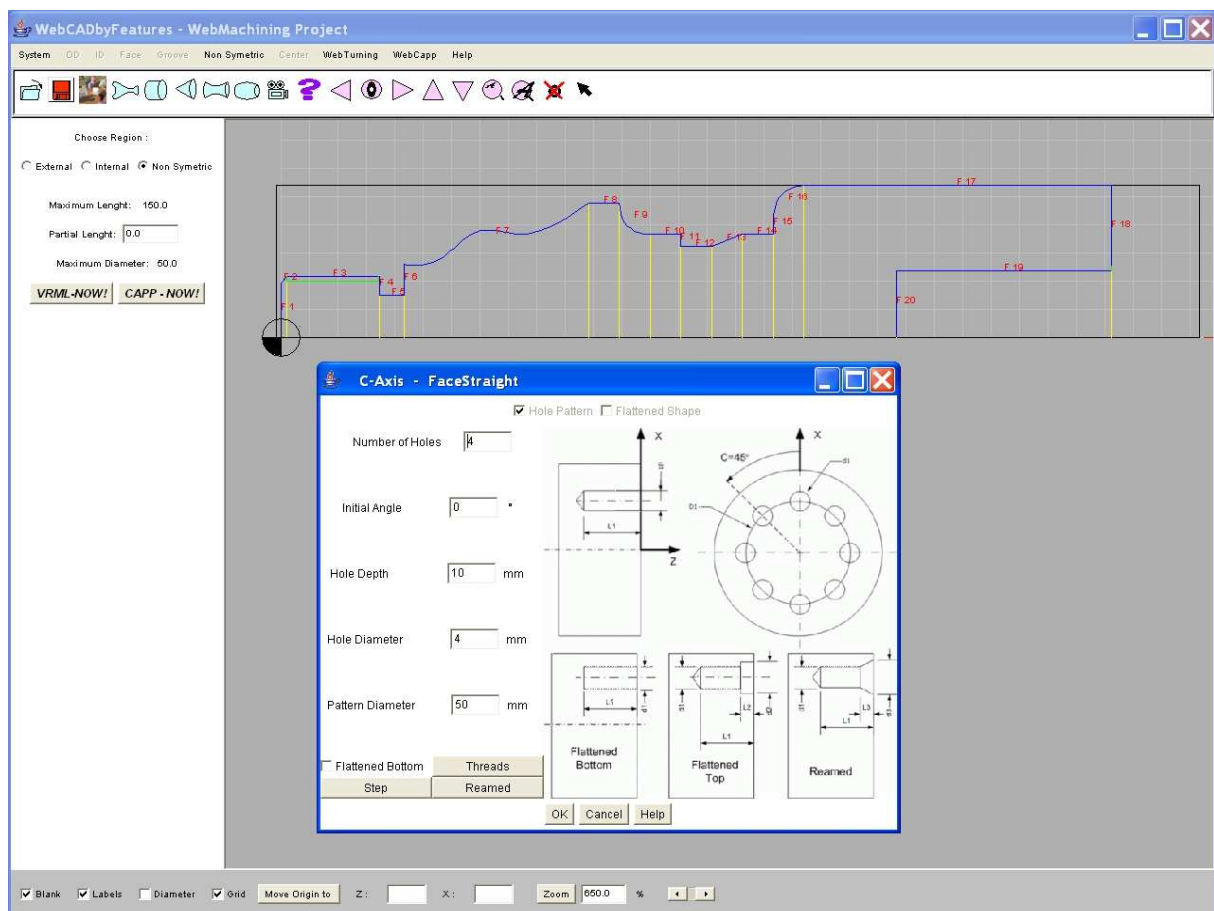


Figura 12.29: Modelagem da peça complexa com *features* de Eixo C, padrão de furos Axial, associada à *feature* de face F18.

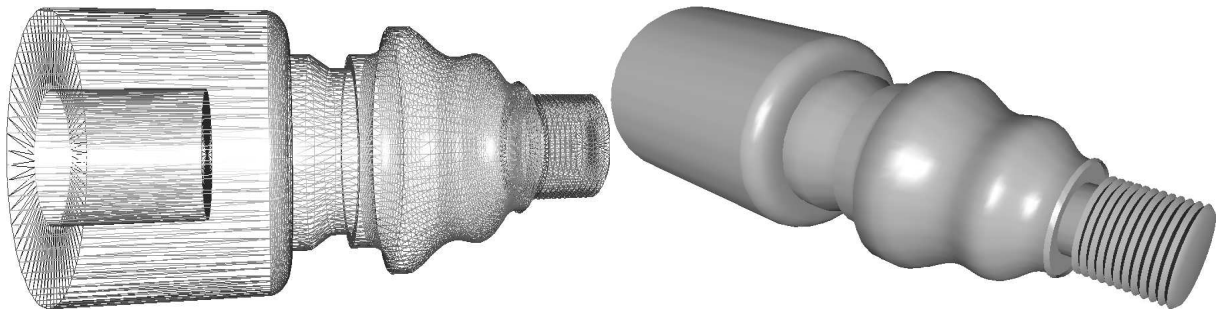


Figura 12.30: Modelagem VRML da peça complexa apresentando as *features* concêntricas em *wire-frame* e isométrica.

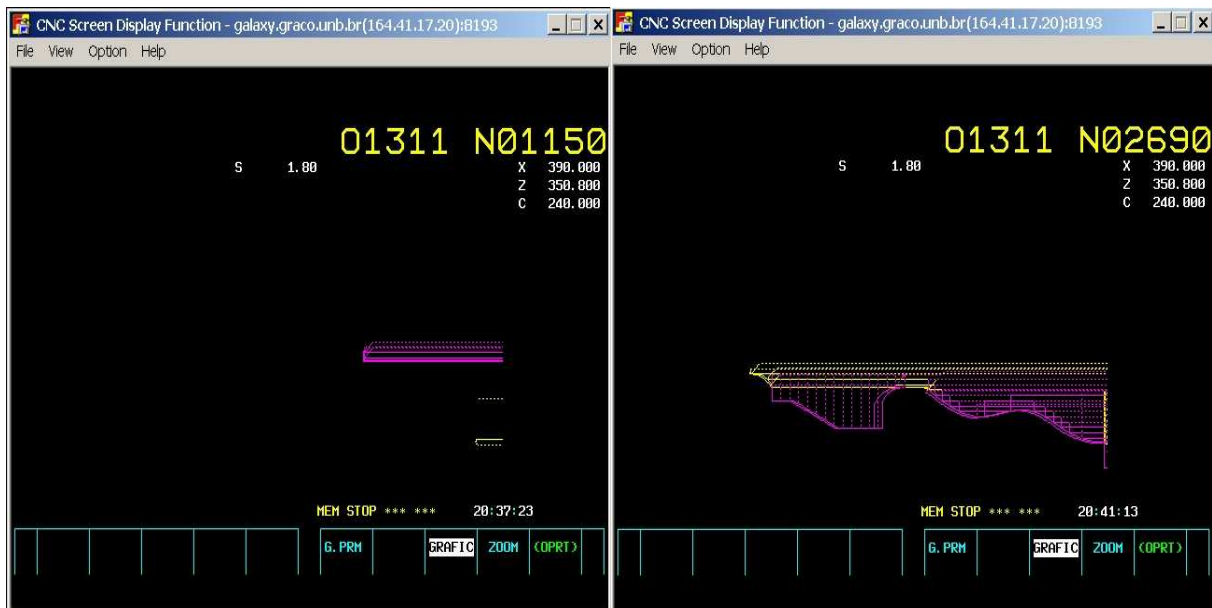


Figura 12.31: Simulação dos programas NC no CNC do Galaxy 15M para a peça complexa no *Setup1* e *Setup2*.

12.5.2 WebCAPP

O procedimento de mapeamento de *features* gera um plano de processo com dois *setups*, sendo que no *Setup1* ocorre a usinagem das *features* externas F17 a F18, das *features* internas F19 a F20 e das *features* de Eixo C, três rasgos radial e três furos axial. Cabe destacar que o desbaste associado à *feature* F19 é realizado com ferramenta não-acionada, pois o diâmetro é de 22 mm, sendo maior portanto que o diâmetro suportado pelas ferramentas acionadas. No *Setup2* ocorre a usinagem das *features* externas F1 a F16, incluindo, *groove* complexo, *splines* e rosca.

A figura 12.31 apresenta a simulação gráfica do programas NC gerado pelo WebCAPP para o *Setup1* e *Setup2*.

12.5.3 WebTurning

A figura 12.32 apresenta a GUI WebDNC que mostra uma tela com a monitoração do processo de usinagem da peça associada aos dois *setups*. A figura 12.33 apresenta detalhes das *features* de região

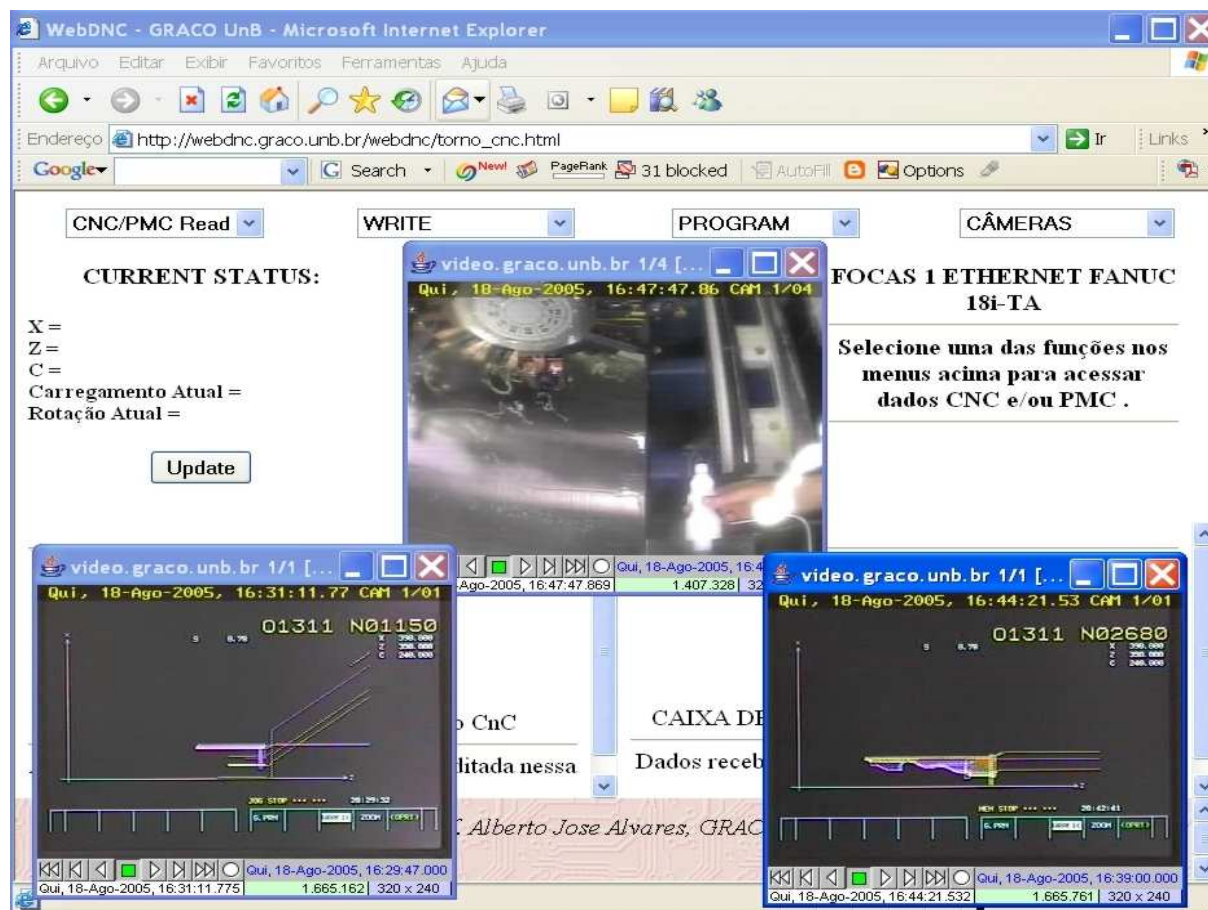


Figura 12.32: GUI WebDNC realizando a monitoração da usinagem da peça complexa.

externa, da peça complexa, usinada.

12.6 Peça Espelho

12.6.1 WebCADbyFeatures

O projeto da peça espelho hiperbólico foi aproximado por uma *feature spline* seguida por uma *feature cilíndrica*, sendo apresentado na figura 12.34. A peça possui apenas nove *features*, sendo utilizado um *blank* do tipo barra sólida (tarugo).

12.6.2 WebCAPP

O procedimento de mapeamento de *features* gera um plano de processo com um único *setup*, sendo que no *Setup1* ocorre a usinagem de todas as *features*. A figura 12.35 apresenta a simulação gráfica do programas NC gerado pelo WebCAPP para o *Setup1*.



Figura 12.33: *Features* externas e internas da peça complexa.

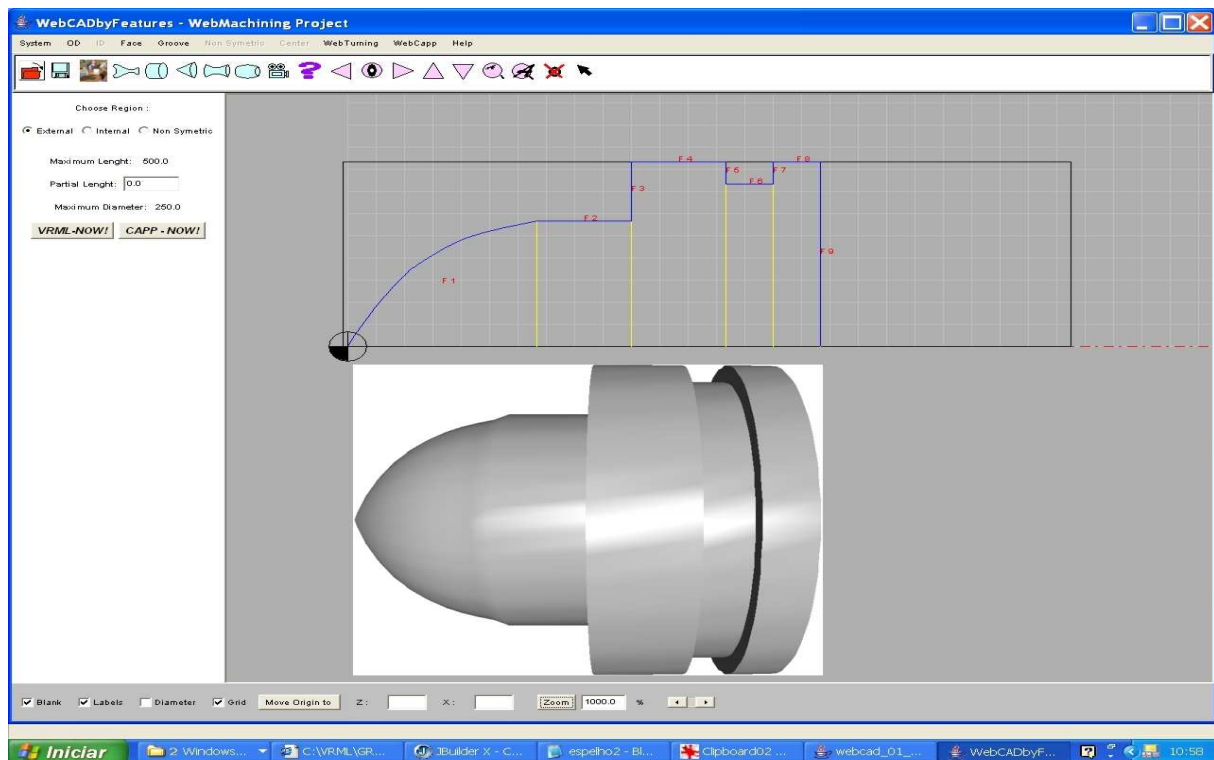


Figura 12.34: Modelagem da peça espelho hiperbólico aproximada por *spline* e o VRML associado.



Figura 12.35: Simulação do programa NC no CNC do Galaxy 15M para a peça espelho hiperbólico modelada por *spline*.

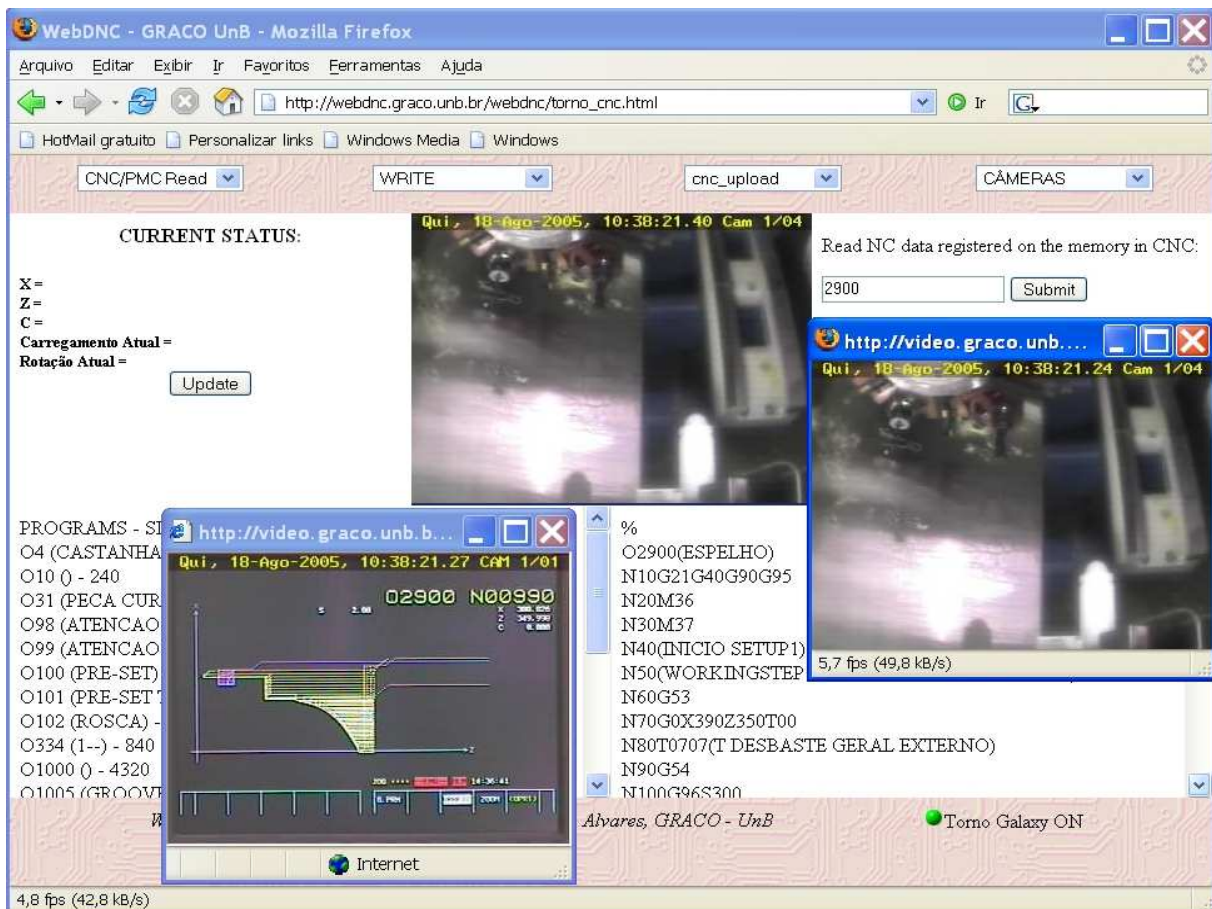


Figura 12.36: GUI WebDNC realizando a monitoração da usinagem da peça espelho.



Figura 12.37: Peça espelho usinada.

12.6.3 WebTurning

A figura 12.36 apresenta a GUI WebDNC que mostra uma tela com a monitoração do processo de usinagem da peça. As outras telas apresentam simulação, programas disponíveis na memória do CNC, fixação da peça no torno, entre outras informações pertinentes à interface de teleoperação. A figura 12.37 apresenta a peça usinada.

Capítulo 13

Conclusões

Este capítulo apresenta as contribuições do trabalho de doutorado e as conclusões associadas à metodologia WebMachining para integração CAD/CAPP/CAM via *Web* e à implementação computacional do sistema WebMachining. Também são apresentadas sugestões de trabalhos futuros, que visam principalmente a complementação da implementação do sistema computacional WebMachining.

13.1 Contribuições do Trabalho

A metodologia concebida utiliza a tecnologia Internet e de comunicação associada à SOA para oferecer um novo paradigma para o desenvolvimento dos futuros ambientes integrados CAD/CAPP/CAM. Estes ambientes serão globais, centrados em rede e espacialmente distribuídos, tendo como *front-end* com o sistema CAD/CAPP/CAM navegadores baseados na *Web*, permitindo também a independência da plataforma computacional do usuário.

Os Sistemas desenvolvidos a partir desta metodologia poderão ser aplicados na indústria e na academia. Esta metodologia de fabricação de peças permite que uma empresa (aqui representada pelo usuário remoto) não necessariamente possua os equipamentos para o projeto, planejamento e fabricação de um lote de peças, além da empresa fabricante das peças poder ter contato direto com o cliente.

A metodologia concebida e o sistema implementado apresentam muitas contribuições, inovações, para o desenvolvimento de sistemas de *e-mft* e telemanufatura baseados na *Web*, integrando atividades de projeto detalhado colaborativo, planejamento de processos não-linear e fabricação através da modelagem por *features*, utilizando uma arquitetura de sistemas multiagentes, protocolos TCP/IP, sistemas distribuídos e SOA.

O sistema é inédito no que concerne à abordagem de projeto colaborativo por síntese de *features* e no processo de fabricação utilizado, sendo voltado para fabricação de peças rotacionais simétricas (*features* concêntricas) e assimétricas (*features* não-concêntricas), incluindo *features* externas e internas, destacando-se as *splines* e as *features* de Eixo C. Inova também na incorporação de funcionalidades associadas à teleoperação via *Web* da máquina-ferramenta CNC permitindo a integração CAD/CAPP/CAM no chão-de-fábrica. O sistema CAPP concebido e implementado utiliza uma nova abordagem de mapeamento de *features* de projeto para *features* de torneamento e usinagem, denominada de “*decomposição orientada ao setup, geometria e operação*”, sendo utilizados conceitos

Tabela 13.1: Síntese das contribuições da tese.

Contribuição	Metodologia (cap. 6)	Algoritmo (caps. 7 e 8)	Implementação (caps. 9-11)
1	Integração CAD/CAPP/CAM		
2	Arquitetura Multiagente		Sistema WebMachining
3	Projeto colaborativo por <i>features</i>		
4	Uso de <i>features Spline</i> e Eixo C		
5			WebCADbyFeatures e Multiagente
6	CAPP generativo e não-linear		WebCAPP: <i>servlets</i> e <i>applets</i>
7			WebTurning: teleoperação
8	SOA: Cliente-servidor		SOA: Applets, Servlets, CGI e JATLite
9		Técnicas para telemanufatura	CGI, HTTP, JATLite, MySQL e Servlets
10		Métodos telemanufatura	e-Mfg & Restrições TCP/IP
11		CAPP: mapeamento de <i>features</i>	
12		GA condições usinagem	WebCAPP
13		GA planos não-lineares	WebCAPP

associados à norma STEP, sendo aderente ao STEP-NC, usando uma estrutura de dados semelhante e trabalhando com planos de processos não-lineares.

Outras contribuições específicas estão associadas à implementação do sistema utilizando a tecnologia Java *applet* e *servlets* que impõem uma maior segurança e restrições associadas à rede de comunicação. Esta implementação permite que o cliente utilize-se de qualquer plataforma computacional, de *Palms* à celulares, desde que o sistema operacional do dispositivo tenha suporte para Java, HTML e conexão Internet.

A seguir são destacadas as contribuições do trabalho relacionadas à integração CAD/CAPP/CAM e à Tecnologia da Informação, sendo as mesmas sintetizadas na tabela 13.1, como inovação metodológica, inovação associada à métodos e algoritmos concebidos e inovação associada à implementação computacional de módulos do sistema WebMachining:

1. A metodologia de integração CAD/CAPP/CAM, denominada de WebMachining, concebida para projeto detalhado colaborativo, planejamento de processo não-linear e fabricação remota de peças rotacionais via *Web* (SOA) é inédita.
2. Implementação computacional do sistema WebMachining (WebCADbyFeatures, WebCAPP e WebCAM), utilizando tecnologia *Web* e sendo disponibilizado para aplicações em EAD, como um laboratório remoto via Internet, cobrindo várias fases do ciclo de desenvolvimento de produto (projeto detalhado colaborativo, planejamento de processo não-linear e fabricação). A implementação computacional dos três módulos do sistemas WebMachining também é inédita.
3. CAD colaborativo baseado em *features* voltado para *Web*: é inovador na sua concepção, bem como, na aplicação voltada ao projeto colaborativo de peças rotacionais simétricas (*features* concêntricas) e assimétricas (*features* não-concêntricas), permitindo a visualização da peça modelada em 2D e 3D (VRML).

4. Tratamento de *features* concêntricas do tipo *splines* e não-concêntricas (Eixo C) pelo sistema integrado CAD/CAPP/CAM, estendendo o modelo CAM-I e STEP-NC para torneamento.
5. Modelagem colaborativa para desenvolvimento de produto via *Web* baseada em sistema multi-agente.
6. CAPP generativo, não-linear, aderente à STEP-NC e voltado para *Web* (*applet* e *servlet*): é inovador, principalmente na aplicação escolhida, voltada para planejamento do processo de peças rotacionais simétricas e assimétricas gerando planos de processos com alternativas (plano de processo não-linear). Não foi encontrada na literatura nenhum sistema CAPP com esta concepção. Vários métodos propostos e implementados para resolução das atividades relacionadas ao planejamento de processo são inovadores, como a “decomposição orientada ao *setup*, geometria e operação”, “geração de planos de processos não-lineares”, “otimização das condições de usinagem baseada em algoritmos genéticos” e “linearização do plano de processo baseado em algoritmos genéticos”.
7. Supervisão, monitoração e teleoperação do centro de usinagem Galaxy 15M da Romi (CNC Fanuc 18i-ta) via *Web*, podendo-se supervisionar, monitorar e comandar a máquina-ferramenta remotamente, alterando-se parâmetros do CNC e do PLC. O sistema de supervisão e teleoperação via *Web* está integrado aos servidores de *streaming* de vídeo e de áudio, que permitem uma imersão do usuário remoto no chão-de-fábrica.
8. Integração baseada em banco de dados relacional compartilhando toda a informação do sistema aliada à arquitetura multiagente e sistemas distribuídos, utilizando-se de *applets* e *servlets* em uma arquitetura cliente-servidor via *Web*.
9. Identificação e discussão de técnicas disponíveis para telemanufatura via *Web*. Isto inclui técnicas e protocolos tais como CGI, HTTP, HTML, Java, sistemas multiagentes, VRML, *servlets*, entre outros.
10. Análise detalhada de características da Internet para aplicação de telemanufatura, levando em conta o balanceamento da funcionalidade disponível na arquitetura cliente-servidor em relação à largura de banda, a fim de minimizar o *delay* inerente ao protocolo TCP/IP.
11. CAPP trabalha com planos de processos alternativos (não-linear), utilizando a estrutura de dados e nomenclatura definida pela norma STEP-NC.
12. Linearização do plano de processos (*Workplan*) baseada em algoritmo genético.
13. Modelagem e otimização das condições de usinagem baseada em algoritmo genético:
 - tratamento de operações de usinagem com alternativas, aumentando a amplitude das operações de usinagem a serem otimizadas, como mandrilamento, alargamento, furação, perfilamento, recartilhamento e corte;
 - a profundidade de corte para cada passe de usinagem nas operações de desbaste não são necessariamente iguais, podendo variar de um passe de desbaste a outro.

Tabela 13.2: Especificações, funcionalidades e requisitos do sistema WebMachining.

Características	WebCADbyFeatures	WebCAPP	WebTurning
SOA: Serviços Web	Projeto Colaborativo	CAPP não-linear	Teleoperação Galaxy 15M
Tecnologia de Features	Concêntricas e Eixo C	Workingsteps	
Tecnologia Web	Agentes, Applets e Servlets	Servlets, XML e Applets	Applets, Inetd, HTML e CGI
Plataforma	Linux, TomCat e Apache	Linux, Apache, TomCat e MySQL	Linux, VMWare e Apache
Linguagem	Java, KQML e VRML	Java e SQL	C, C++ e Java
Tempo Execução: Server	10s	4s	10s
Largura de Banda	32 kbits/s	32 kbits/s	76 kBytes/s (vídeo e áudio)
Conhecimento usuário	iniciante	nenhum	nenhum
Projeto com 40 features	modelagem: 5 minutos	automático: 15 s	ao vivo: 20 s
Padrões adotados	CAM-I e STEP	STEP-NC e ISO 6983	ISO 6983, Focas1, Ogg e jpg
Plug-in	VRML e Java	Java	Ogg e Java
Política segurança Java	escrita		escrita
Sockets	164.41.17.50:80	164.41.17.50:80, 8080, 4444	164.41.17.53: 80, 8000

13.2 Implementação Computacional WebMachining

O sistema WebMachining é inédito sendo acessado através da URL <http://WebMachining.AlvaresTech.com>, tendo vários dos seus módulos disponíveis para uso, oferecendo assim um laboratório remoto e um sistema de prototipagem rápida, em um contexto de *e-Mfg*, permitindo o projeto colaborativo (WebCADbyFeatures), planejamento de processo não-linear (WebCAPP) e fabricação a distância (WebTurning) via *Web*. A tabela 13.2 sintetiza as especificações, funcionalidades e requisitos de largura de banda para os três módulos do sistema WebMachining.

Em nenhuma das referências bibliográficas consultadas é apresentado a descrição de uma metodologia e sistema semelhante, bem como, de outra implementação computacional semelhante disponibilizada via *Web*.

13.2.1 WebCADbyFeatures

A implementação do sistema WebCADbyFeatures para desenvolvimento de produto colaborativo é baseado em modelagem por síntese de *features* de projeto para peças rotacionais (*features* concêntricas e não-concêntricas) via Internet, disponibilizado por um *browser* e *applet* Java. Este software, denominado WebCADbyFeatures, é um dos módulos do sistema e da metodologia WebMachining, que descreve um *framework* para integração CAD/CAPP/CAM.

Foi desenvolvido um sistema multiagente baseado na ferramenta JATLite, que viabiliza o projeto colaborativo, sendo implementado em uma arquitetura cliente-servidor, constituída por servidores, páginas HTML e Java *applets*, que permitem que o usuário remoto faça a modelagem colaborativa da peça em 2D e sua visualização em 2D e 3D, através de VRML. Dentre as características do sistema WebCADbyFeatures têm-se:

1. Permite o projeto detalhado colaborativo via *Web* utilizando um sistema multiagente como moderador;

2. Utiliza servidores multi-plataforma baseados em *servlets*, JATLite, HTTP, MySQL e FTP; implementados em linguagem Java, HTML, Javascript e PHP. Os servidores foram desenvolvidos em plataforma Linux, por ser mais estável e robusta, quando comparado à plataforma Windows, que também pode ser utilizada;
3. Cliente baseado em *applet* Java, usando AWT, possibilitando total compatibilidade com os *browsers*, bastando ativar a máquina virtual Java;
4. Não é necessário nenhum software complementar para modelagem de produto, apenas a instalação de um *plug-in* para visualização da peça em VRML;
5. Sistema multi-usuário e multi-tarefa, baseado em *threads*, tanto no lado do servidor quanto no lado do cliente;
6. Comunicação a distância entre pessoas, eliminando as barreiras geográficas e temporais para desenvolvimento de produto, permitindo a implementação de Engenharia Simultânea;
7. Modelagem usando *splines* para *features* do tipo *general_revolution* e possibilidade de introdução de *features* não-concêntricas (*features* de Eixo C), extrapolando STEP NC-Part 12 (ISO 14649, 2003);
8. Rapidez e segurança na comunicação entre os agentes; e
9. Integrado a um sistema de planejamento do processo generativo (WebCAPP) e fabricação remota (WebTurning) via *Web* através de *servlets* e *applets*;
10. Nesta versão foi implementado a modelagem VRML via *servlets* apenas para *features* concêntricas, não tendo-se implementado para nenhuma *feature* não-concêntrica.

13.2.2 WebCAPP

Os principais métodos propostos para as atividades associadas ao microplanejamento do WebCAPP foram implementados, em especial o novo método concebido para mapeamento de *features* de projeto em *features* de fabricação (usinagem e torneamento) baseado em *setup*, geometria e operação, levando em conta alternativas de usinagem.

Os métodos propostos para otimização de condições de usinagem e *Workplan* baseados em algoritmos genéticos foram desenvolvidos em MatLab, a fim de facilitar a implementação computacional, não se tendo tempo hábil para integrá-lo via *Web* ao sistema WebMachining. Estes métodos trabalham *off-line* e não estão totalmente integrados nesta versão do sistema WebMachining.

Algumas *features* de Eixo C (ferramentas rotativas) foram implementadas nesta versão do sistema WebCAPP, as demais serão incluídas nas próximas versões.

13.2.3 WebTurning

A GUI implementada para a teleoperação do centro de torneamento Galaxy 15M pode ser acessada através da URL <http://WebDNC.AlvaresTech.com>. Esta implementação está concluída parcialmente, tendo-se finalizado o servidor WebCam e o cliente para monitoração *on-line*, baseada em

vídeo e áudio, denominada por NetCam. Os servidores de teleoperação do CNC já implementados, permitem a execução de cerca de 70 funções, das 300 funções disponíveis. Escolheu-se as funções mais pertinentes para teleoperação do centro de torneamento.

O protocolo FOCAS1/ethernet só permite acesso direto ao CNC da máquina. Algumas funções de PLC, associadas ao *ladder*, são feitas indiretamente através de chamadas ao PLC via FOCAS1. Funções associadas ao teclado do CNC, como "cycle start", "MDI", "Edit", "JOG", entre outras, são acessadas via *ladder*, indiretamente pelas funções de escrita de PMC (CLP).

O sistema WebTurning apresenta como contribuição uma arquitetura cliente/servidor, baseada em tecnologia *Web* e multiplataforma, podendo ser acessado via *browser* sem a necessidade de nenhum software proprietário para teleoperação. Permite também a imersão do usuário remoto no chão-de-fábrica através da monitoração por vídeo e áudio em "tempo real" e por detecção de movimento, gravação de imagens e *playback* de eventos no chão-de-fábrica.

É necessário dotar o sistema de alguma inteligência para resolver conflitos que poderão ocorrer durante o processo de teleoperação. Para tornar o sistema de teleoperação mais seguro, utiliza-se uma arquitetura de controle do tipo supervisorio, onde as ações de controle são executadas localmente, a partir da solicitação de um comando remoto feita pelo cliente. Assim o sistema WebTurning é classificado como "controle supervisorio com predomínio do controle realizado pelo operador humano".

13.3 Sugestões para Trabalhos Futuros

A metodologia WebMachining para integração CAD/CAPP/CAM de peças rotacionais está concluída. Esta metodologia pode ser adaptada para o domínio de peças prismáticas, podendo aproveitá-la em grande parte, com exceção do WebCAPP que é mais específico, apesar de se trabalhar com ferramenta rotativa (furação, fresamento, entre outras).

Com relação à implementação computacional, foi gerada e disponibilizada uma primeira versão do sistema WebMachining, que tem muito a evoluir ainda, não apenas nas principais funcionalidades já implementadas associadas ao coração dos três módulos, mas principalmente com relação a várias funcionalidades complementares e de alguns refinamentos de interface, que não se teve condições de implementar devido às limitações temporais para conclusão do trabalho de doutorado.

A seguir é apresentada uma lista de trabalhos futuros que poderão complementar o sistema WebMachining e aperfeiçoá-lo também:

1. Acrescentar as *features* de Eixo C previstas e não implementadas;
2. Acrescentar um módulo de reconhecimento de *features* de projeto a partir de arquivos STEP ou IGES, gerado por outro sistema CAD comercial ou acadêmico, para a estrutura de *features* de forma de projeto utilizada pelo sistema WebMachining. Assim o sistema WebMachining incorporaria não só a técnica de projeto por *features*, mas a capacidade de trabalhar também com reconhecimento de *features*, a partir de um arquivo de CAD no formato STEP ou IGES, dando maior versatilidade ao sistema desenvolvido e a possibilidade de utilizar projetos legados de outros sistemas CAD, transformando o sistema WebCADbyFeatures em um sistema híbrido, baseado em síntese de *features* e reconhecimento de *features*;

3. Integrar os métodos de otimização de condições de usinagem e de linearização do *Workplan* baseados em algoritmos genéticos usando o MatLab via *Web* ao sistemas WebMachining;
4. Portar para Java (*servlet* e *applet*) os métodos de otimização de condições de usinagem e de linearização do *Workplan* baseados em algoritmos genéticos;
5. Finalizar a representação do plano de processo não-linear através de grafo E/OU via *applet* (WebCAPP);
6. Gerar mais alternativas de ferramental e operações de usinagem no planejamento de processo;
7. Melhorar a documentação do plano de processo e sua apresentação para o usuário;
8. Integrar o sistema WebMachining ao sistema de programação da produção associado à Célula Flexível de Manufatura (FMC) do GRACO, na qual o centro de torneamento Galaxy 15M está integrado (chão-de-fábrica);
9. Gerar código G para tornos de outros fabricantes, que estariam disponíveis para serem utilizados pelo sistema WebTurning em outras células ou sistemas de fabricação, como por exemplo pelo FMS da Sociesc de Joinville. Assim poderia ser levado em consideração na geração do plano de processo máquinas alternativas, disponíveis em sistemas de manufaturas remotos, podendo-se inclusive utilizar a abordagem da Matriz de Halevi, neste caso. Seria necessário também incluir um sistema de programação da produção, prevendo a integração dos vários sistemas de fabricação utilizados, no caso UnB, UFSC e Sociesc;
10. Realizar uma análise metrológica mais aprofundada das peças fabricadas, comparando o protocolo de medição das peças com as especificações funcionais associadas às tolerâncias dimensionais, forma, posição, orientação e rugosidade, quando presentes. Nos estudos de casos foram realizadas apenas medições de diâmetros e de comprimento das cotas críticas das peças fabricadas;
11. Implementar *servlet* VRML para *features* de Eixo C;
12. Implementação das demais funções FOCAS1 no servidor WebDNC, bem como o desenvolvimento de um cliente totalmente baseado em Java, emulando as telas do CNC, aumentando-se o grau de imersão do usuário remoto no chão-de-fábrica;
13. Estudar a possibilidade de aplicação da metodologia WebMachining associada às técnicas de prototipagem rápida voltada para a medicina reconstrutora, como por exemplo em traumas de faces ou próteses de membros;
14. Estudar a possibilidade de implementação de alguns módulos do sistema WebMachining baseado em RMI (*Remote Method Invocation*) e CORBA, permitindo a chamada entre objetos Java e objetos escritos em outras linguagens, como ocorre com o protocolo FOCAS1 (linguagem C usando SO Windows). Alguns testes foram feitos com RMI para monitoração de dados dinâmicos do centro de torneamento via FOCAS1, como velocidade, posição (X, Z e C), carregamento, entre outros, que se mostraram bastante promissores, e certamente serão implementados no futuro.

Apêndice A

Publicações e Projetos Associados à Tese

O desenvolvimento da tese propiciou, até o momento, a publicação de dois *papers* em revista internacional, quatro *papers* em revistas nacionais, dez *papers* em congressos internacionais (Portugal, Alemanha, Venezuela, USA e Brasil) e vinte *papers* em congressos nacionais. A lista de *papers* está disponível na URL http://Alvarestech.com/alvares_lattes.html.

Foi viabilizado um projeto de cooperação fomentado pelo CNPq no valor de R\$ 430.000,00 entre o programa de pós-graduação em Sistemas Mecatrônicos da Universidade de Brasília e os programas de pós-graduação em Engenharia Mecânica da Universidade Federal de Santa Catarina e da Universidade Federal de Uberlândia. A parceria com a UFSC, através do GRIMA, visa a implantação de uma FMC no Graco e a integração do sistema WebMachining nesta FMC. Este projeto viabilizou a compra de um micrômetro laser da Mitutoyo, de uma garra pneumática da Schunk, válvulas solenóides e acessórios pneumáticos da Festo, ferramentas de corte da Sandvik, 15 computadores, dois *notebooks*, entre outros, que estão sendo utilizados na implantação da FMC. Possibilitou o intercâmbio de professores e alunos entre a UnB e a UFSC através do pagamento de passagens e diárias.

Já a compra do Centro de Torneamento Galaxy 15M da Romi, conjunto de câmeras NTSC e três computadores, foram viabilizados a partir da contratação de um projeto FINEP pelo Graco, no valor de cerca de R\$ 350.000,00.

Foi viabilizado a doação de softwares por parte da FANUC, associados à comunicação via protocolos TCP/IP (rede local), destacando os softwares *CNCSCreen*, *BOP* e *Ladder*, bem como as APIs FOCAS1 e DNC1, que permitiu o desenvolvimento do sistema WebTurning.

Foi viabilizado a doação de portas-ferramenta e insertos por parte da Sandvik e da sua representante para a região centro-oeste, a empresa Kayman.

Foi obtido através da Fiat Automóveis a doação de um Robô Industrial ASEA IRB-6, com cinco graus de liberdade, que foi integrado à FMC.

Como resultado do desenvolvimento do sistema WebCam, que realiza a monitoração do chão-de-fábrica através da captura de vídeo e áudio, obteve-se a aprovação de um projeto de desenvolvimento de produto pelo Edital MCT/FINEP/CT-INFO-SOFTWARE-01/2005 no valor de R\$ 358.000,00, para o desenvolvimento de uma família de produtos baseados no módulo WebCam (NetCam) do sistema WebMachining, denominado de: “NetCam - Desenvolvimento de Hardware e Software Embarcado em Plataforma Linux Para Monitoração, Processamento, Gravação e Transmissão de Vídeo Digital”.

Associado à tese estão sendo orientados dois trabalhos de dissertação de mestrado na UnB visando

a implementação de uma FMC rotacional (constituída pelo centro de torneamento Galaxy 15M, robô ASEA IRB-6, micrômetro laser, robô móvel Nomad XR4000 (AGV), garra pneumática Schunk, paletes de peças brutas e acabadas e unidade de gerenciamento), e a integração do sistema Web-Machining à unidade de gerenciamento da FMC, e ao sistema de programação da produção (<http://WebFMC.AlvaresTech.com>).

Todos estes resultados foram obtidos a partir do desenvolvimento desta tese de doutoramento, que permitiu ganhos tangíveis associados à infra-estrutura dos laboratórios do Grupo de Automação e Controle (GRACO) e do Grupo de Integração da Manufatura (GRIMA), bem como ganhos intangíveis associados à estruturação e desenvolvimento do programa de pós-graduação em Sistemas Mecatrônicos da UnB e formação de recursos humanos, em especial, na área de pesquisa de Automação da Manufatura.

Apêndice B

URLs Associadas à Metodologia WebMachining

B.1 Metodologia Para Teleoperação Via Web

A metodologia para teleoperação via *Web* está disponível em ÁLVARES (2003) e na URL <http://WebMachining.AlvaresTech.com/qualificacao3.pdf>.

B.2 Metodologias IDEF

São utilizadas as metodologias IDEF para modelagem funcional (IDEF0) e modelagem de informações (IDEF1X) do sistema WebMachining (HARRINGTON, 1984). Os arquivos (formatos fontes das ferramentas, pdf, ps e HPGL) contendo os modelos estão disponíveis em <http://WebMachining.AlvaresTech.com/idef>.

B.3 Modelagem Física e Conceitual do Banco de Dados MySQL

O modelo IDEF1X foi convertido em um modelo conceitual Entidade-Relacionamento e a seguir gerado o modelo físico da Base de Dados em formato MySQL (<http://www.mysql.com>) utilizando a ferramenta computacional Sybase Power Designer 9.0 da empresa Sybase (<http://www.sybase.com>). Os arquivos contendo o modelo estão disponíveis em <http://WebMachining.AlvaresTech.com/modeloer>.

B.3.1 Modelagem Conceitual

O modelo conceitual Entidade-Relacionamento da Base de Dados WebMachining a partir do modelo IDEF1X elaborado¹, está disponível em <http://WebMachining.AlvaresTech.com/modeloer/conceitual>.

¹Não foi possível gerar o modelo físico da base de dados WebMachining para MySQL utilizando a ferramenta VisualModeler pois a mesma não é compatível com o MySQL. Assim foi necessário utilizar uma outra ferramenta computacional que permitisse a geração da base de dados física, o que foi feito com o *software* Sybase Power Designer. Este *software* não trabalha com a metodologia IDEF1X.

B.3.2 Modelagem Física

O modelo físico da base de dados relacional em formato MySQL foi gerado automaticamente a partir do modelo conceitual entidade-relacionamento. Os arquivos contendo o modelo estão disponíveis em <http://WebMachining.AlvaresTech.com/modeloer/fisico>.

B.4 Biblioteca de *Features*

É utilizada a biblioteca de *features* de forma para planejamento de processo definida por CAM-I (1986). As definições das *features* de forma especificada na proposta CAM-I, bem como, os dados necessários para planejamento de processo para estas *features*, que foram modelados pela abordagem IDEF1X, estão disponíveis em ÁLVARES (2003) e na URL <ftp://WebMachining.AlvaresTech.com/cami>.

Apêndice C

Modelos Matemáticos Para Otimização

Neste apêndice são apresentados os modelos matemáticos associados à otimização das condições de usinagem.

C.1 Modelo Matemático Passe Único para Cálculo de Tempos de Usinagem: Faceamento¹ e Torneamento² Cilíndrico, Cônico e Circular

Considerando que a taxa de avanço, definida como velocidade de avanço V_f (mm/min), é igual ao produto do avanço f (mm/rev) pela velocidade rotacional N (rev/min) da peça; e a velocidade rotacional N é igual a velocidade de corte V_c (m/min) dividido pelo comprimento da circunferência associado ao diâmetro D (mm) da peça, logo:

$$V_f = fN \quad (\text{C.1})$$

$$N = \frac{1000V_c}{\pi D} \quad (\text{C.2})$$

Para operações de torneamento cilíndrico, cônico e circular, bem como para faceamento (torneamento transversal) ocorre a variação do diâmetro da peça, de D_1 a D_2 . Quando especifica-se V_c como uma velocidade de corte constante, há a necessidade de variar N entre D_1 e D_2 , sendo necessário redefinir N como:

$$N = \frac{1000V_c}{2\pi} \left(\frac{1}{D_1} + \frac{1}{D_2} \right) \quad (\text{C.3})$$

Para interpolação linear (fig. 7.30 (c)) entre dois pontos $P_1(Z_1, X_1)$ e $P_2(Z_2, X_2)$ o diâmetro D de um ponto $P(z,x)$ sobre a linha reta que passa por P_1 e P_2 é

$$D = 2x \quad (\text{C.4})$$

O tempo infinitesimal dt , para um deslocamento infinitesimal dl de uma ferramenta com uma

¹Torneamento transversal retilíneo

²Torneamento longitudinal

velocidade de avanço V_f no diâmetro D , é dado pela seguinte forma diferencial:

$$dt = \frac{dl}{V_f} \quad (C.5)$$

Incorporando as equações (C.1)-(C.4) na equação (C.5) tem-se:

$$dt = \frac{2\pi x dx}{1000V_c f \text{sen}\theta} \quad (C.6)$$

onde θ é $\arctan [(X_2 - X_1)/(Z_2 - Z_1)]$ no intervalo aberto $(0, \pi)$ e $(\pi, 2\pi)$. O tempo de movimento da ferramenta para o deslocamento entre P_1 e P_2 , **torneamento cônico**, é

$$\begin{aligned} t &= \int_{x_1}^{x_2} \frac{2\pi x dx}{1000V_c f \text{sen}\theta} \\ &= \frac{\pi}{1000V_c f} \left| \frac{X_2^2 - X_1^2}{\text{sen}\theta} \right| \end{aligned} \quad (C.7)$$

Se θ não for igual a $0, \pi/2, \pi$, ou $3\pi/2$ o torneamento linear pode ser classificado como um torneamento cônico.

No caso de **torneamento cilíndrico**, onde θ na equação (C.7) é 0 ou π , a equação torna-se

$$t = \frac{\pi |x(Z_2 - Z_1)|}{500V_c f} \quad (C.8)$$

onde $x=x_1=x_2$; e $|Z_2 - Z_1|$ é o comprimento do torneamento longitudinal retilíneo.

No caso de **faceamento**, onde θ na equação (C.7) é $\pi/2$ ou $3\pi/2$ ($\text{sen}\theta = \pm 1$), o tempo de corte será

$$t = \frac{\pi}{1000V_c f} |X_2^2 - X_1^2| \quad (C.9)$$

A análise para interpolação circular (figura 7.30 (b)) é análoga ao caso linear, sendo que o tempo infinitesimal é dado por

$$dt = \frac{R d\theta}{V_f} \quad (C.10)$$

Usando o centro do arco $P_c(Z_c, X_c)$, a coordenada X do ponto $P(z,x)$ é igual a

$$x = X_c + R \text{sen}\theta \quad (C.11)$$

O tempo de movimento da ferramenta para **torneamento circular**, tempo de corte, entre $P_1(Z_1, X_1)$ e $P_2(Z_2, X_2)$ é

$$\begin{aligned} t &= \int_{\theta_1}^{\theta_2} \frac{2\pi R}{1000V_c f} (X_c + R \text{sen}\theta) d\theta \\ &= \frac{2\pi R}{1000V_c f} |X_c(\theta_2 - \theta_1) - R(\cos\theta_2 - \cos\theta_1)| \end{aligned} \quad (C.12)$$

onde θ_1 é $\arctang((X_1 - X_c)/(Z_1 - Z_c))$ e θ_2 é $\arctang((X_2 - X_c)/(Z_2 - Z_c))$, no intervalo fechado $[0, 2\pi]$.

C.2 Modelo Matemático Multi-passe para Tempo de Corte de Perfil Contínuo

Neste caso utiliza-se a abordagem multi-corte, onde o desbaste é dividido em dois estágios com n passes de usinagem e o acabamento em um único passe, conforme descrito anteriormente (figura 7.30). No caso mais genérico as seguintes operações de usinagem são necessárias para remoção das *features* de usinagem:

- **Primeiro estágio de desbaste:** $(n-1)$ passes associados às operações de torneamento retilíneo longitudinal ou torneamento retilíneo transversal (sangramento), associados aos ciclos fixos G71 ou G72 respectivamente, deixando um sobremetal para o segundo estágio de desbaste, que executará o passe n . A trajetória de corte é linear onde a profundidade de corte para cada passe pode ser calculada para cada linha L_g ³ por $x - (x_I - \sum_{g=1}^{n-1} dr_g) = 0$, onde $P_I(Z_I, X_I)$ é o ponto inicial (figura 7.30), dr_g é a profundidade de corte para cada passe, onde $g=1, 2, \dots, n-1$. Os pontos de intersecção $P_{N(g)}(Z_{N(g)}, X_{N(g)})$ de cada linha L_g com o perfil da peça mais o sobremetal de acabamento ds é calculado para cinco casos:

1. segmento cônico entre $P_{1t}(Z_{1t}, X_{1t})$ e $P_{2t}(Z_{2t}, X_{2t})$ (Eq. (C.13)):

$$X_{N(g)} = X_I - \sum_{i=1}^g dr_g ; Z_{N(g)} = Z_{1t} + \frac{Z_{2t} - Z_{1t}}{X_{2t} - X_{1t}} \left[(d_t - \sum_{i=1}^g dr_g) - X_{1t} \right] \quad (C.13)$$

2. face de um segmento entre $P_{1f}(Z_{1f}, X_{1f})$ e $P_{2f}(Z_{2f}, X_{2f})$ (Eq. (C.14)):

$$X_{N(g)} = X_I - \sum_{i=1}^g dr_g ; Z_{N(g)} = Z_{1v} \text{ ou } Z_{N(g)} = Z_{2v} (Z_{1v} = Z_{2v}) \quad (C.14)$$

3. arco de circunferência convexo entre $P_{1c}(Z_{1c}, X_{1c})$ e $P_{2c}(Z_{2c}, X_{2c})$, sendo $P_c(Z_c, X_c)$ é o centro e r_a é o raio arco de circunferência (Eq. (C.15)):

$$X_{N(g)} = X_I - \sum_{i=1}^g dr_g ; Z_{N(g)} = Z_c + \sqrt{r_a^2 - \left[(d_t - \sum_{i=1}^g dr_g) - X_c^2 \right]^2} \quad (C.15)$$

4. arco de circunferência côncavo entre $P_{1c}(Z_{1c}, X_{1c})$ e $P_{2c}(Z_{2c}, X_{2c})$, sendo $P_c(Z_c, X_c)$ é o centro e r_a é o raio arco de circunferência (Eq. (C.16)):

$$X_{N(g)} = X_I - \sum_{i=1}^g dr_g ; Z_{N(g)} = Z_c - \sqrt{r_a^2 - \left[(d_t - \sum_{i=1}^g dr_g) - X_c^2 \right]^2} \quad (C.16)$$

5. entre dois pontos de intersecção do próprio perfil da peça, ou seja ponto crítico, onde uma

³ L_g é a linha que descreve a trajetória longitudinal de usinagem para cada passe de desbaste no primeiro estágio.

feature de projeto muda para a outra (Eq. (C.17)):

$$X_{N(g)} = X_{\text{ponto}_{\text{critico}}} ; Z_{N(g)} = Z_{\text{ponto}_{\text{critico}}} \quad (\text{C.17})$$

A seguir calcula-se o comprimento de corte para cada passe relativo ao torneamento longitudinal, $H_{(g)}$, no primeiro estágio de desbaste, onde Z_I é a coordenada Z do ponto inicial $P_I(Z_I, X_I)$ e d_s a profundidade de corte de acabamento:

$$H_{(g)} = |Z_{N(g)} - Z_I| - d_s \quad (\text{C.18})$$

Logo o tempo de corte para cada passe pode ser calculado através das equações (C.8) e (C.41), resultando em

$$T_{hr(g)} = \frac{\pi |X_{(g)} H_{(g)}|}{500 V_{cr} f_r} \quad (\text{C.19})$$

Sendo $H_{(g)} = |Z_{N(g)} - Z_I| - d_s$; $X_{(g)} = X_{N(g)} = X_I - \sum_{i=1}^g dr_g$, V_{cr} e f_r a velocidade de corte e avanço para desbaste. O tempo de corte total para o primeiro estágio de desbaste S_{r1} será

$$S_{r1} = \sum_{g=1}^{n-1} T_{hr(g)} \quad (\text{C.20})$$

- **Segundo estágio de desbaste (semi-acabamento):** passe n (último passe de desbaste) onde será executado o perfilamento/cópia de desbaste constituído por uma seqüência contínua de operações de torneamento retilíneo, torneamento cônico, torneamento circular ou faceamento, deixando um sobremetal d_s para o passe de acabamento, associados aos ciclos fixos G71 ou G72. Assim o perfil 2D interno e/ou externo da peça (poliforma) é constituído por um conjunto de arcos de circunferência e de segmentos de retas paralelas, perpendiculares e oblíquas ao eixo rotacional da peça. Deve-se calcular o tempo de corte para cada conjunto de torneamento longitudinal, segmento retilíneo, tendo $P_{1(i)}(Z_{1(i)}, X_{1(i)})$ e $P_{2(i)}(Z_{2(i)}, X_{2(i)})$, os pontos inicial e final no perfil, logo para o enésimo passe de desbaste do torneamento de segmento retilíneo i , tem-se:

$$T_{hr(i)(n)} = \frac{\pi |X_{(i)(n)} (Z_{2(i)} - Z_{1(i)})|}{500 V_{cr} f_r}$$

$$X_{(i)(n)} = \frac{D_{(i)}}{2} + d_s$$

$$T_{hr(i)(n)} = \frac{\pi \left| \left(\frac{D_{(i)}}{2} + d_s \right) (Z_{2(i)} - Z_{1(i)}) \right|}{500 V_{cr} f_r} \quad (\text{C.21})$$

O tempo total de corte para o passe de desbaste de segmentos retilíneos $S_{hr(n)}$ onde n_h é o número de segmentos retilíneos, será

$$S_{hr(n)} = \sum_{i=1}^{n_h} T_{hr(i)(n)} \quad (\text{C.22})$$

Para torneamento cônico do cone j entre dois pontos $P_{1(j)}(Z_{1(j)}, X_{1(j)})$ e $P_{2(j)}(Z_{2(j)}, X_{2(j)})$, as

coordenadas de $P_{1(j)(n)}$ e $P_{2(j)(n)}$ e os diâmetros respectivos da peça acabada $D_{1(j)}$ e $D_{2(j)}$, para o enésimo passe de desbaste é

$$x_{1(j)(n)} = \frac{D_{1(j)}}{2} + ds ; x_{2(j)(n)} = \frac{D_{2(j)}}{2} + ds \quad (\text{C.23})$$

$$T_{tr(j)(n)} = \frac{\pi}{1000V_{cr}f_r} \left| \frac{\frac{1}{4} (D_{2(j)}^2 - D_{1(j)}^2) + (D_{2(j)} - D_{1(j)})ds}{\text{sen}\theta_{(j)}} \right| \quad (\text{C.24})$$

$$\text{onde } \theta_{(j)} = \arctan \left(\frac{X_{2(j)} - X_{1(j)}}{Z_{2(j)} - Z_{1(j)}} \right) = \arctan \left(\frac{D_{2(j)} - D_{1(j)}}{2(Z_{2(j)} - Z_{1(j)})} \right) ; 0 < \theta_{(j)} < \pi ; \pi < \theta_{(j)} < 2\pi$$

O tempo total de corte para o passe de desbaste de segmentos cônicos $S_{tr(n)}$ onde n_t é o número de segmentos cônicos, será

$$S_{tr(n)} = \sum_{j=1}^{n_t} T_{tr(j)(n)} \quad (\text{C.25})$$

O tempo de corte para o enésimo passe de desbaste para todos os segmentos de faceamento $S_{vr(n)}$ pode ser obtido da equação (C.24) fazendo $\text{sen}\theta = \pm 1$ e n_v é o número de segmentos de faceamento:

$$S_{vr(n)} = \sum_{k=1}^{n_v} \frac{\pi}{1000V_{cr}f_r} \left| \frac{1}{4} (D_{2(k)}^2 - D_{1(k)}^2) + (D_{2(k)} - D_{1(k)})ds \right| \quad (\text{C.26})$$

Para o arco de circunferência l entre dois pontos $P_{1(l)}(Z_{1(l)}, X_{1(l)})$ e $P_{2(l)}(Z_{2(l)}, X_{2(l)})$, raio do arco $r_{a(l)(n)}$ do enésimo passe de desbaste será

$$r_{a(l)(n)} = R_{(l)} + ds \quad (\text{C.27})$$

onde $R_{(l)}$ é o raio da peça acabada para o arco de circunferência l . Trocando R na equação (C.12) por $r_{a(l)(n)}$ (Eq. (C.27)), o tempo de corte do enésimo passe de desbaste para o arco de circunferência l , dado por $T_{cr(l)(n)}$ será

$$T_{cr(l)(n)} = \frac{\pi(R_{(l)} + ds)}{500V_{cr}f_r} \left| X_{c(l)}(\theta_{2(l)} - \theta_{1(l)}) - (R_{(l)} + ds)(\cos\theta_{2(l)} - \cos\theta_{1(l)}) \right| \quad (\text{C.28})$$

onde

$$\theta_{1(l)} = \arctan \left(\frac{X_{1(l)} - X_{c(l)}}{Z_{1(l)} - Z_{c(l)}} \right) = \arctan \left(\frac{\frac{D_{1(l)}}{2} - X_{c(l)}}{Z_{1(l)} - Z_{c(l)}} \right)$$

$$\theta_{2(l)} = \arctan \left(\frac{X_{2(l)} - X_{c(l)}}{Z_{2(l)} - Z_{c(l)}} \right) = \arctan \left(\frac{\frac{D_{2(l)}}{2} - X_{c(l)}}{Z_{2(l)} - Z_{c(l)}} \right)$$

$D_{1(l)}$ e $D_{2(l)}$ são os diâmetros da peça acabada para o arco de circunferência l nos pontos $P_{1(l)}$ e $P_{2(l)}$, respectivamente; e $P_{c(l)}(Z_{c(l)}, X_{c(l)})$ é o centro do arco de circunferência l . O tempo de corte para o segundo estágio de desbaste, último passe de desbaste, para todos os arcos de circunferência será

$$S_{cr(n)} = \sum_{l=1}^{n_c} T_{cr(l)(n)} \quad (\text{C.29})$$

onde nc é o número de arcos circulares na peça usinada.

O tempo total de corte para o segundo estágio de desbaste S_{r2} será dado pela soma de cada parcela das operações de desbaste de perfilamento

$$S_{r2} = S_{hr(n)} + S_{tr(n)} + S_{vr(n)} + S_{cr(n)} \quad (C.30)$$

- **Acabamento:** passe de acabamento com profundidade de corte “ ds ”, onde será executado o perfilamento/cópia de acabamento constituído por uma seqüência continua de operações de torneamento cilíndrico, torneamento cônico, torneamento circular ou faceamento, associado ao ciclo fixo G70. O cálculo é semelhante ao segundo estágio das operações de desbaste, sendo dividido em quatro tipos de operações já tratadas. O tempo de corte para o passe de acabamento para cada conjunto de torneamento de segmento retilíneo, onde $P_{1(i)}(Z_{1(i)}, X_{1(i)})$ e $P_{2(i)}(Z_{2(i)}, X_{2(i)})$, são os pontos inicial e final no perfil.

Assim o passe de acabamento para **torneamento logitudinal**, segmento retilíneo, será

$$T_{hs(i)} = \frac{\pi |X_{s(i)}(Z_{2(i)} - Z_{1(i)})|}{500V_s f_s}$$

$$X_{s(i)} = \frac{D^{(i)}}{2}$$

$$T_{hs(i)(n)} = \frac{\pi \left| \left(\frac{D^{(i)}}{2} + d_s \right) (Z_{2(i)} - Z_{1(i)}) \right|}{500V_{cs} f_s} \quad (C.31)$$

O tempo total de corte para o passe de acabamento de segmentos retilíneos S_{hs} onde n_h é o número de segmentos retilíneos, será

$$S_{hs} = \sum_{i=1}^{n_h} T_{hs(i)} \quad (C.32)$$

Para **torneamento cônico do cone j** entre dois pontos $P_{1(j)}(Z_{1(j)}, X_{1(j)})$ e $P_{2(j)}(Z_{2(j)}, X_{2(j)})$, as coordenadas de $P_{1(j)(n)}$ e $P_{2(j)(n)}$ e os diâmetros respectivos da peça acabada $D_{1(j)}$ e $D_{2(j)}$, para o passe de acabamento é

$$x_{1(j)} = \frac{D_{1(j)}}{2}; \quad x_{2(j)} = \frac{D_{2(j)}}{2} \quad (C.33)$$

$$T_{ts(j)} = \frac{\pi}{1000V_{cs} f_s} \frac{\left| \left(D_{2(j)}^2 - D_{1(j)}^2 \right) \right|}{4sen\theta_{(j)}} \quad (C.34)$$

onde $\theta_{(j)} = \arctan \left(\frac{X_{2(j)} - X_{1(j)}}{Z_{2(j)} - Z_{1(j)}} \right) = \arctan \left(\frac{D_{2(j)} - D_{1(j)}}{2(Z_{2(j)} - Z_{1(j)})} \right)$; $0 < \theta_{(j)} < \pi$; $\pi < \theta_{(j)} < 2\pi$.

O tempo total de corte para o passe de acabamento de segmentos cônicos $S_{ts(n)}$ onde n_t é o número de segmentos cônicos, será

$$S_{ts} = \sum_{j=1}^{n_t} T_{ts(j)} \quad (C.35)$$

O tempo de corte para o passe de acabamento para todos os segmentos de **faceamento** S_{vs} pode ser obtido da equação (C.34) fazendo $\text{sen}\theta = \pm 1$ e n_v é o número de segmentos de faceamento:

$$S_{vs} = \sum_{k=1}^{n_v} \frac{\pi}{1000V_{cs}f_s} \left| \frac{1}{4} (D_{2(k)}^2 - D_{1(k)}^2) \right| \quad (\text{C.36})$$

Para o arco de circunferência l entre dois pontos $P_{1(l)}(Z_{1(l)}, X_{1(l)})$ e $P_{2(l)}(Z_{2(l)}, X_{2(l)})$, raio do arco $r_{a(l)}$ do passe de acabamento será

$$r_{a(l)} = R_{(l)} \quad (\text{C.37})$$

onde $R_{(l)}$ é o raio da peça acabada para o arco de circunferência l . Trocando $r_{a(l)}$ na equação (C.12) por $R_{(l)}$ (Eq. (C.37)), o tempo de corte para o passe de acabamento para o arco de circunferência l , dado por $T_{cs(l)}$ será

$$T_{cs(l)} = \frac{\pi R_{(l)}}{500V_{cs}f_s} \left| X_{c(l)}(\theta_{2(l)} - \theta_{1(l)}) - R_{(l)}(\cos\theta_{2(l)} - \cos\theta_{1(l)}) \right| \quad (\text{C.38})$$

onde

$$\theta_{1(l)} = \arctan \left(\frac{X_{1(l)} - X_{c(l)}}{Z_{1(l)} - Z_{c(l)}} \right) = \arctan \left(\frac{\frac{D_{1(l)}}{2} - X_{c(l)}}{Z_{1(l)} - Z_{c(l)}} \right)$$

$$\theta_{2(l)} = \arctan \left(\frac{X_{2(l)} - X_{c(l)}}{Z_{2(l)} - Z_{c(l)}} \right) = \arctan \left(\frac{\frac{D_{2(l)}}{2} - X_{c(l)}}{Z_{2(l)} - Z_{c(l)}} \right)$$

$D_{1(j)}$ e $D_{2(j)}$ são os diâmetros da peça acabada para o arco de circunferência l nos pontos $P_{1(l)}$ e $P_{2(l)}$, respectivamente; e $P_{c(l)}(Z_{c(l)}, X_{c(l)})$ é o centro do arco de circunferência l . O tempo de corte para o passe de acabamento para todos os arcos de circunferência será

$$S_{cs} = \sum_{l=1}^{n_c} T_{cs(l)} \quad (\text{C.39})$$

onde n_c é o número de arcos circulares na peça usinada.

O tempo total de corte para o passe de acabamento S_f será dado pela soma de cada parcela das operações de acabamento de perfilamento

$$S_f = S_{hs} + S_{ts} + S_{vs} + S_{cs} \quad (\text{C.40})$$

C.3 Modelo Matemático Multi-passe para Tempo de Corte de Torneamento Longitudinal

O modelo multi-passe utilizado para cálculo de tempos é semelhante ao modelo adotado para torneamento retilíneo de desbaste no primeiro estágio, sendo constituído por uma operação multi-passe de desbaste, formada por n passes de desbaste, e por um único passe de acabamento. O modelo de tempos para as operações de desbaste e acabamento é apresentado a seguir.

- **Desbaste:** calcule o comprimento de corte para cada passe relativo ao torneamento longitudinal,

$H_{(g)}$ onde Z_1 é a coordenada Z do ponto inicial $P_I(Z_1, X_1)$; $P_{N(g)}(Z_{N(g)}, X_{N(g)})$ é o ponto crítico de intersecção de cada linha associada ao passe de usinagem; e d_s a profundidade de corte de acabamento:

$$H_{(g)} = |Z_{N(g)} - Z_I| - d_s \quad (C.41)$$

Logo o tempo de corte para cada passe pode ser calculado através das equações (C.8) e (C.41), resultando em

$$T_{hr(g)(m)} = \sum_{m=1}^{n_r} \left(\frac{\pi |X_{(g)(m)} H_{(g)(m)}|}{500 V_{cr} f_r} \right)$$

Sendo $H_{(g)} = |Z_{N(g)} - Z_I| - d_s$; $X_{(g)} = X_{N(g)} = X_I - \sum_{g=1}^n dr_g$, V_{cr} e f_r a velocidade de corte e avanço para desbaste longitudinal, respectivamente, e n_r o número de *features* presentes na peça e que serão usinadas por este procedimento. O tempo de corte total para os n passes de desbaste longitudinal S_{rr} será

$$S_{rr} = \sum_{g=1}^n T_{hr(g)(m)}$$

- **Acabamento:** calcula-se o único passe de acabamento após a conclusão do enésimo passe de desbaste, onde o tempo de usinagem será dado por

$$T_{hs} = \frac{\pi |D_{(i)}(Z_{2(i)} - Z_{1(i)})|}{500 V_s f_s}$$

Onde $D(i)$ é diâmetro externo da *feature* OD. O tempo total de corte para o passe de torneamento longitudinal é S_{hs} , sendo n_r o número de segmentos retilíneos (*features*) presentes na peça:

$$S_{hs} = \sum_{m=1}^{n_r} T_{hs(m)} \quad (C.42)$$

C.4 Modelo Matemático Multi-passe para Tempo de Corte de Sangramento, Corte de Peças e Faceamento

O modelo multi-passe utilizado para cálculo de tempos é semelhante ao modelo adotado para faceamento, no segundo estágio de desbaste. Em função da geometria da peça será necessário diversos posicionamentos da ferramenta de sangramento ao longo da direção longitudinal da peça (eixo Z), a fim de concluir a usinagem da *feature* de usinagem, ou de um único posicionamento para corte da peça.

A figura C.1 apresenta a trajetória de uma ferramenta de sangramento para gerar a superfície desejada. A trajetória da ferramenta durante a usinagem é semelhante ao faceamento. Logo o cálculo de tempo de usinagem é análogo ao caso do cálculo de faceamento para segundo estágio de ferramenta de desbaste. Pode-se ter durante a usinagem um passe de desbaste e um passe final de acabamento para efetuar o sangramento da peça com velocidade de corte e avanço diferentes para cada passe, usando a mesma ferramenta. Já para corte tem-se apenas um único passe de desbaste. Pode-se utilizar o ciclo fixo G75.

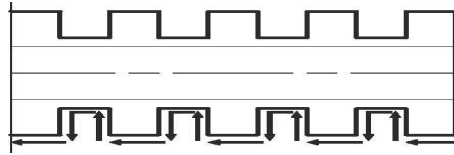


Figura C.1: Modelo de trajetória de ferramenta para sangramento (SANDVIK, 2003).

Assim o modelo de sangramento será constituído por dois passes (desbaste e acabamento), sendo igual ao modelo para faceamento. Já o modelo de corte de peças é constituído apenas pelo modelo de desbaste, associado ao corte da peça. O modelo de tempo para os dois passes é apresentado a seguir:

- **Desbaste:** O tempo de corte para o passe de desbaste para todos os sangramentos S_{gr} pode ser obtido da equação (C.24) fazendo $sen\theta = \pm 1$ e n_{gr} é o número de operações de sangramento:

$$S_{gr} = \sum_{k=1}^{n_{gr}} \frac{\pi}{1000V_{cr}f_r} \left| \frac{1}{4} (D_{2(k)}^2 - D_{1(k)}^2) + (D_{2(k)} - D_{1(k)})d_s \right| \quad (C.43)$$

- **Acabamento:** O tempo de corte para o passe de acabamento para todos os sangramentos S_{gs} pode ser obtido da equação (C.34) fazendo $sen\theta = \pm 1$ e n_{gs} é o número de operações de sangramento de acabamento:

$$S_{gs} = \sum_{k=1}^{n_{gs}} \frac{\pi}{1000V_{cs}f_s} \left| \frac{1}{4} (D_{2(k)}^2 - D_{1(k)}^2) \right| \quad (C.44)$$

C.5 Modelo Matemático Passe Único para Tempo de Corte de Recartilhamento

O modelo de tempos é semelhante ao faceamento podendo ser calculado da mesma forma com um único passe. O tempo de corte para o passe de todos os recartilhados S_k pode ser obtido da equação (C.34) fazendo $sen\theta = \pm 1$ e n_k é o número de operações de recartilhamento:

$$S_k = \sum_{k=1}^{n_k} \frac{\pi}{1000V_{cs}f_s} \left| \frac{1}{4} (D_{2(k)}^2 - D_{1(k)}^2) \right| \quad (C.45)$$

C.6 Modelo Matemático Passe Único para Tempo de Corte de Furação e Alargamento

O tempo de corte de furação é dado por:

$$T_d = \frac{\pi D_t L_w}{1000V_c f} \quad (C.46)$$

$l_w = |(Z_2 - Z_1)|$ = comprimento da superfície a ser usinada na direção longitudinal

f = avanço (mm/revolução)

D_t = Diâmetro da ferramenta (mm)

V_c = Velocidade de Corte (m/min)

O tempo total de corte para furação é S_d , sendo n_d o número de furos na peça:

$$S_d = \sum_{i=1}^{n_d} T_{d(i)} \quad (\text{C.47})$$

C.7 Modelo Matemático Passe Único para Tempo de Corte de Rosqueamento Cilíndrico

O modelo de corte é semelhante ao torneamento cilíndrico para o passe de acabamento, podendo ser calculado para um único passe pelo modelo abaixo:

$$T_{th(i)} = \frac{\pi |D_{(i)}(Z_{2(i)} - Z_{1(i)})|}{500V_s f_s}$$

Onde $D(i)$ é diâmetro externo da rosca. O tempo total de corte para o passe de rosqueamento é S_{th} , sendo n_{th} o número de segmentos retilíneos com roscas (passes):

$$S_{th} = \sum_{i=1}^{n_{th}} T_{th(i)} \quad (\text{C.48})$$

C.8 Modelos de Otimização de Usinagem Propostos

C.8.1 Modelo Baseado no Mínimo Custo

A unidade de custo de produção é utilizada como critério de medida de otimização das condições de usinagem sendo utilizado o modelo apresentado por CHEN e SU (1998). São consideradas muitas restrições práticas, incluindo limites máximos e mínimos para as condições de usinagem, restrições de força de corte e potência da máquina, restrição de acabamento superficial, restrição de vida de ferramenta, restrições de região de corte estável e de temperatura interface cavaco/ferramenta/peça, restrição deflexão da peça, entre muitas outras menos significativas.

A função objetivo a ser otimizada leva em consideração o critério econômico associado ao mínimo custo de produção unitário, que inclui:

- o custo associado ao tempo real de usinagem da peça;
- o custo de máquina parada relativo às operações de carga e descarga da máquina e movimentação da ferramenta em vazio;
- custo de troca de ferramenta;
- e custo da ferramenta.

O custo de usinagem C_m (\$/peça) é dado por $C_m = k_o T_M$

onde k_o é a soma do custo direto e encargos sobre o trabalho de usinagem e T_M é o tempo de corte total calculado pelo somatório dos tempos de usinagem para os passes de usinagem (desbaste e acabamento) das operações necessárias para usinagem das *features* e que foram calculadas no item anterior, sendo

$$T_M = S_{r1} + S_{r2} + S_f + S_{hr} + S_{hs} + S_{gr} + S_{gs} + S_k + S_d + S_{th} \quad (C.49)$$

O custo de máquina parada (improdutiva) é C_l (\$/peça) sendo calculado por

$$C_l = K_o T_I \quad (C.50)$$

onde T_I é o tempo de máquina parada, ou seja sem remoção de cavaco. Este custo é dividido em um termo constante (t_c) relativo ao tempo de carga e descarga da máquina e um termo variável (t_v) associado à movimentação da ferramenta em vazio, sem remoção de material. O tempo de movimentação da ferramenta sem remoção de material (t_v) pode ser representado como a distância transversal l_a (mm) dividido pela velocidade de avanço rápido da máquina V_{fa} (mm/min). Então

$$t_v = \frac{l_a}{V_{fa}} \quad (C.51)$$

A distância transversal rápida pode ser calculada para as operações de torneamento retilíneo, cônico, circular, faceamento e perfilamento para os passes associados ao primeiro e segundo estágio de desbaste e para o passe de acabamento (figura 7.30), sendo o somatório dos passes de usinagem igual a (n-1) no primeiro estágio de desbaste, mais um passe de desbaste no segundo estágio e um passe de acabamento, sendo, neste caso:

$$l_a = \sum_{g=1}^{n-1} H_g + \sqrt{2}(n-1)e + 2\overline{P_I P_M} + 2\overline{P_I P_L} - 2d_s \quad (C.52)$$

onde P_I é o ponto inicial, P_M é o ponto final e P_L é ponto de menor corte (menor diâmetro usinado na peça).

Deve-se agora incluir os outros movimentos de ferramenta no vazio para as demais operações de usinagem presentes na peça, que nada mais será do que um caso particular do caso mais genérico de l_a .

Para torneamento longitudinal a equação é a mesma. Para faceamento, recartilhamento, corte da peça e sangramento, os termos H_g e "e" são iguais a zero, devendo-se calcular os segmentos associados ao posicionamento $\overline{P_I P_M}$ e ao corte $\overline{P_I P_L}$ para operações de desbaste e acabamento, sendo o novo l_{af} dado por

$$l_{af} = 2\overline{P_I P_M} + 2\overline{P_I P_L} - 2d_s \quad (C.53)$$

Para furação o termo d_s é nulo, pois não há operação de acabamento. Este cálculo assume a utilização da mesma ferramenta para operações de desbaste e acabamento. No caso de haver uma troca de ferramenta para operações de acabamento, deve-se incluir o cálculo do deslocamento associado à movimentação até o plano de aproximação e de troca de ferramenta, por exemplo.

Assim o tempo de máquina parada T_I pode ser calculado incluindo o somatório de todas as contribuições de movimentação de ferramenta, associada a todas as operações de usinagem realizada na peça para remoção de todas as *features* de usinagem (n_{fu}) determinadas, sendo:

$$T_I = t_c + t_v = t_c + \frac{l_a}{V_{fa}} \quad (C.54)$$

O custo de máquina parada (improdutiva) será

$$C_l = k_o T_I = k_o \left\{ \sum_{j=1}^{n_{fu}} \left[t_c + \frac{\sum_{g=1}^{n-1} H_g + \sqrt{2}(n-1)e + 2\overline{P_I P_M} + 2\overline{P_I P_L} - 2d_s}{V_{fa}} \right] \right\} \quad (C.55)$$

O custo de troca de ferramenta C_R (\$/peça) é calculado por

$$C_R = k_o T_R \quad (C.56)$$

onde T_R é o tempo associado à troca de ferramenta (min). O tempo associado à troca de ferramenta pode ser escrito em termos da vida da ferramenta t_l (min), do tempo requerido para trocar a ferramenta t_e (min) e o tempo de corte T_M , sendo

$$T_R = t_e \frac{T_M}{t_l} \quad (C.57)$$

onde $\left(\frac{T_M}{t_l}\right)^{-1}$ representa o número de peças que podem ser produzidas antes que a ferramenta seja trocada.

A equação de vida da ferramenta é dado pela fórmula de Taylor expandida:

$$t = \frac{C_0}{V^\alpha f^\beta a_p^\gamma} \quad (C.58)$$

onde α, β, γ e C_0 são as constantes da fórmula de Taylor para uma determinada ferramenta de corte usinar um determinado material, par peça-ferramenta. Pode-se utilizar a mesma ferramenta de corte para os passes de desbaste e acabamento ou ferramentas diferentes, necessitando assim de modelos específicos de vida de ferramenta para cada caso, a fim de se evitar erros grosseiros na definição dos modelos, pois o desgaste de ferramenta é diferente para operações de desbaste e acabamento.

Quando se utiliza a mesma ferramenta para operações de desbaste e acabamento a vida da ferramenta t_l deve ser calculada segundo SHIN e JOO (1992) por uma composição balanceada da taxa de desgaste para desbaste e acabamento, dado por

$$t_l = w t_r + (1 - w) t_s \quad (C.59)$$

onde $t_r = \frac{C_0}{V_r^\alpha f_r^\beta a_{pr}^\gamma}$ representa a vida para desbaste

$t_s = \frac{C_0}{V_s^\alpha f_s^\beta a_{ps}^\gamma}$ representa a vida para acabamento; e w é um peso da equação de vida variando no intervalo aberto (0,1), sendo definido de maneira empírica.

O custo de troca de ferramenta C_R para cada ferramenta pode ser expresso por

$$C_R = k_o T_R = k_o t_e \frac{T_M}{t_l} \quad (C.60)$$

O custo da ferramenta C_T (\$/peça) é calculado por

$$C_T = k_t \frac{T_M}{t_l} \quad (C.61)$$

onde k_t é o custo por gume (aresta de corte) da ferramenta (\$/gume).

O custo total de produção por unidade produzida UC (\$/peça) é

$$UC = C_M + C_l + C_R + C_T \quad (C.62)$$

$$UC = k_o T_M + k_o T_I + k_o \left(t_e \frac{T_M}{t_l} \right) + k_t \left(\frac{T_M}{t_l} \right)$$

$$UC = \sum_{t=1}^{n_{tool}} \left[k_{o(n_{tool})} T_M + k_{o(n_{tool})} T_I + k_{o(n_{tool})} \left(t_e \frac{T_M}{t_l} \right) + k_{t(n_{tool})} \left(\frac{T_M}{t_l} \right) \right] \quad (C.63)$$

n_{tool} é o número total de ferramentas utilizadas na fabricação da peça. Assim o custo deve ser calculado para cada ferramenta utilizada, sendo esta a função objetivo a ser otimizada para o menor custo de produção.

C.8.2 Modelo Baseado na Máxima Taxa de Produção

O menor tempo total de produção da peça é utilizada como critério de medida de otimização das condições de usinagem sendo utilizado o modelo apresentado por BAYKASOGLU e DERELI (2002) e DERELI *et al.* (2001). São também consideradas as mesmas restrições práticas aplicadas ao modelo de custo, durante o processo de otimização da função objetivo.

A função objetivo a ser otimizada leva em consideração o critério econômico associado ao mínimo tempo de produção unitário, utilizando o modelo de tempo apresentado anteriormente, que inclui:

- tempo real de usinagem da peça (T_M);
- tempo máquina parada (improdutivo) relativo às operações de carga e descarga da máquina, setup e movimentação da ferramenta em vazio (T_I);
- tempo troca de ferramenta (T_T).

O tempo total de produção por unidade produzida UT (min) é

$$UT = T_M + T_I + T_T \quad (C.64)$$

$$UT = T_M + T_I + \left(t_e \frac{T_M}{t_l} \right) \quad (C.65)$$

Sendo esta a função objetivo a ser otimizada para o menor tempo de produção.

C.8.3 Restrições Práticas de Usinagem

Restrições práticas são impostas durante as operações de desbaste e acabamento. As restrições das condições de usinagem referem-se as faixas de utilização (valores mínimo e máximo) da velocidade de corte, avanço e profundidade de corte disponíveis na base de dados de ferramentas, associadas ao par material da peça-ferramenta. Deve-se levar em conta as limitações de rotação e avanço da máquina-ferramenta escolhida, também disponível na base de dados. A seguir são apresentadas as restrições utilizadas na otimização das condições de usinagem para os dois critérios de otimização utilizados (TAN e CREESE, 1995 & CHANG *et al.*, 1998).

Principais Restrições Associadas às Operações de Desbaste⁴

- Restrição na velocidade de corte: $V_{crL} \leq V_{cr} \leq V_{crU}$
- Restrição no avanço: $f_{rL} \leq f_r \leq f_{rU}$
- Restrição na profundidade de corte: $a_{prL} \leq a_{pr} \leq a_{prU}$
- Restrição na vida da ferramenta: $T_L \leq t_r \leq T_U$
- Restrição na força de corte: $F_r = k_f f_r^\mu a_{pr}^\nu \leq F_U$, onde F_r é a força de usinagem durante o desbaste; k_f , μ e ν são as constantes da equação de Kienzle para o par peça/ferramenta, e F_U é a máxima força de usinagem disponível.
- Restrição força de corte calculada em uma viga engastada e carregada à uma distância L: $F_r = \frac{3EIR\delta}{L^3} \leq F_U$, onde E é o módulo de elasticidade do material, I é o momento de inércia, R é o raio da peça, L é o comprimento da peça e δ é a deflexão da peça no ponto de aplicação da força.
- Restrição na potência de corte: $P_r = \frac{k_f f_r^\mu a_{pr}^\nu V_r}{6120\eta} \leq P_U$, onde P_r é a potência de corte durante a usinagem (kW), η é o fator de eficiência associado a potência da máquina e P_U é a potência máxima disponível para corte (KW).
- Restrição associada a região de corte estável (NARANG e FISCHER, 1993): $V_r^\lambda f_r a_{pr}^\nu \geq S_L$ onde λ e ν são constante associadas ao par peça/ferramenta e S_L é o limite da região de corte estável.

Principais Restrições Associadas às Operações de Acabamento⁵

- Restrição na velocidade de corte: $V_{csL} \leq V_{cs} \leq V_{csU}$
- Restrição no avanço: $f_{sL} \leq f_s \leq f_{sU}$
- Restrição na profundidade de corte: $a_{psL} \leq a_{ps} \leq a_{psU}$
- Restrição na vida da ferramenta: $T_L \leq t_s \leq T_U$

⁴Os índices $_{rL}$ e $_{rU}$ referem-se aos limites inferior e superior para cada restrição considerada nas operações de desbaste.

⁵Os sufixos $_{sL}$ e $_{sU}$ referem-se aos limites inferior e superior para cada restrição considerada nas operações de acabamento.

- Restrição na força de corte: $F_s = k_f f_s^\mu a_{ps}^\nu \leq F_U$, onde F_s é a força de usinagem durante o acabamento; k_f , μ e ν são as constantes da equação de Kienzle para o par peça/ferramenta, e F_U é a máxima força de usinagem disponível.
- Restrição na potência de corte: $P_s = \frac{k_s f_s^\mu a_{ps}^\nu V_s}{6120\eta} \leq P_U$, onde P_r é a potência de corte durante a usinagem (kW), η é o fator de eficiência associado a potência da máquina e P_U é a potência máxima disponível para corte (KW).
- Restrição associada à região de corte estável (NARANG e FISCHER, 1993): $V_s^\lambda f_s a_{ps}^\nu \geq S_L$ onde λ e ν são constante associadas ao par peça/ferramenta e S_L é o limite da região de corte estável.

- Restrição acabamento superficial:

$$f_s \leq \sqrt{8r_\epsilon Rt} \tag{C.66}$$

ou

$$f_s \leq \sqrt{31r_\epsilon Ra} \tag{C.67}$$

, pois $\frac{Ra}{Rt} = 0,256$, onde r_ϵ é raio arredondamento da ponta da ferramenta, Ra é a rugosidade média e Rt é a rugosidade total.

$$f_{sU} \leq 0,8 \left[\frac{8 * Ra * 10^{-3}}{2 * 0,256} \right]^{0,5} \tag{C.68}$$

Para $Ra \leq 3,2\mu m$ ou

$$f_{sU} \leq 0,8 \left[\frac{8 * 1,6 * Ra * 10^{-3}}{0,256} \right]^{0,5} \tag{C.69}$$

Para $Ra > 3,2\mu m$

- Restrição da profundidade de corte como função da dureza Brinell (BNH) do material e da rugosidade superficial Ra :

$$a_{psU} \leq \frac{32Ra}{BNH^{0,8}} \tag{C.70}$$

Restrições Miscelâneas

- Relacionamentos entre os parâmetros de desbaste e acabamento, segundo CHANG *et al.* (1991), são apresentados a seguir:

$$V_s > k_1 V_r ; \quad f_r > k_2 f_s ; \quad a_{pr} > k_3 a_{ps}$$

onde k_1, k_2, k_3 são coeficientes de relacionamento e $k_1, k_2, k_3 \geq 1$

- Restrição de profundidade de corte total levando em conta a ocorrência de operações de desbaste e acabamento: a profundidade de corte de acabamento (a_{ps}) deve ser igual a máxima profundidade de corte a ser removida da peça bruta (a_p) menos o somatório da profundidade de corte para cada passe de desbaste. Os passes de desbaste podem ter a mesma profundidade de corte ou profundidades de cortes diferentes. Quando tiverem a mesma profundidade de corte a profundidade total será dada por $n * a_{pr}$. Quando as profundidades de corte forem diferentes, o

somatório das profundidades de corte a ser removida nos n passes de desbaste será $\sum_{i=1}^n a_{pr(i)}$.

Logo

$$a_{ps} = a_p - \sum_{i=1}^n a_{pr(i)} \text{ ou } a_{ps} = a_p - n \star a_{pr} \quad (\text{C.71})$$

Os limites para o número de passes de desbaste é dado por

$$N_L \leq n \leq N_U \quad (\text{C.72})$$

onde os limites superior e inferior do número de passes são

$$N_L = \frac{a_p - a_{ps}}{a_{prU}}, \quad N_U = \frac{a_p - a_{ps}}{a_{prL}} \quad (\text{C.73})$$

No caso do modelo trabalhar com a mesma profundidades de corte para os passes de desbaste, tem-se para cada possível profundidade de corte de acabamento (a_{ps}) uma correspondente profundidade de corte para desbaste, que pode ser calculado por:

$$a_{pr} = \frac{a_p - a_{ps}}{n} \quad (\text{C.74})$$

Para o caso de passes de desbaste com profundidades diferentes, tem-se que o número de passes de desbaste ($n_{(i)}$) é variável, sendo calculado a partir da máxima profundidade de corte permitida para operações de desbaste (limite superior, a_{prU}) e das profundidades de cortes para o passe de acabamento ($a_{ps(i)}$), que varia entre os limites superior e inferior permitido pela ferramenta para a operação de acabamento, sendo:

$$n_{(i)} = \frac{a_p - a_{ps(i)}}{a_{prU}} \quad (\text{C.75})$$

Desta forma ter-se-á duas abordagens com profundidade de corte de desbaste fixa ou variável, aumentando a complexidade do modelo.

Apêndice D

Ciclo de Vida do Software WebMachining Modelado em UML

Este apêndice apresenta a modelagem visual do ciclo de vida de desenvolvimento do software WebMachining, baseado em UML e focado na dimensão componente de processo envolvendo os estágios de análise, projeto, implementação e uso¹. Para cada um dos três módulos do sistema WebMachining (WebCadbyFeatures, WebCAPP e WebTurning), são analisados *use-cases*, usando a notação UML e a ferramenta Rational Rose 2003.

D.1 Introdução

A metodologia utilizada no ciclo de desenvolvimento do software é baseada em UML utilizando a ferramenta computacional Rational Rose 2003 (QUARTRANI, 2000), que emprega o processo de desenvolvimento de *software Rational Unified Process*, sendo estruturado em duas dimensões: tempo (fases) e componentes de processo (atividades). A metodologia divide o ciclo de desenvolvimento de software em cinco estágios (fluxo de trabalho do projeto), representado pelas atividades: requisitos, análise, projeto, implementação e teste (uso). Já as fases (tempo) que ocorrem em cada atividade são divididas em: concepção, elaboração, construção e transição.

Utilizou-se como abordagem, uma implementação rápida de protótipos com funcionalidades restritas, desenvolvendo rapidamente as fases associadas ao ciclo de vida do software de forma iterativa e incremental, para se chegar à versão final do sistema WebMachining.

Os estágios (atividades) do desenvolvimento do software são documentados através de diagramas UML a partir de um estudo de requisitos (exigências) do sistema modelado baseado em *use-cases*, que definem um modelo de comando de utilização, apresentando as funções pretendidas pelo sistema WebMachining. A tabela D.1 apresenta os *use-cases* para os três sub-sistemas do ambiente integrado WebMachining.

Cada estágio do ciclo de desenvolvimento de software utiliza-se de determinados diagramas UML, sendo utilizados os seguintes diagramas na documentação do sistema WebMachining:

1. Requisitos: *use-cases diagrams*;

¹Uso e teste do sistema WebMachining são analisados nos capítulos 9, 10 e 11.

Tabela D.1: *Use-cases* para os três sub-sistemas do ambiente integrado WebMachining.

WebCADbyFeatures
cria novo projeto
cria <i>feature</i>
gera VRML
projeto colaborativo
salva arquivo biblioteca de <i>features</i>
WebCAPP
inicia CAPP lendo arquivo biblioteca de <i>features</i>
gera plano de processo
salva arquivo código G
WebTurning
<i>download</i> arquivo código G para máquina-ferramenta
monitoração visual chão-de-fábrica
teleoperação máquina-ferramenta

2. Análise: *activity diagrams*;
3. Projeto: *class diagrams & sequence diagrams*;
4. Implementação: *component diagrams & deployment diagrams*;
5. Teste: execução dos programas e estudos de casos.

D.2 WebCADbyFeatures

O WebCADbyFeatures permite a criação e a manipulação do modelo de *features* para a peça bruta e acabada, o armazenamento dessas informações em um banco de dados, a validação do modelo de *features* e a visualização do modelo geométrico em 2D e 3D. É constituído por uma GUI que suporta a modelagem por *features*. Esta interface CAD possui menus, opções de visualização, mensagens de erro, manipulação de *features*, comunicação com o servidor do banco de dados, entre outras funcionalidades.

Os principais componentes são: a GUI, a biblioteca de *features*, Interface Gráfica 2D, componentes de visualização 2D (primitivas gráficas como retas e arcos) e componentes de visualização 3D (VRML). As informações referentes às *features* são manipuladas através de um sistema de gerenciamento de banco de dados.

D.2.1 Use-Case Diagram

O principal *use-case* do WebCADbyFeatures é relativo ao projeto de uma peça. Após o usuário efetuar o *login* no sistema, deve-se escolher entre duas opções distintas: projetar uma nova peça ou abrir um projeto já existente. Após essa opção, é apresentado ao usuário a tela principal do WebCADbyFeatures, com todas as opções de projeto por *features*. O usuário pode, então, editar a peça, ou seja, inserir, modificar parâmetros, visualizar a peça em 3D (VRML) ou mesmo excluir *features*.

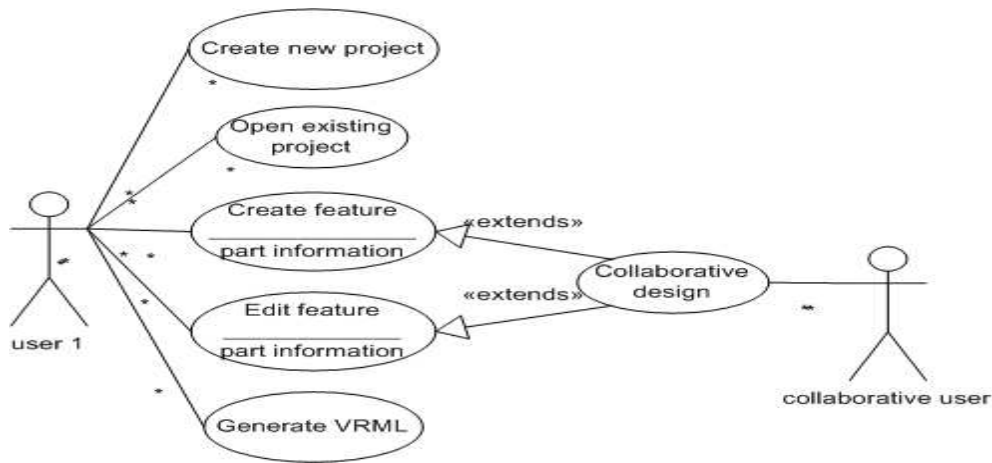


Figura D.1: Diagrama Use-Case do WebCADbyFeatures.

Nesta fase pode-se também utilizar a opção de projeto colaborativo, que será melhor detalhada no próximo item. A Figura D.1 descreve o diagrama de *use-case* do WebCADbyFeatures.

D.2.2 Activity Diagram

O Diagrama de Atividade do WebCADbyFeatures é mostrado na figura D.2, e descreve de maneira geral o fluxo de dados dentro desse subsistema. Na figura D.3, é mostrado o Diagrama de Atividade de projeto colaborativo, em que um usuário (Agente de Interface) conecta-se a outro utilizando-se do padrão de linguagem de mensagens descrito no KQML. Nesta fase um usuário pode compartilhar informações de projeto com o outro agente, bem como trocar dados e mensagens de texto, possibilitando, assim, a colaboração no desenvolvimento da peça.

D.2.3 Modelo Estático

O WebCADbyFeatures utiliza-se de uma biblioteca de *features*, na qual são definidos todos os tipos de *features* e seus parâmetros e relacionamentos geométricos e não-geométricos. Utiliza-se também de classes oriundas do JATLite (*Java Agent Template Lite*), uma biblioteca que permite o rápido desenvolvimento de sistemas multiagentes. As relações entre o pacote principal do WebCADbyFeatures, o pacote Java, a biblioteca de *features* e o pacote JATLite, são mostrados na figura D.4, utilizando-se um modelo estático de classes UML. As figuras D.5 e D.6 apresentam os diagramas de classe para *ODStraight* e para *CADAgentFrame* (pacoteProtocolLayer.Example.CAD).

D.2.4 Modelo Dinâmico

Na tentativa de ilustrar parte do modelo dinâmico do WebCADbyFeatures, recorreu-se à análise de um *use-case* generalizado de inserção de uma *feature* qualquer, quando do modelamento de uma peça. A figura D.7 mostra o Diagrama de Seqüência para esse *use-case* do WebCADbyFeatures.

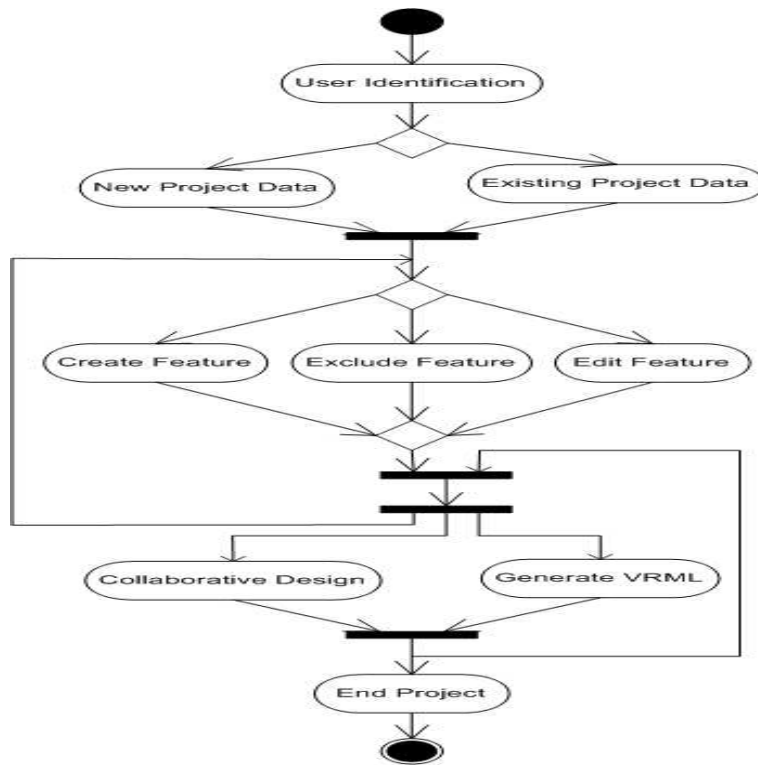


Figura D.2: Diagrama de Atividade do WebCADbyFeatures.

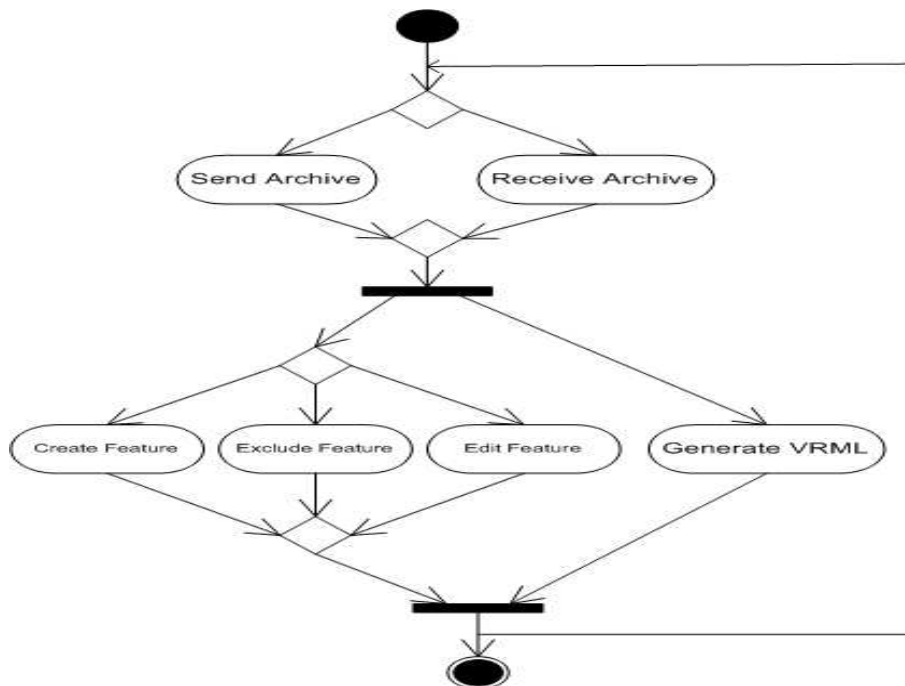


Figura D.3: Diagrama de Atividade do projeto colaborativo.

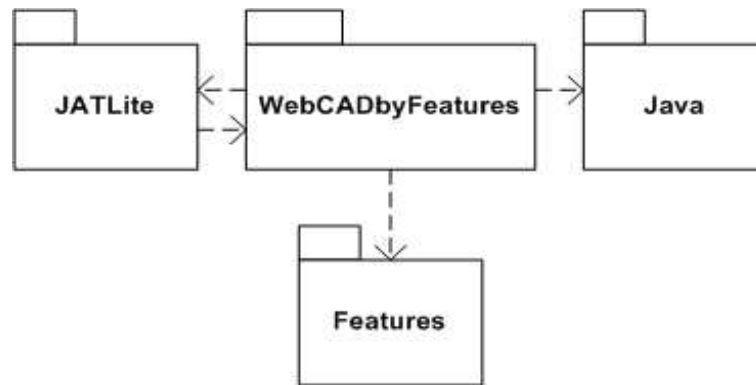


Figura D.4: Diagrama Estático WebCADbyFeatures.

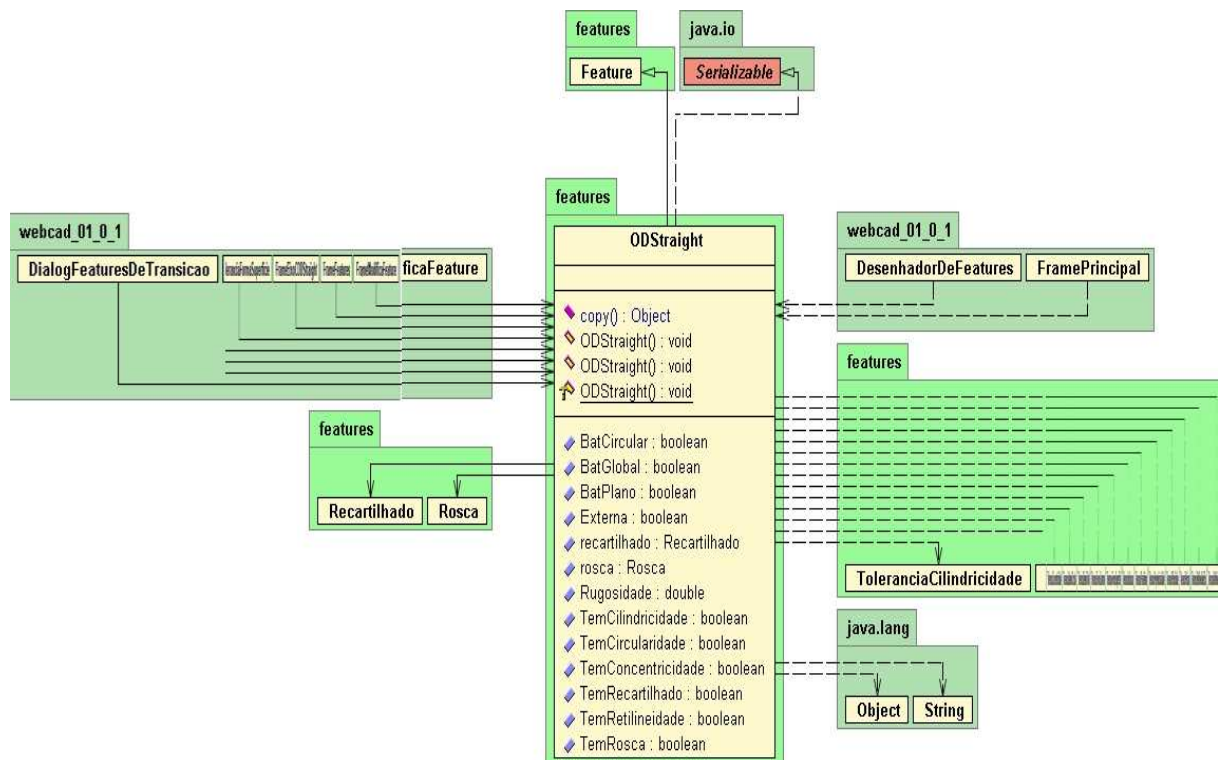


Figura D.5: Diagrama de classe (modelo estático) para a classe *ODStraight* (super classe *Feature* e pacote *features*) e seu relacionamento com as demais classes e pacotes *WebCADbyFeatures*.

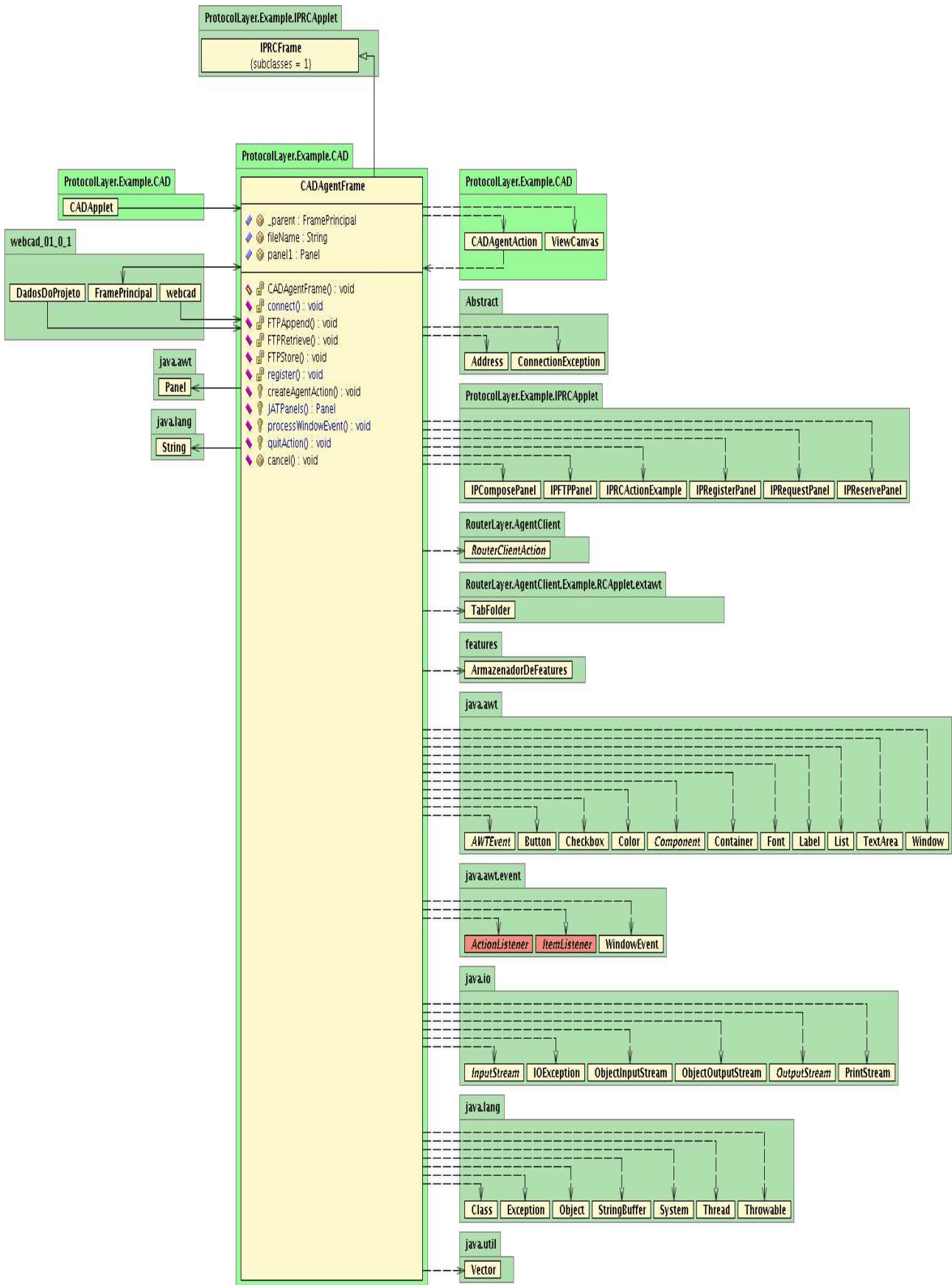


Figura D.6: Modelo estático para a classe `CADAgentFrame` (pacote `ProtocolLayer.Example.CAD`) associado ao `JATLite` (projeto colaborativo, agente de interface) e seu relacionamento com as demais classes e pacotes `WebCADbyFeatures`.

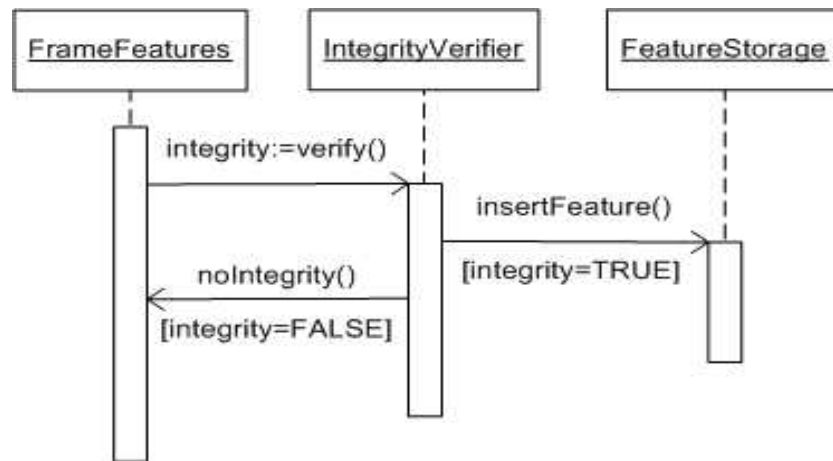


Figura D.7: Diagrama de Seqüência para use-case de inserção de uma *feature* qualquer do WebCADbyFeatures.

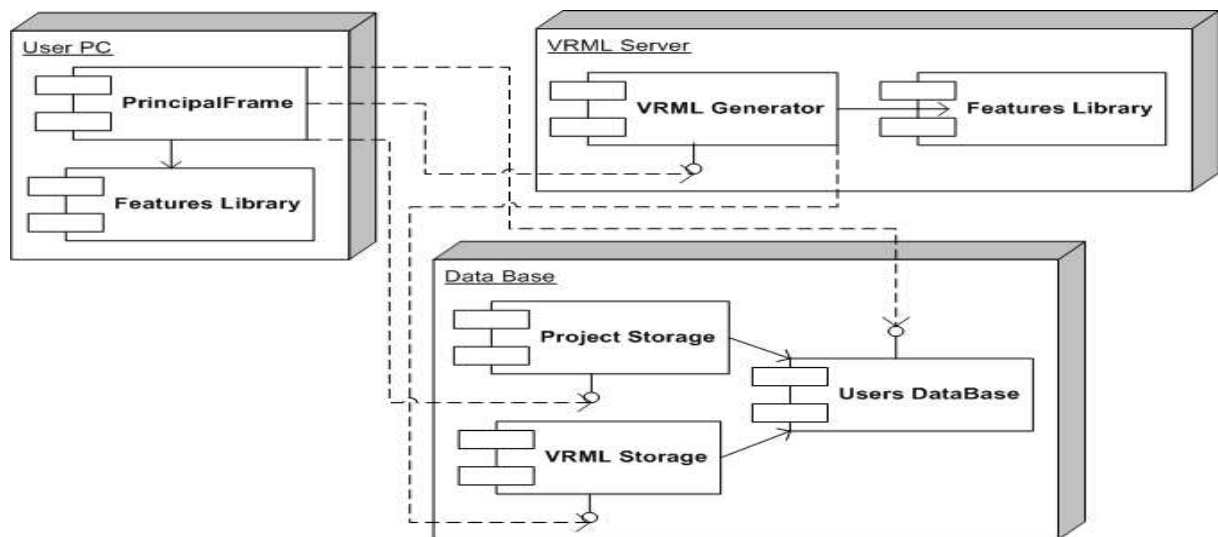


Figura D.8: *Component and Deployment Diagrams* do WebCADbyFeatures.

Neste *use-case*, tem-se um *FrameFeatures*, que possibilita a parametrização de uma determinada *feature* escolhida pelo usuário, um *VerificadorDeIntegridade*, que verifica os parâmetros inseridos pelo usuário, avisando-o caso algum parâmetro esteja incorreto. Se a *feature* tiver algum parâmetro incorreto, o objeto *VerificadorDeIntegridade* emite uma mensagem de erro para o usuário. Já se a *feature* estiver íntegra, o *VerificadorDeIntegridade* envia uma mensagem ao objeto *ArmazenadorDeFeatures* o qual irá armazenar as informações referentes àquela *feature*.

D.2.5 Implementação do WebCADbyFeatures

O modelo físico do WebCADbyFeatures pode ser visto na figura D.8. Neste modelo estão descritos apenas os principais diagramas de Utilização, bem como os principais componentes de cada bloco. A implementação do WebCADbyFeatures pode ser acessada na URL http://webmachining.alvarestech.com/cad_by_features.

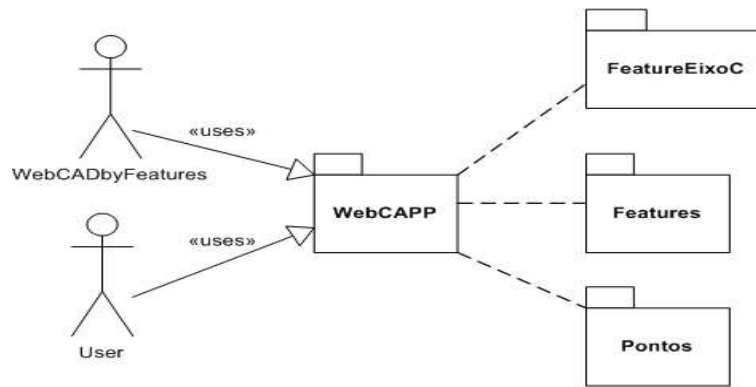


Figura D.9: Use-Case do WebCAPP.

D.3 WebCAPP

O planejamento de processo para a fabricação da peça é elaborado pelo WebCAPP, que é executado no servidor via *servlet* (pacote *WebCAPPservlet*) ou localmente via *applet* (*WebCAPPapplet*). Via *applet* são apresentadas todas as saídas associadas as dez atividades do WebCAPP, onde o usuário pode acompanhar as decisões tomadas pelo WebCAPP, entretanto é requerido um esforço computacional que o usuário talvez não possua, sendo necessário também uma máquina virtual Java versão 1.4, bem como ter instalado o software MatLab que executa os dois algoritmos genéticos. Como *servlet* o WebCAPP gera uma saída da representação do plano de processo linearizado no formato tabular e o programa NC, apresentados através da linguagem XML, sendo independente da máquina Java disponível no computador do usuário, bem como da disponibilidade do MatLab. A figura D.9 apresenta o *use-case* para o WebCAPP.

O pacote *WebCAPP servlet* recebe uma classe *ArmazenadorDeFeatures* de um projeto em desenvolvimento ou acabado de forma automática. Como *applet* é necessário salvar o projeto modelado por *features* localmente e carrega-lo via *applet* para que o mesmo seja executado e gerado o planejamento de processo. O WebCAPP pode ser chamado pelo usuário a partir de um projeto já salvo ou a partir do WebCADbyFeatures. Esse pacote se utiliza da biblioteca de *features* para reconhecer o desenho proveniente do *ArmazenadorDeFeatures*, bem como o pacotes *Pontos* e *FeaturesEixoC* para determinar os sistemas de coordenadas necessários para a elaboração do código G.

As atividades do CAPP iniciam-se com a normalização das *features* instanciadas pelo projetista no WebCADbyFeatures. O procedimento de normalização consiste em identificar as *features* e representá-las de uma única forma, evitando redundâncias em sua representação e facilitando o mapeamento de *features* para a visão de usinagem.

D.3.1 Decomposição Orientada ao Setup

Uma vez compatibilizado os dados da visão de projeto, inicia-se a primeira fase do CAPP, a decomposição orientada ao *setup*, que é responsável por determinar as possíveis *features* de fixação e superfícies delimitadoras, com alternativas.

Feature de fixação é a superfície de fixação da peça nas castanhas da máquina-ferramenta. Tal *feature* tem algumas restrições, como não poder conter superfícies secundárias, como roscas e recar-

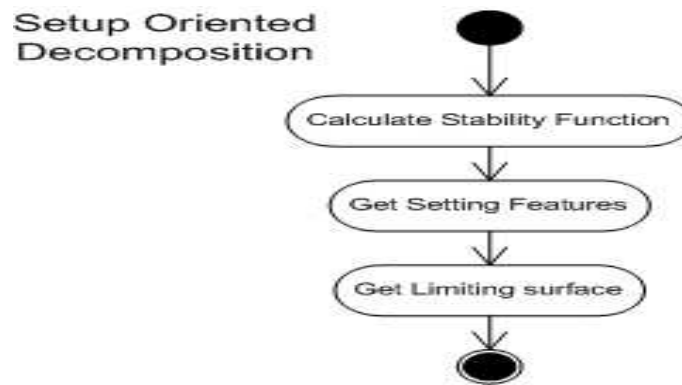


Figura D.10: *Use-Case* decomposição orientada ao *setup*.

tilhados, entre outras. A linha delimitadora, também denominada de superfície delimitadora, é uma linha que determina quais *features* serão feitas em qual *setup*.

Para determinar estas duas entidades, segue-se o procedimento descrito no diagrama de *use-case* da figura D.10. O método *CalculateStabilityFunction* verifica algumas características da peça para determinar o número de *setups* que serão necessários. Isso feito, iniciam-se os procedimentos para *GetSettingFeatures* e *GetLimitingSurface*, que definirão as superfícies de fixação e as superfícies (linhas) delimitadoras, respectivamente.

D.3.2 Decomposição Orientada à Geometria

Nesta fase definem-se os blocos geométricos funcionais que serão usinados em cada *setup*. Estas estruturas funcionais são as *features* de usinagem e de torneamento, que são armazenadas em um vetor que por sua vez está dentro de um vetor de *setups*. O diagrama *use-case* da decomposição orientada à geometria é apresentado na figura D.11.

De acordo com as diversas características, um conjunto de *features* pode ser interpretado como por exemplo: *feature* de usinagem de face, que possui como característica básica ter uma maior movimentação no eixo X em comparação com a movimentação no eixo Z; *feature* de torneamento (Geral) que possui características monotônicas (diâmetros somente crescentes ou somente decrescentes); e *feature* de torneamento especial, na qual os diâmetros crescem e decrescem, gerando reentrâncias. Além destas têm-se ainda as *features* de Eixo C, que são identificadas através de mapeamento 1:1, sendo um procedimento bem mais simples que o aplicado às *features* concêntricas.

D.3.3 Decomposição Orientada à Operação

A decomposição orientada à operação (fig. D.12) consiste em montar a informação adquirida numa estrutura de *Workingsteps* e *Workplans*, tal como o previsto pelo Step-NC. Nesta fase são montados os *Workingsteps*, determinando “n” operações de usinagem para cada *feature* de usinagem ou torneamento identificada, criando sub-geometrias se necessário. Nesta fase também se determinam os planos de segurança e aproximação, computando junto com os passos anteriores, todos os parâmetros necessários para se traçar uma boa estratégia de movimentação de ferramenta, que serão incluídos no procedimento de geração do código G.

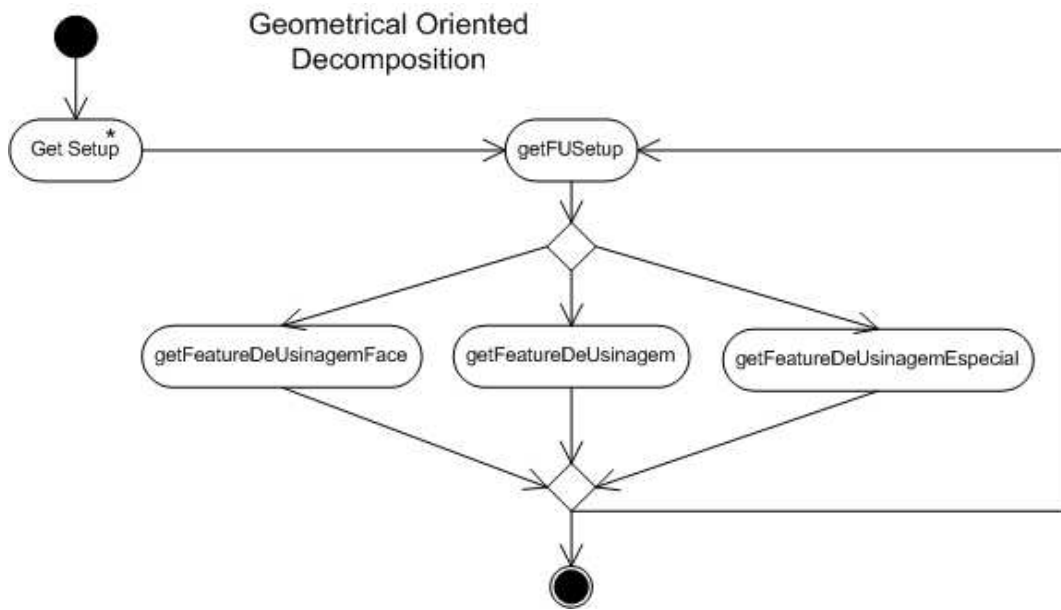


Figura D.11: Use-case decomposição orientada à geometria.

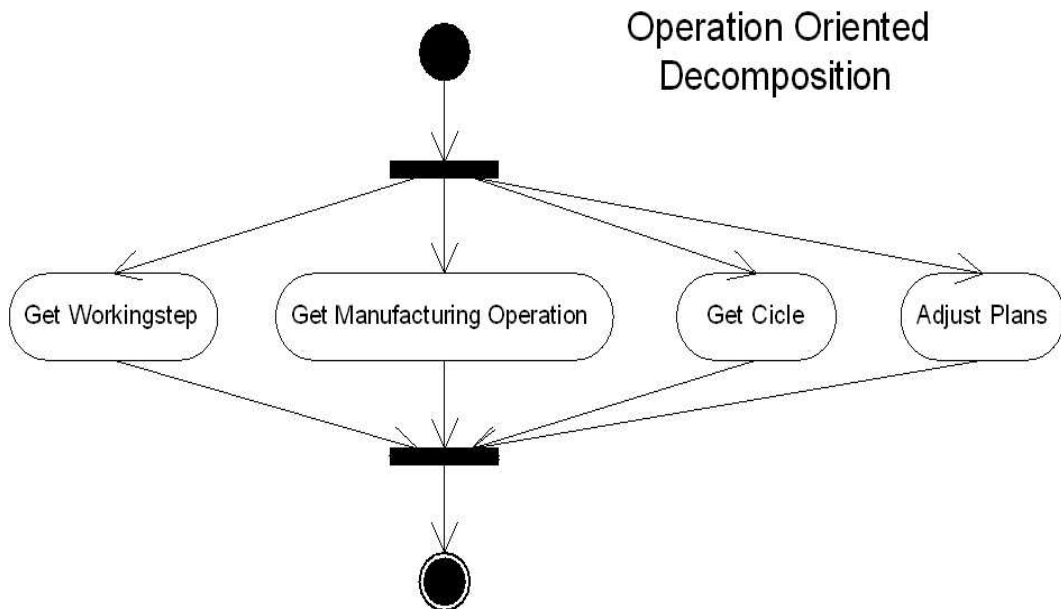


Figura D.12: Use-case decomposição orientada à operação.

D.3.4 Geração do Código G

A geração do Código G inicia-se com a leitura de cada *Setup* (*Setup1* e *Setup2*), onde é traçada uma estratégia para cada *feature* de usinagem com as informações agregadas durante as fases anteriores. Assim é gerado um arquivo texto que pode ser carregado na máquina-ferramenta via WebTurning ou enviado para um *Generate XML Process Planning*, que gera um arquivo XML para o usuário mostrando o plano de processo linearizado com o código G ao final do arquivo XML.

São também executados os módulos referentes à escolha de ferramenta e condições de usinagem, bem com o armazenamento do plano de processo com alternativas, visto que a cada etapa são geradas várias alternativas e ao final o algoritmo faz a linearização do plano de processo, segundo um determinado critério de otimização, armazenando as outras soluções como alternativas.

D.3.5 Modelo Estático: *Class Diagrams*

A figura D.13 apresenta o diagrama de classe principal (pacotes) associado ao módulo WebCAPP e o relacionamento com os demais pacotes.

A figura D.14 apresenta o diagrama de classe *Workingstep* que é composto pelas classes *FeatureDeUsinagem* e *OperaçãoDeUsinagem*.

A figura D.15 apresenta o diagrama de classe *FeatureDeUsinagem*.

A figura D.16 apresenta o diagrama de classe *OperacaoDeUsinagem* e seus relacionamentos com as classes *FerramentaUtilizada* e *TecnologiaDeUsinagem*.

A figura D.17 apresenta o diagrama de classe *CodigoG* e seus relacionamentos com as classes *FerramentaUtilizada*, *Maquina* e *InfoDecomposicaoOrientadaAOperacao*.

A figura D.18 apresenta o diagrama de classe *EncontrandoSuperficiesDelimitadoras* e seus relacionamentos com a classe *InfoDecomposicaoOrientadaAOperacao*.

D.4 WebTurning

São apresentados apenas os diagramas de classes associados ao pacote do sub-módulo WebCam. O sub-módulo WebCNC, associados aos programas desenvolvidos em linguagem C, não são orientados a objetos, ficando fora da modelagem UML.

A figura D.19 apresenta o diagrama de classe principal (pacotes) associado ao módulo NetCam (WebCam) e o relacionamento com os demais pacotes, que permite a visualização de vídeo e áudio em tempo real e vídeo gravado do chão-de-fábrica (GUI NetCAM).

A figura D.20 apresenta o modelo estático (diagrama de classes) para a classe *NetCAMAVI* (pacote com.AlvaresTech.NetCAM) e seu relacionamento com as demais classes e pacotes do módulo Web-Cam, que permite a gravação de vídeo em formato *.avi*.

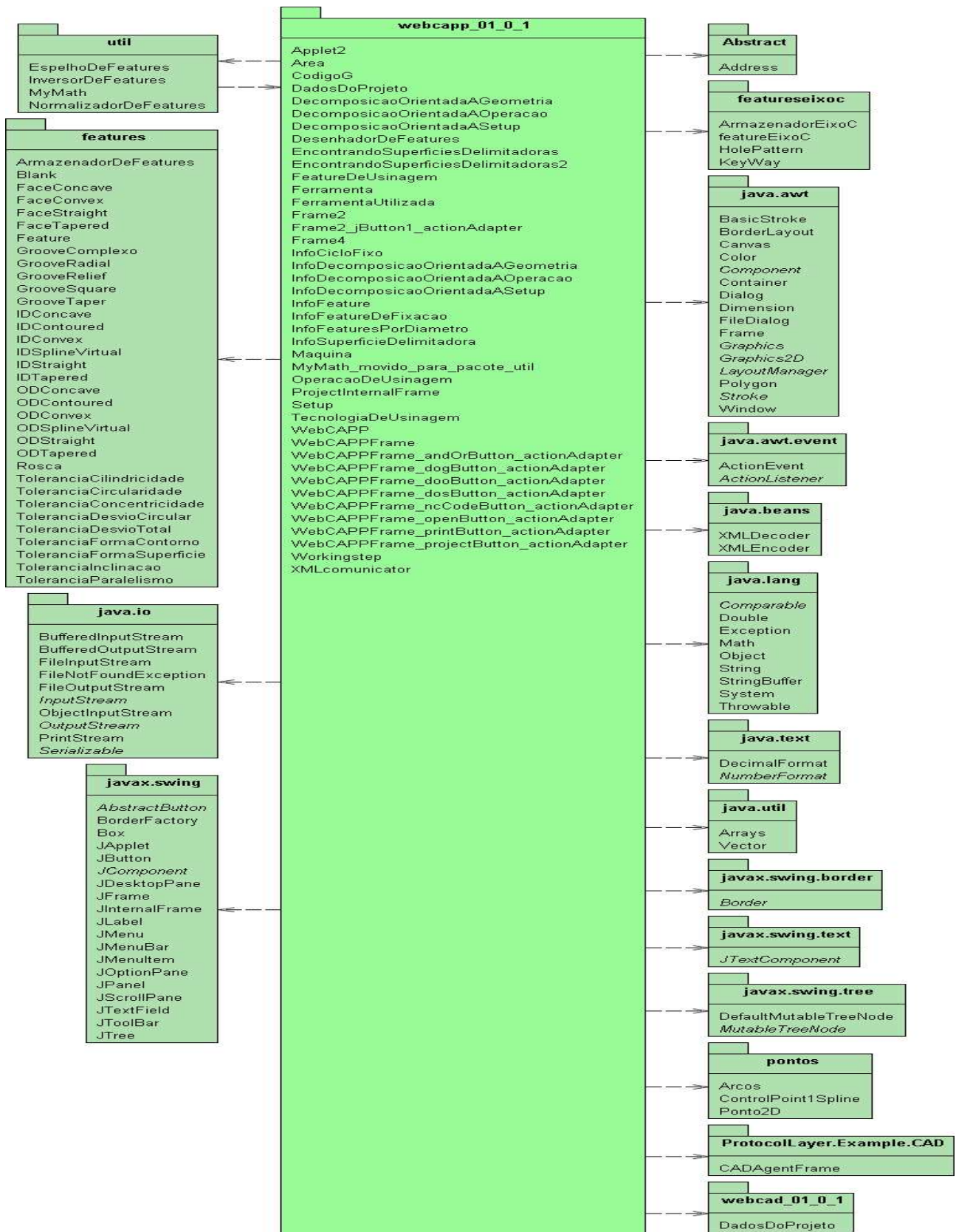


Figura D.13: Diagrama de classe principal (pacotes) associado ao módulo WebCAPP e relacionamento com demais pacotes.

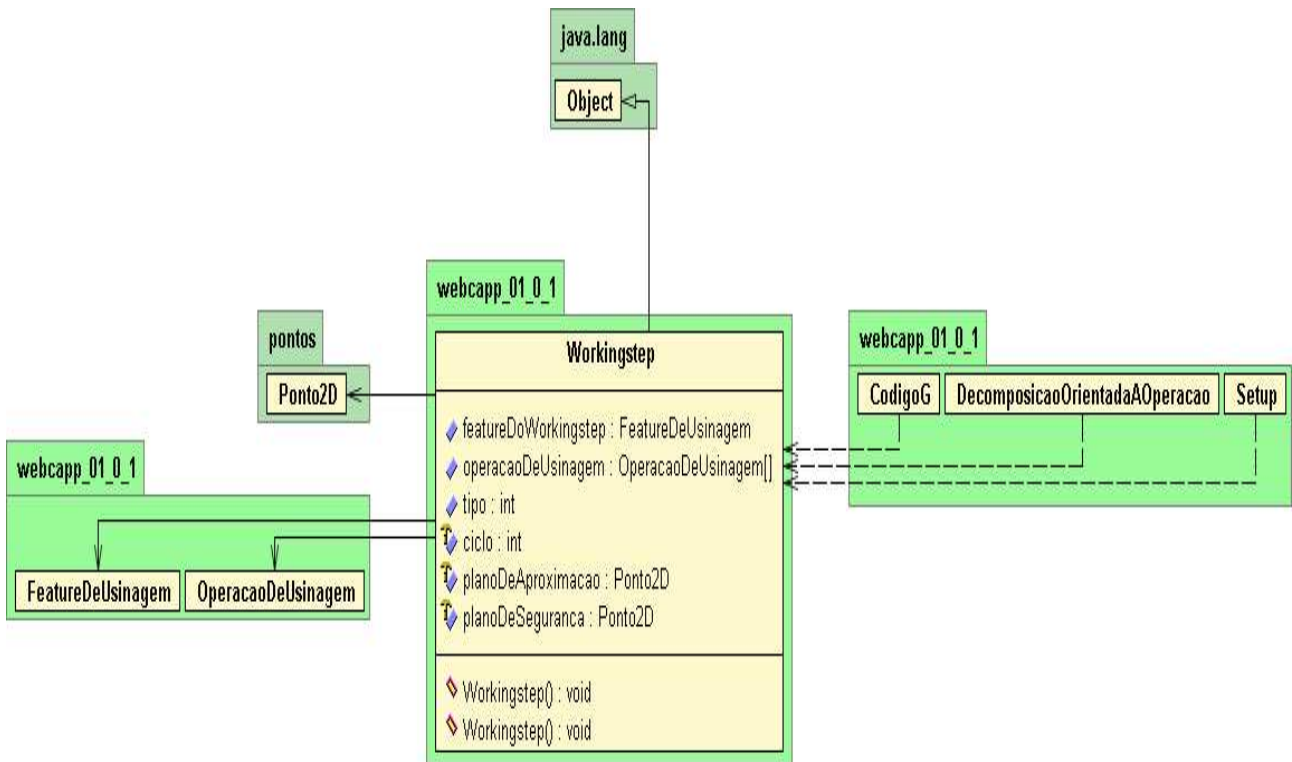


Figura D.14: Diagrama de classe *Workingstep* que é composto pelas classes *FeatureDeUsinagem* e *OperaçãoDeUsinagem*.

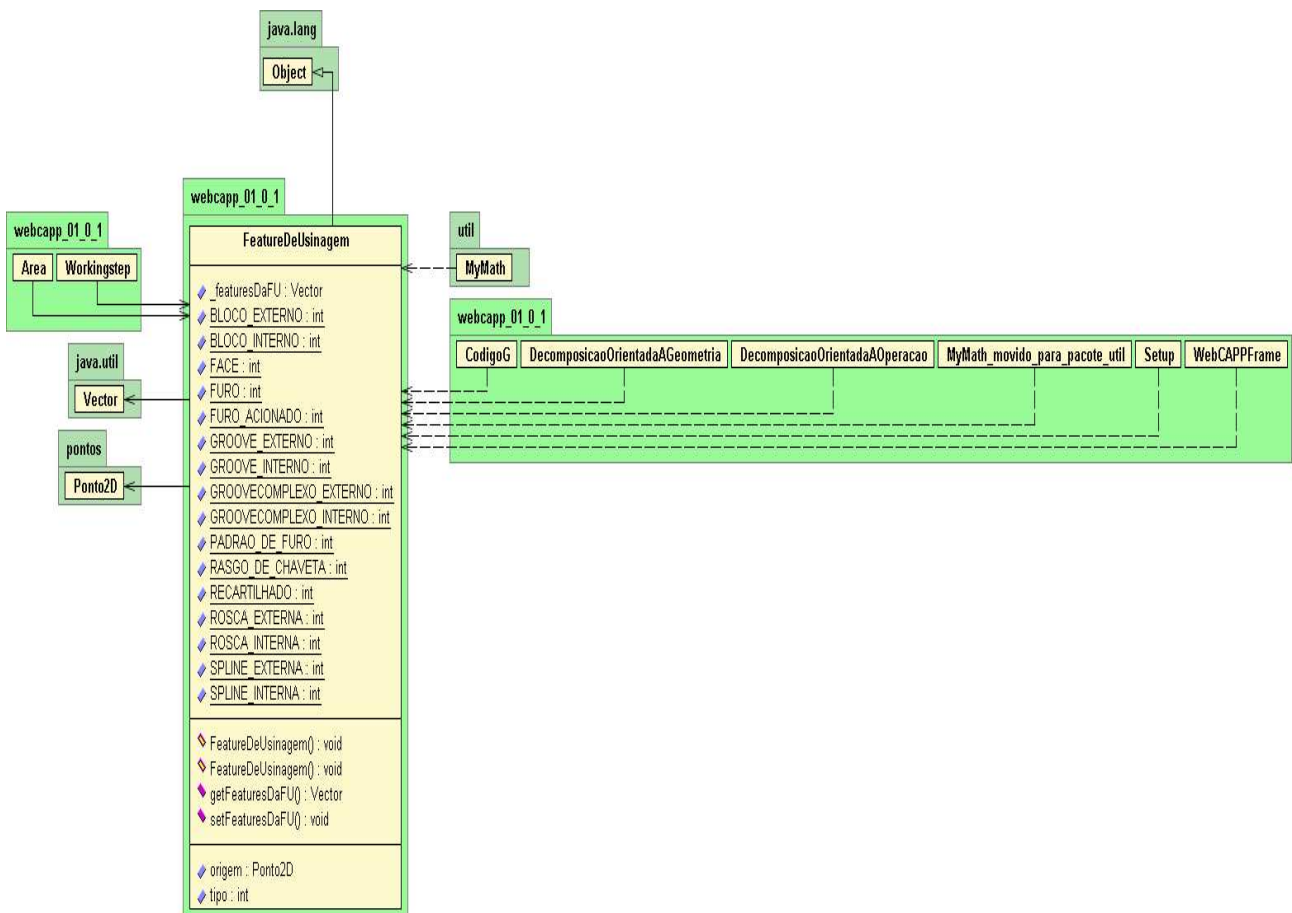


Figura D.15: Diagrama de classe *FeatureDeUsinagem*.

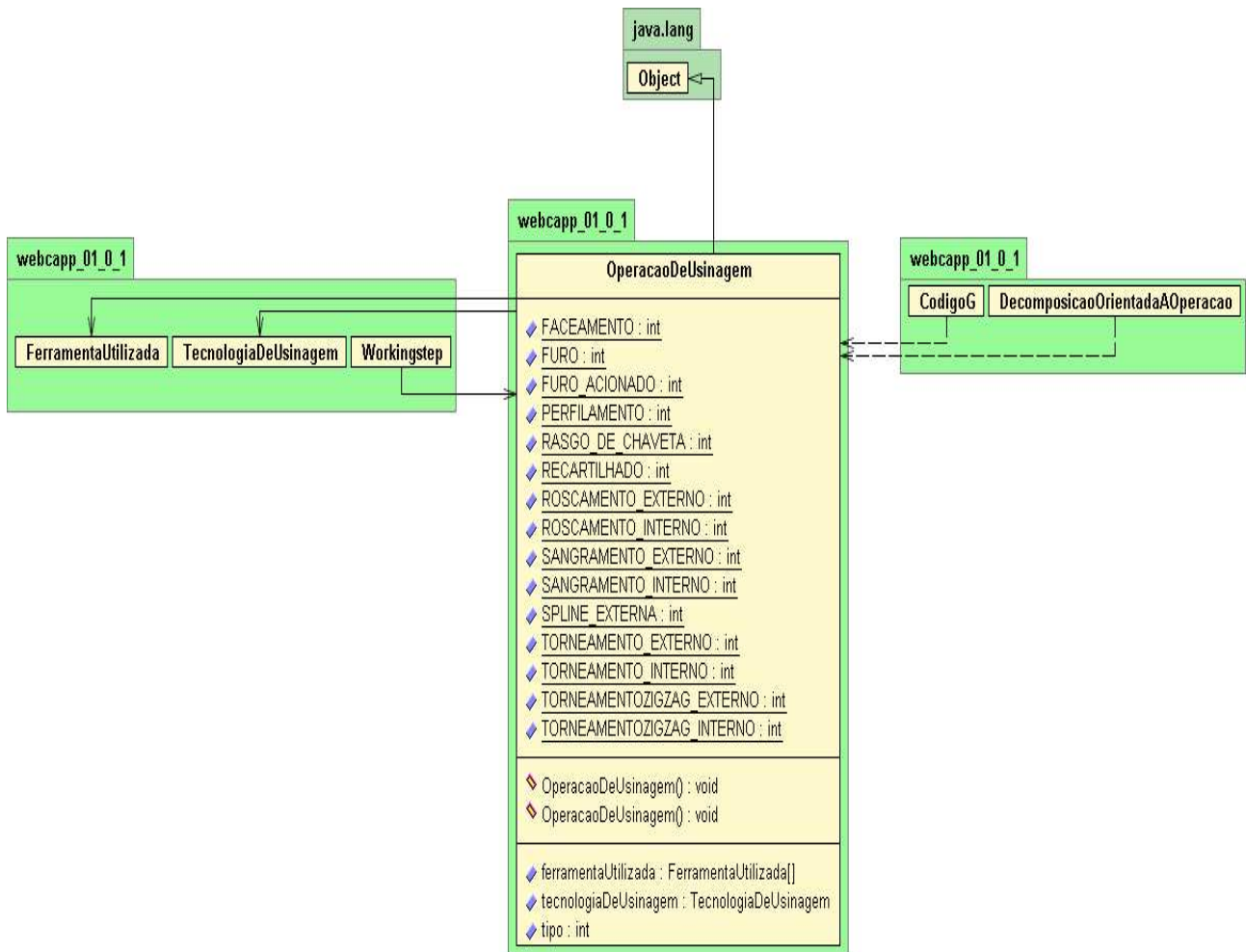


Figura D.16: Diagrama de classe *OperacaoDeUsinagem* e seus relacionamentos com as classes *FerramentaUtilizada* e *TecnologiaDeUsinagem*.

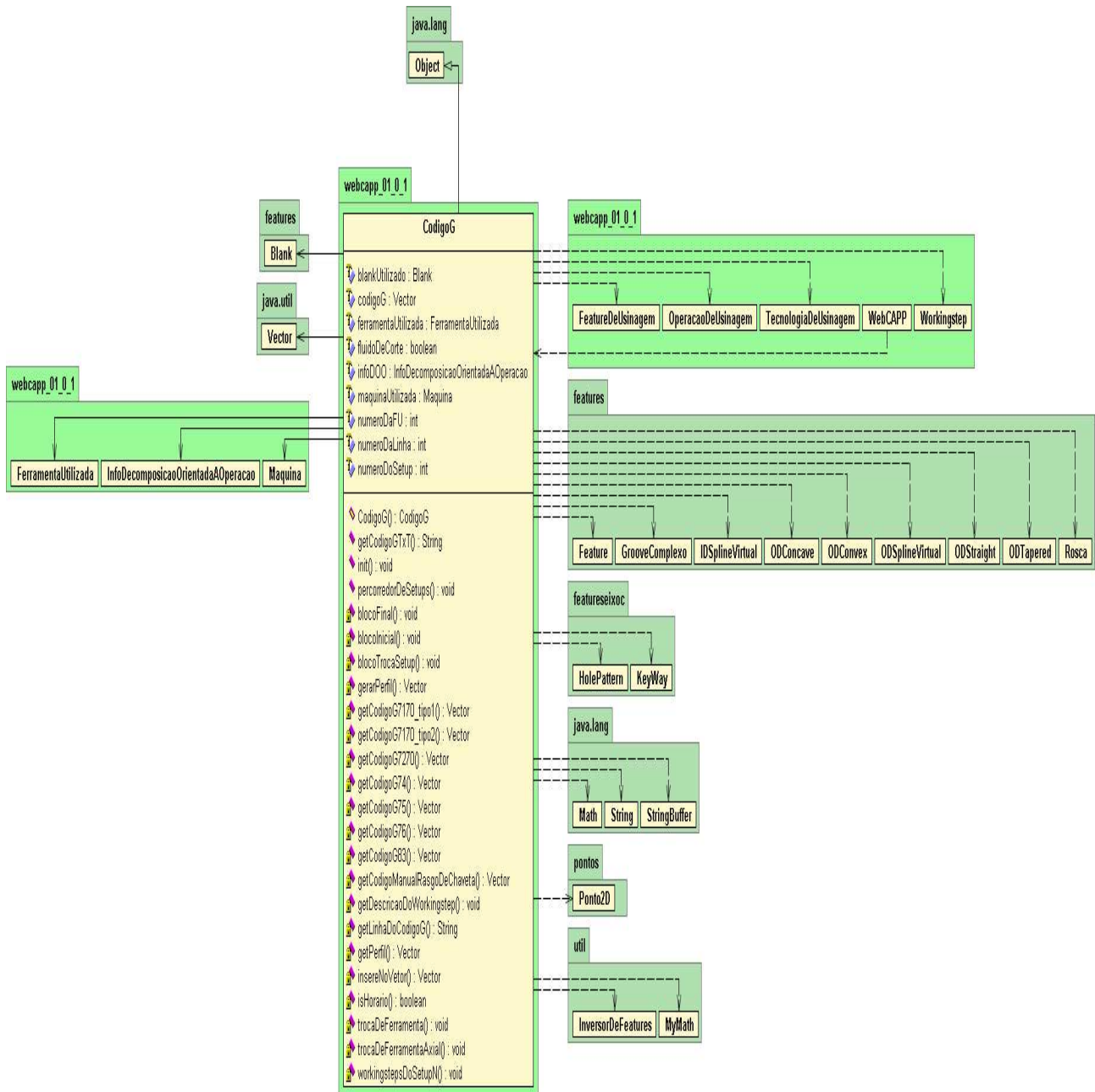


Figura D.17: Diagrama de classe `CodigoG` e seus relacionamentos com as classes `FerramentaUtilizada`, `Maquina` e `InfoDecomposicaoOrientadaAOperacao`.

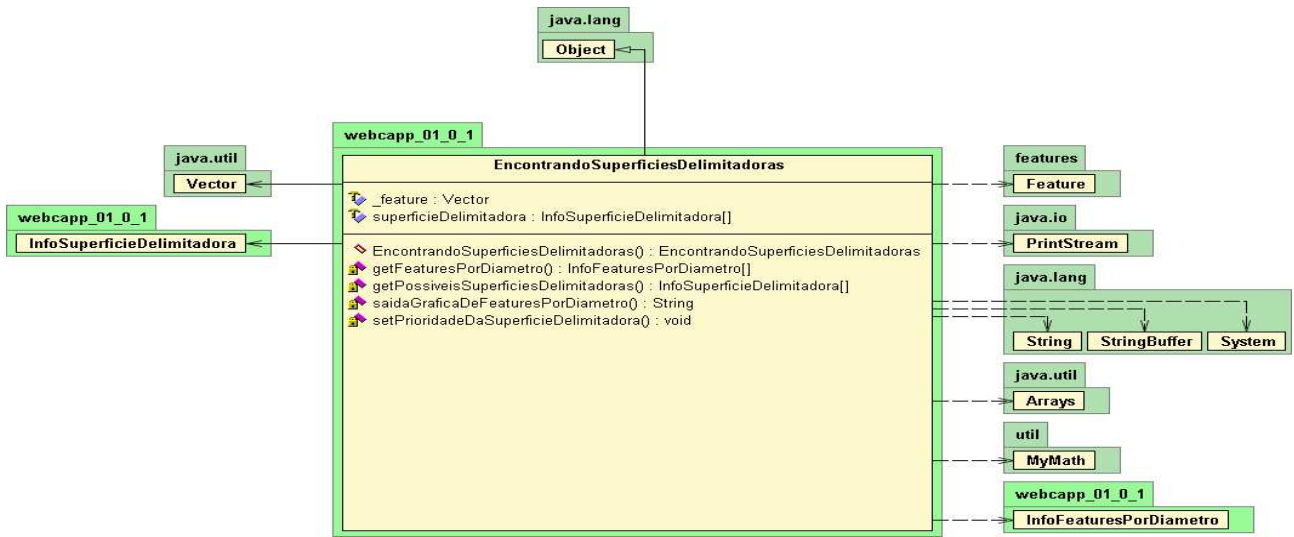


Figura D.18: Diagrama de classe *EncontrandoSuperficiesDelimitadoras* e seus relacionamentos com a classe *InfoDecomposicaoOrientadaAOoperacao*.

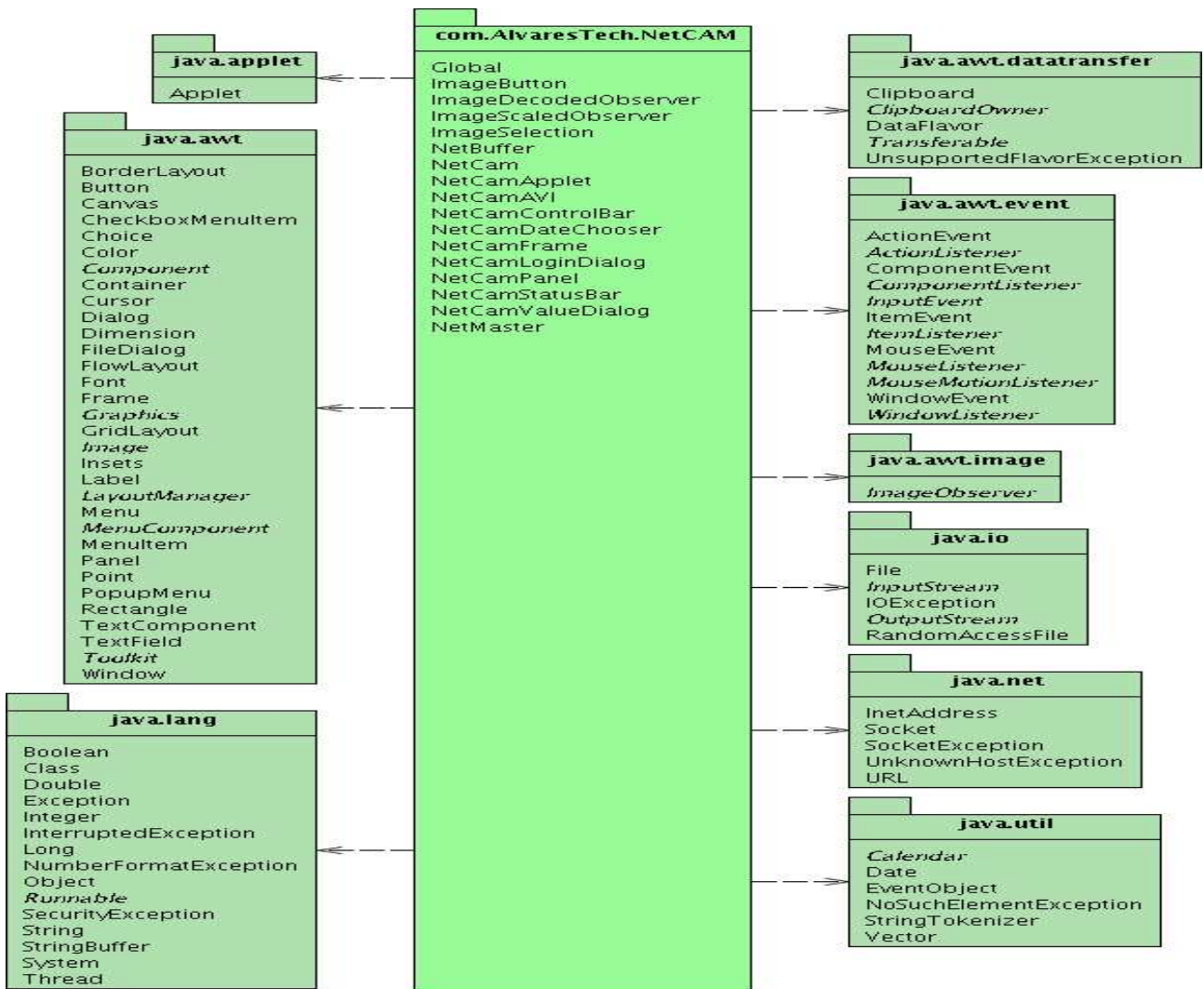


Figura D.19: Diagrama de classe principal (pacotes) associado ao módulo NetCam e relacionamento com demais pacotes, que permite a visualização de vídeo e imagens em tempo real e vídeo gravado do chão-de-fábrica (GUI NetCAM).

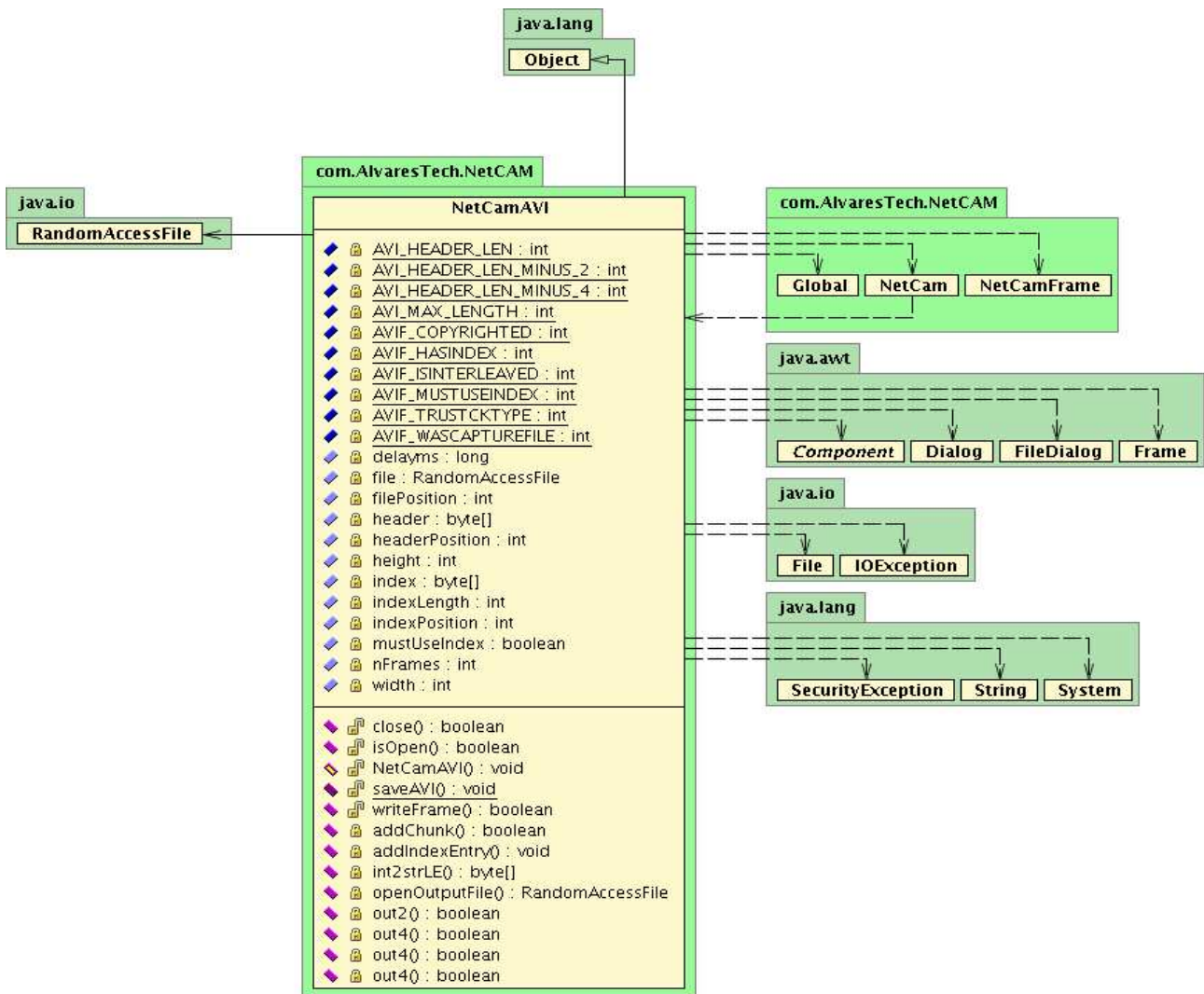


Figura D.20: Modelo estático (diagrama de classes) para a classe `NetCAMAVI` (pacote `com.AlvaresTech.NetCAM`) e seu relacionamento com as demais classes e pacotes do módulo Web-Cam, que permite a gravação de vídeo em formato `.avi`.

Referências Bibliográficas

- Adamczyk, Z. e Kociolek, K., 2001, CAD/CAM Technological Environment creation as an interactive application on the Web, *Journal of Materials Processing Technology*, Vol. 109, pp. 222-228.
- Adamczyk Z., Jonczyk D., Kociolek K., A new approach to a CAD/CAM system as a part of distributed environment, Intranet database, *Journal of Materials Processing Technology*, 5793, 1-6, 2002.
- Ahn, S., Sequin C. e Wright. P., 1999, Internet-Based Design an Manufacturing, Final Report 1988-99 for micro Project, 98-136, TRW.
- Ahn S., et al., Cybercut: An internet based CAD/CAM System, *ASME Journal of Computing and Information Science in Engineering*, Vol. 1, No. 1, 2001.
- Alting, L. et al., XPLAN - An Expert Process Planning System and its Further Development, *Matador 26th*, 1988, UMIST, 155-163.
- Alting, Leo; Zhang, Hong-Chao. Computer Aided Process Planning: the State-of-the-Art Survey. *International Journal of Production Research*, v.27, n.4, p.553-585, 1989.
- Álvares, A. J, 2003, "Uma Metodologia para Integração CAD/CAPP/CAM Voltada para Manufatura Remota de Peças Rotacionais Baseada na Internet", Qualificação para Tese de Doutorado, Programa de Pós-Graduação em Engenharia Mecânica, UFSC, <http://WebMachining.AlvaresTech.com>
- Álvares, A. J., A Tecnologia de Medição por Coordenadas como Base para Regulação da Qualidade Geométrica do Processo de Usinagem CNC, Dissertação de Mestrado, UFSC, maio, 1990.
- Álvares, A. J, Ferreira, J. C. E., Metodologia para implantação de Laboratórios Remotos Via Internet na área de Automação da Manufatura, II COBEF, Uberlândia, MG, 05/2003.
- Álvares, A. J. e Romariz, L. J., 2002, Telerobotics: Methodology for the Development of a Through-the-Internet Robotic Teleoperated System, *Journal of the Brazilian Society of Mechanical Sciences*, Vol. XXIV, No. 2, pp. 122-126.
- Álvares, A. J, 2002, Monografia da Disciplina de Estudo Dirigido: TeleManufatura Aplicada a Operações de Usinagem, Programa de Pós-Graduação em Engenharia Mecânica, UFSC, <http://WebMachining.AlvaresTech.com/telemanufatura>.

- Álvares, A. J, Ferreira, J. C. E., Shimabukuro, L. e Fernandes, C. A., 2002, Um sistema de telemanufatura baseado na web orientado ao processo de oxicorte, XXII ENEGEP, Curitiba.
- Álvares, A. J, 2001, Monografia da Disciplina de Estudo Dirigido: Métodos para o Projeto, Planejamento do Processo e Fabricação de Peças Assistidos por Computador, Programa de Pós-Graduação em Engenharia Mecânica, UFSC, <http://WebMachining.AlvaresTech.com/capp>.
- Álvares, A. J, Ferreira, J. C. E, Uma Nova Abordagem de CAPP Para Peças Rotacionais Baseada em Mapeamento de Features In: 3o Congresso Nacional de Engenharia Mecânica (CONEM), 2004, Belém, Para.
- Álvares, A. J. . E-Work y E-Mfg: La Tecnología de Información Aplicada En La Automatización de Manufactura. In: I Congreso de Ingeniería Industrial e Ingeniería en Mantenimiento Mecánico, CIIMAN 2004, 2004, Merida, Venezuela.
- Álvares, A. J. ; Soares, G. F. ; Araújo, F. H. P. . WebCAPP:IMPLEMENTAÇÃO DE UM CAPP VARIANTE BASEADO NA CODIFICAÇÃO OPTIZ. In: II Congresso Nacional de Engenharia Mecânica, CONEM 2002, 2002.
- Álvares, A. J. ; Ferrira, J. C. E. . WebMachining: Implementation of a Collaborative CAD/CAPP/CAM System for E-Manufacturing Through the Internet. In: The 38th CIRP - International Seminar on Manufacturing Systems, 2005, Florianópolis-SC.
- Álvares, A. J, Ferreira, J. C. E, CAD/CAPP/CAM Integration Methodology for the Remote Manufacture of Cylindrical Parts Through the Web In: 17o Congresso Brasileiro de Engenharia Mecânica (COBEM), 2003.
- Álvares, A. J., Cano, C. V., Teixeira, E. L. S. Modeling and Implementation of a Flexible Manufacturing Cell (FMC). In: 18th International Congress of Mechanical Engineering, 2005, Ouro Preto.
- Arezoo, B., Ridgway, K., Al-Ahmari, A.M.A., Selection of cutting tools and conditions of machining operations using an expert system, *Computers in Industry*, 42, pp. 43-58, 2000.
- Arnett, M.F., Dulaney, E. e Harper, E. 1994. *Inside TCP/IP*. New Riders Publishing. Indianapolis. USA.
- ASA - American Standard - B 5.22, Single Point Tools and Tools Posts. New York 18, ASME, 1950.
- Avila H. M., Weber S. F., A specification of the domain of process planning: Properties, Problems and Solutions, Report University of Kaiserslautern, 2000.
- Baker, A. D.; Parunak, H.V.D. ; Erol, K. - Agents and the Internet: Infrastructure for Mass Customization. *IEEE Internet Computing*, Vol. 3, No. 5, September-October 1999, pp. 62-69.9.
- Banerjee, P.; Zetu, D. (2001). *Virtual manufacturing*. John Wiley Sons, Inc, New York. 320p.

- Bastos, Ricardo M.; Oliveira, José P. - A Conceptual Modeling Framework for Multi-agent Information System. ER2000 Conference, LNCS (Lectures Notes in Computer Science) vol. 1920, pp.295-308, 2000.
- Baykasoglu, A., Dereli T., Novel algorithmic approach to generate the 'number of passes' and 'depth of cuts' for the optimization routines of multipass machining, *International Journal of Production Research*, Taylor Francis, Volume 40, Number 7/May 01, 2002.
- Bhaskara, R.S.V., Shunmugam, M.S., Narendran, T.T., Optimal sub-division of the depth of cut to achieve minimum production cost in multi-pass turning using a genetic algorithm, *Journal of Material Processing Technology*, 79, 1998, pp. 101-108.
- Bidarra, R., Van den Berg, E. e Bronsvort, W. F., Collaborative Modeling with Features, 2001, Proceedings of DET'01, 2001 ASME Design Engineering Technical Conferences, Pittsburgh, USA.
- Boogert R., Tool management in computer aided process planning, PhD thesis, University of Twente, 1994.
- Bronsvort, Willem F.; Jansen, Frederik W. Multi-View Feature Modelling for Design and Assembly. In: SHAH, Jami J.; MÄNTYLÄ, Martti; NAU, Dana S. *Advances in Feature Based Manufacturing*. Amsterdam : ELSEVIER, 1994. p.107-128.
- Brown, S. M. e Wright, P. K., 1998, A Progress Report on the Manufacturing Analysis Service, an Internet-Based Reference Tool, *Journal of Manufacturing Systems*, Vol. 17, No. 5, pp.389-401.
- Brückner, S, Wyns, J, Peeters, P, Kollingbaum, M. - Designing Agents for the Manufacturing Process Control. In Proceedings of Artificial Intelligence and Manufacturing Research Planning Workshop - State of the Art State of the Practice, AAAI Press, Albuquerque, New Mexico, pp. 40-46, 1998.
- Burdea, G. Coiffet, P. - *Virtual Reality Technology*, John Wiley Sons, New York, NY, 1994.
- Buxton, W. Telepresence: integrating shared task and person spaces. Proceedings of Graphics Interface' 92, pp. 123-129, 1992.
- Camarinha-Matos-Matos, L.M.; Afsarmanesh, H. *Balanced Automation Systems: Architectures and Design Methods (BASYS-95)*. Vitória : CHAPMAN HALL, 1995. p.159-170.
- Camarinha-Matos, L.M., Barata, J., Boissier, R., Leitão P., et al., *Integrated and Distributed Manufacturing, a Multi-Agent Perspective*, Proceedings of 3rd Workshop on European Scientific and Industrial Collaboration, Enschede, Netherlands, 27-29 June, 2001, <http://www.ipb.pt/~pleitao/papers/wesic2001.pdf>.
- CAM-I's illustrated glossary of workpiece form features, R-80-PPP-02.1, 1981.
- CAM-I report R-86-PPP-01, 1986.
- CAM-I, Deere Company, *Part Features for Process Planning*, Moline Illinois, CAM-I, 1986.

- Cay, F. e Chassapis, C., 1997, An IT view on perspectives of computer aided process planning research, *Computers in Industry*, 34, pp. 307-337.
- Chan FL, Spiller MD, Newton AR: WELD An Environment for Web-Based Electronic Design , Proceedings of the 1998 ACM Design in Automation Conference, June 1998, San Fransisco, pp. 146-151.
- Chan S, Wong M, Ng V: Collaborative Solid Modelling on the WWW , Proceedings of the 1999 ACM Symposium on Applied Computing, March 1999, San Antonio, pp. 598-602.
- Chang, T.C., Wysk R.A. e Wang, H.P., 1998, *Computer Aided Manufacturing*, Prentice Hall International Series in Industrial and Systems Engineering, W.J. Fabrycky e J.H. Mize (eds.), 2ns Edition, 1998.
- Chang T.C., Anderson D.C., Mitchell O.R., QTC - An integrated design/manufacturing/inspection system for prismatic parts, *Computers in Engineering Conference, CIE '88*, 1988, 417 - 426.
- Chang T-C., *Expert process planning for manufacturing*, Addison Wesley, Reading, 1990.
- Cheah, R. S., Lee, B. S. e Lim, R. L., Design and implementation of an MMS environment on ISODE, *Computer Communications*, 20, 1997, 1354-1364.
- Chen, M.C., Chen K.Y., Optimization of multipass turning operations with genetic algorithms: a note, *International Journal of Production Research*, Volume 41, Number 14/20, September, 2003.
- Chen, M.-C., Su, C.-T, 1999, Computer-aided optimization of multi-pass turning operations for continuous forms on CNC lathes, *IIE Transactions*, Jul 1999, 31, pp. 583-596.
- Chen, M.-C. and Su, C.-T., 1998, Optimization of machining conditions for Turning Cylindrical Stocks into Continuous Finished Profiles, *International Journal of Production Research*, Vol. 36, No. 8, pp. 2115-2130, 1998.
- Chen, M.-C. and Tsai, D.-M., 1996, A Simulated Annealing Approach for Optimization of Multi-pass Turning Operations, *International Journal of Production Research*, Vol. 34, No. 10, pp. 2803-2825, 1996.
- Chep, A., Tricarico, L. Bourdet, P., Galantucci, L., Desiggn of Object-oriented database for the definition of Machining Operation Sequences of 3D Workpieces, *Computers In. Engng*, Vol. 34, No. 2, 257-279, 1998.
- Cherng J. G., Shao X., Chen Y., Sferro P. R., Feature-based part modeling and process planning for rapid response manufacturing, *Computers ind. Engng.*, Vol. 34, No. 2, 515-530, 1998.
- Cho, H.; Derebail, A.; Hale, T. et al. A Formal Approach to Integrating Computer-Aided Process Planning and Shop Floor Control. *ASME Journal of Engineering for Industry*, v.116, p.108-116, 1994.
- Cho, K. K., Lee, S. H., Ahn, J. H., 1991, Development of Integrated Process Planning and Monitoring System for Turning Operation, *Anais CIRP*, Vol. 40, pp. 423-427.

- Chung, C., Peng, Q., The selection of tools and machines on web-based manufacturing environments, *International Journal of Machine Tools and Manufacture*, Volume 44, Issues 2-3, February 2004, Pages 317-326.
- Corney J. e Lim T., 2001, *3D Modeling with ACIS*, Saxe-Coburg Publications, 2a. Edição, 2001.7 - 426.
- Cutkosky M.R., Tenenbaum J.M., Muller D., Features in process based design, *ASME Computers in Engineering (CIE) Conference*, San Francisco, 1988, 557-562.
- Dalton, B., *Techniques for Web Telerobotics*, Thesis of Doctor of Philosophy Department of Mechanical and Materials Engineering University of Western Australia Perth, Australia, 2001.
- Davies, B.J., *The Integration of Process Planning with CAD/CAM including The use of Expert System*, *Int Conference of Comp. Aided Production Engineering*, 1986.
- Deb S., Ghosh K., Deb S. R., *Machining process planning for rotational components by using a neural network approach*, *Proceedings of th 29 International Conference on Computers and Industrial Engineering*, pp 117-122, Montreal, Canada, november, 2001.
- Deo, N., *Graph Theory with Applications to Engineering and Computer Science*, Prentice-Hall, Inc., 1974.
- Dépincé, P., Amara H., Hascoët, J. Y., *Human Integration in s distributed CAPP framework*, *Anais do International IFIP Conference on Feature Modeling and Advanced Design-For-The-Life-Cycle Systems*, Valenciennes, França, 12-14 de Junho, 2001.
- Detand J., *A computer aided process planning system generating non-linear process plans*, *Catholic University of Leuven, Belgium, PhD thesis*, 1993.
- Derebail, A., Cho, H., Wysk, R., *Process plan representation for shop floor control*, *Capítulo 14 do livro Process planning and process plan representation for control*. In S. B. Joshi and J. S. Smith (ed), *Computer Control of Flexible Manufacturing Systems* (New York: Chapman Hall), pp 379-404., 1994.
- Dereli, T., Filiz, I.H., Baykasoglu, A., *Optimizing cutting parameters in process planning for prismatic parts by using genetic algorithms*, *Int. J. Prod. Res.*, Vol 39, No. 15, 3303-3328, 2001.
- Desai, V. S. e Pande, S. S., 1991, *GFM - an interactive feature modeller for CAPP of rotational components*, *Computer Aided Engineering Journal*, October 1991, pp. 217-220.
- Dhage, S., Usher, J. M., *Computer-Aided tool selection for turning and boring*, *Computers and Industrial Engineering*, Vol 25, No. 1-4, pp. 207-210, 1993.
- Dietrich U, Von Lukas U, Morche I: *Cooperative modeling with TOBACO* , *Proceedings of the Team-CAD97 Workshop on Collaborative Design*, May 1997, Atlanta, pp. 115-122.
- Domazet D., Maniac M., *CADROT A product modeller and CAD module for integrated CAD/CAPP/CAM System for rotational parts*, *Proc. Matador 28th, UMIST*, 1990.

- Dong J., Vijayan S., Features extraction with the consideration of manufacturing process, *Int. J. Prod. res.*, 1997, Vol. 35, No. 8, pp. 2135-2155.
- Erickson, R. E., 1988, The state of the art in computer aided process planning, CAM-I report.
- Erve A.H. van 't, Computer Aided Process Planning for Part Manufacturing, an expert system approach, PhD thesis, University of Twente, 1988.
- Estrem, W. A., An evaluation framework for deploying Web Services in the next generation manufacturing enterprise, *Robotics and Computer-Integrated Manufacturing*, Volume 19, Issue 6, December 2003, Pages 509-519.
- Evershein W. et al., Application of Automatic Process Planning and NC Programming, The CASA/SME AUTOFACT Conference, west Germany, 1980.
- Ferreira, J.C.E., 1996, Planejamento do Processo Assistido por Computador - CAPP, Apostila, Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, Florianópolis, 1996.
- Ferreira J. C. E., Wysk, R. A., An Investigation of the Influence of Alternative Process Plans on Equipment Control, *Journal of Manufacturing Systems*, Vol.19/N.6, 2001, 393-406.
- Fanuc, FOCAS1 : FANUC Open CNC API Specifications version 1 - FOCAS1/Ethernet CNC/PMC Data window library, <http://www.graco.unb.br/alvares/romi/Focas1/Disk2/Doc/FWLIB32.HTM>, 2003.
- Franklin S., Graesser A. - Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents - Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages - Springer-Verlag, 1996.
- Fuh, J.Y.H., Li. W. D., Advances in collaborative CAD: the-state-of-the art, *Computer-Aided Design* xx (2004) 1-11.
- Gadh R., Flexible mappings as utilized within a flexible features-based design approach, *proc. IFIP W.G. 5.3 Conf. on Feature modeling and recognition in advanced CAD/CAM systems, Valenciennes (F)*, Vol.1, 1994, 205-223.
- Gilbert, W. W., 1950. *Economics of Machining, Machining - Theory and Practice*, American Society for Metals, pp. 465 - 485.
- Giusti, F., Santochi M., Dini, G., 1989, KAPLAN: a Knowledge-Based Approach to Process Planning of Rotational Parts, *Annals CIRP*, Vol. 38, pp. 481-484.
- Groover, M. *Automation, Production System, and Computer Integrated Manufacturing*, 1987, Prentice Hall.
- Gruber, T. Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*. Vol. 43, 1995.

- Gupta, R., Batra, J.L., LAL, G.K., Determination of optimal subdivision of depth of cut in multipass turning with constraints, *Int. J. Prod. Res.* 1995, vol. 33, no.9, pp. 2555-2565.
- Halevi, G. e Weill, R.D., *Principles of Process Planning: A Logical Approach*, Chapman Hall, 1995.
- Halevi, G., *Restructuring the Manufacturing Process: Applying the Matrix Method*, St. Lucie Press, 1999.
- Hall, B. B. Beej's guide to network programming using internet sockets. Disponível na Internet via WWW. URL: <http://www.eest.esuchico.edu/~beej/guide/net> 1999.
- Ham, L. e Lu, S.C.Y., 1988, Computer-aided process planning: the present and the future, *Anais CIRP* 37 (2), 1988, pp. 591-601.
- Han, J., Application of ISO 14649 STEP-NC Data Model for CAPP/NC Integration, *These Industrial Engineering*, Pohang University of Science Technology, February 2001.
- Han, J. H. e Requicha, A. A. G., Modeler-independent Feature Recognition in a Distributed Environment. *Computer-Aided Design*, 30(6), 453-463, 1998.
- Hao, Qi, Shen, W., Wang, L., Towards a cooperative distributed manufacturing management framework, *Computers in Industry*, Volume 56, Issue 1, January 2005, Pages 71-84.
- Hardwick, M., Spooner, D. L., Rando, T, e Morrir, K. C., Sharing Manufavturing Information in Virtual Enterprises, *Communications of the ACM*, 39(2), 46-54, 1996.
- Harrington, J.Jr., 1984, *Understanding the Manufacturing Process - Key to Successful CAD/CAM Implementation*, Marcel Dekker, INC., New York abdd Basel, 1984.
- Hartley, J. R., *Concurrent Engineering - Shortning Lead Times, Raising Quality and Lowering Costs*, Productivity Press (1992), Chaper 2.
- Hashmi, K., Baradie, M. A., Ryan, M., Fuzzy Logic Based Intelligent Selection of Machining Parameters, *Computers in Industry*, 35, 1998, 571-574.
- Hinduja S., Kroeze B., Selection of Tools for finishing operations on turned components, Crestin J.P., McWaters, J.F., *Software for Discrete Manufacturing*, North Holland, 1986, 295-305.
- Hitachi Seiki, [<http://www.hitachiseikiusa.com/controls/index.html>.]
- Horstmann C. S. e Cornell G., *Core Java 2 - Volume 1 - Fundamentos*, Makron Books, 2000.
- Horstmann C. S. e Cornell G., *Core Java 2 - Volume 2 - Recursos Avançados*, Makron Books, 2000
- Houten F.J.A.M., Kals H.J.J., Round, a flexible technology based process and operations planning system for NC lathes, *Proceedings of the 16th CIRP International Seminar on Manufacturing Systems*, Tokyo, 1984.
- Houten F.J.A.M. van, *PART: a computer aided process planning system*, PhD thesis, University of Twente, Enschede, 1991.

- Huang, Hefeng. A Generative Process Planning System for Turned Components. Manchester, 1988. Tese (Doutorado em Fabricação Mecânica-Planejamento de Processos). Manufacturing and Machine Tool Division, Mechanical Engineering.
- Ibrahim, R. N., Kee, P. K., Shrivastava, Optimisation of tool path geometry for efficient machining, J. Mater. Process Technol., 4, p. 215-226, 1994.
- IEEE, Robots on the WEB - Global Remote Control Through Internet Teleoperation, IEEE Robotics Automation Magazine, Vol. 7, No.1, 2000.
- Irani, S.A., Koo. H.Y. e Raman, S., 1995, Feature-based operation sequence generation in CAPP, International Journal of Production Research 33 (1), 1995, pp. 17-39.
- ISO 1101, Technical drawings - Geometrical tolerancing - Tolerances of form, orientation, location and run-out - Generalities, definitions, symbols, indications on drawings, 1983.
- ISO 6983-1, Numerical control of machines program format and definition address words. Part 1. Data Format for positioning, line and contouring control system, First edition, 1982.
- ISO TC184/WG3 N324 -T7, ISO 10303 - Part 224 Mechanical Product Definition for Process Planning Using Form Features, South Carolina, EUA, 1994.
- ISO 10303-224, Industrial Automation Systems and Integration—Product Data Representation and Exchange. Part 224: Application Protocol: Machined Product Definition for Process Planning Using Machining Features, 1997.
- ISO 14649 Data model for Computerized Numerical Controllers - Part 1: Overview and fundamental principles, Final Draft International Standard, 2003.
- ISO 14649 Data model for Computerized Numerical Controllers - Part 10: General process data, Final Draft International Standard, 2003.
- ISO 14649 Data model for Computerized Numerical Controllers - Part 12: Process Data for Turning, Draft International Standard, V09 April 2003.
- Jennings, N.R. and Wooldridge, M.J. Applications of Intelligent Agents. Agent Technology: Foundations, Applications, and Markets. Jennings, N.R. and Wooldridge, M.J (Eds.), Springer, pp. 3-28, 1998.
- Jennings, N.R. - Agent Theories, Architecture and Languages: a survey. LNAI (Lectures Notes in Artificial Intelligent), Intelligent Agents, Jennings N.R., Wooldridge M. (Eds.), vol. 890, Springer, pp. 1-39, 1994.
- Junior, A. C. S., Uma Abordagem para Criação e Compartilhamento de dados de Peças Através da Integração CAD-RDBMS, Dissertação de Mestrado, UFSC, 2001.
- Júnior, F. F., Desenvolvimento de um sistema de monitoração e supervisão para o processo de torneamento, Dissertação de mestrado, Escola de Engenharia de São Carlos da USP, 2002.

- Juri, A.H., Reasoning about machining operations using feature based models, *Int. J. Prod. res.*, 1990, vol. 28, no1, 153-171.
- Khan, Z., Prasad, B., Singh, T., Machining Condition Optimization by Genetic Algorithms and Simulated Annealing, *Computers Ops Res.*, Vol24, No. 7, pp 647-657, 1997.
- Kao, Y. C. e Lin, G. C., 1996, CAD/CAM Collaboration and remote Machining, *Computer Integrated Manufacturing Systems*, Vol. 9, No. 3, pp. 149-160.
- Kayacan, M. C., Filiz, I. H., Sönmez, A. I., Baykasoglu, A. e Dereli, T., 1996, OPPS-ROT: An optimised process planning system for rotacional parts, *Computers in Industry*, 32, 181-195.
- Kee, P.K., Development of computer-aided machining optimisation for multi-pass rough turning operations, *Int. J. Economics*, 37, pp. 215-27, 1994.
- Kim CY, Kim N, Kim Y, Kang SH, O Grady P, Internet-based Concurrent Engineering: An Interactive 3D System with Markup , *Proceedings of the 1998 ASME Computers in Engineering Conference*, September 1998, Atlanta, CIE- 5522.
- Kiritsis, D., Porchet, M., A Generic Petri net model for dynamic process planning and sequence optimization, *Advances in Engineering Software*, Vol. 25, 1996, 61-71.
- Kruth, J. P. e Detand, J., 1992, A CAPP System for Nolinear Process Plans, *Anais CIRP Vol. 41/1/1992*, pp. 489-492.
- Kruth J. P., Detand J., Zeir G. V., Kempenaers J., Pinte J., Methods to improve the response time of a CAPP system that generates non-linear process plans, *Advances in Engineering Softwares*, 25, 9 - 17, 1996.
- Lan, H., Ding, Y., Hong, J., Huang, H., Lu, B., A web-based manufacturing service system for rapid product development, *Computers in Industry*, Volume 54, Issue 1, May 2004, Pages 51-67.
- Lee, J., E-manufacturing: fundamental, tools, and transformation, *Robotics and Computer-Integrated Manufacturing*, Volume 19, Issue 6, December 2003, Pages 501-507.
- Lee, J.W., A study on the determination of actual cutting time in NC turning, *Int. J. Prod. Res.*, Vol. 26, No. 9, pp. 1547-1559, 1988.
- Lee, Kok-Meng, Daley, W. and McKlin, T., 1998, An interactive learning tool for dynamic systems and control, presented at *International Mechanical Engineering Congress Exposition*, Anaheim, CA.
- Lee, I. B. H., Lim, B. S., Nee, A. Y. C., Knowledge-based process planning system for the manufacture of progressive dies, *Int. J. Prod. Res.*, 1993, Vol. 31, No. 2, 251-278.
- Lee, J. Y., Han, S. B., Kim, H., Park, S. B., 1999, Network-centric Feature-based Modeling, *Pacific Graphics 1999*.

- Lenderink A., The integration of process and production planning in small batch part manufacturing, PhD thesis, University of Twente, 1994.
- Li D, Muntz R, COCA: Collaborative Objects Coordination Architecture , Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work, pp. 179-188.
- Li, W.D., A Web-based service for distributed process planning optimization, Computers in Industry, Volume 56, Issue 3, April 2005, Pages 272-288.
- Li WD, Ong SK, Fuh JYH, Wong YS, Lu YQ, Nee AYC., Featurebased design in a collaborative and distributed environment, Comput-Aided Des, 2004;36(9):775-97.
- Li, W.D., Ong, S.K., Nee, A.Y.C., A Web-based process planning optimization system for distributed design, Computer-Aided Design, Volume 37, Issue 9, August 2005, Pages 921-930.
- Li Y., Shao X., Li P., Liu Q., Design and implementation of a process-oriented intelligent collaborative product design system, Computers in Industry, Volume 53, Issue 2, February 2004, Pages 205-229.
- Lu, W. F., Liu, X. F., Chang, H. C. e Dong, L., 1997, Feature-based Representation and Indexing of Axisymmetric Parts in Machining Process Planning using Case-Based Reasoning, 1082-3409/97, IEEE, pp. 406-413.
- Malek, L. A., Wolf, C. e Guyot, P. D., 1998, Telemanufacturing: A Flexible Manufacturing Solution, Int. J. Production Economics, Vol. 56-57, pp. 1-12.
- Manet, Desenvolvimento de um Sistema Inteligente de Produção com Gestão Remota Via Internet, Unicamp-USP-UFSC-UNESP-UnB, Projeto de Pesquisa submetido ao CNPq, 2002.
- Manufacturing on the Internet: Use 21st Century Techniques to Speed Your Product Cycles, John Wiley Sons, 2a. Edição, 1997.
- Massaropi. E., Masiero, P., Modelagem geométrica e funcional de placas de torno para auxiliar a determinação do fluxo de peças em sistemas produtivos, Anais Congresso Brasileiro de Engenharia de Fabricação - 1 Cobef, Curitiba, 2001.
- Martino, T. D., Falcidieno, B. e Hasinger, S., Design and Engineering Process Integration Through a Multiple View Intermediate Modeller in a Distributed Object-oriented System Environment, Computer-Aided Design, 30(6), 437-452, 1998.
- Mayer R. J., Su C. J. , Sun T., et al., ECTOF: a Feature Representation Tecnique for Concurrent Engineering Applications, Journal of Design and Manufacturing, vo. 4, no. 1, 1994, 49-65.
- Maziero, N. L., Nascimento, V., Geração de trajetórias de usinagem para torneamento de desbaste e acabamento num sistema baseado em features, Anais Congresso Brasileiro de Engenharia de Fabricação - 2 Cobef, Uberlandia, 2003.
- McIntosh, R. L., Open-source tools for distributed device control within a Service-Oriented Architecture, Journal of the Association for Laboratory Automation, vo. 9, no. 6, 2004, 404-410.

- Mesquita, N. G. M., Andrade, S. M. V., Oliveira, J. A. C., Carvalho, H. M. B., Banco de Dados de Ferramentas de Corte Para um Sistema CAD/CAPP/CAM, Revista Máquinas e Metais, Ano XXXIX, N. 442, Nov., 2002, 128 - 139.
- Metcalf, V. Gierth, A. et al. Programming UNIX Sockets in C - Frequently Asked Questions. Disponível na Internet via WWW. URL: <http://www.ibrado.com/sock-faq/> 1998.
- Mills A., Collaborative Engineering and the Internet: Linking Product Development Partners Via the Web, SME, 2001.
- Multi-Agent Systems Home Page www.multiagent.com
- Multi-Agent Research Group, <http://www.cs.wustl.edu/~mas/main.html>
- Multi-Agent Systems Laboratory <http://mas.cs.umass.edu/index.shtml>.
- Nam T.J, COLLIDE, A shared 3D workspace for CAD, Proceedings of the 1998 Conference on Network Entities, Leeds.
- Nam TJ : An Investigation of Multi-user Design Tools for Collaborative 3D Modelling , 1998 Conference on Computer Supported Cooperative Work, Doctoral Colloquium, November 1998, Seattle.
- Narang R. V., Fischer, G.W., Development of a Framework to Automate Process Planning Functions and to Determine machining Parameters, International Journal of Production Research, Vol. 31, N.8, 1921-1942, 1993.
- Nof, S. Y. et al., Handbook of Industrial Robotics, John Wiley Sons, 2 ed, Canada, 1999.
- Nof, S. Y., Collaborative E-Work And E-Mfg.: The State Of The Art And Challenges For Production And Logistics Managers, International Federation of Automatic Control 11th IFAC Symposium on Information Control Problems in Manufacturing (INCOM), Salvador, Brasil April 5-7th, 2004.
- Offodile O. F., Malek L. L. A., The virtual manufacturing paradigm: The impact of IT/IS outsourcing on manufacturing strategy, In. J. Production Economics, 75, 147 - 159, 2002.
- Onwubolu, G.C., Kumalo, C., Optimization of multipass turning operations with genetic algorithms, International Journal of Production Research, Taylor Francis, Volume 39, Number 16/November 10, 2001.
- Onwubolu, G.C., Kumalo, C., Multi-pass turning operations optimization based on genetic algorithms, Proceedings of the Institution of Mechanical Engineers, 215,1, pp. 117-124, 2001.
- Peng, Q., Hall, F. R. e Lister, P. M., Application and evaluation of VR-based CAPP system, Journal of Materials Processing Technology, 107, 2000, 153-159.
- Phillips R.H. et al., MICROPLAN: A microcomputer based expert system for generative process planning, Knowledge based expert system for manufacturing, 1986 ASME, Vol. 24.

- Phillips R.H., Mouleeswaran C.B., A knowledge-based approach to generative process planning, AUTOFACT 85, 1985, Detroit-USA.
- Pradhan, S. S. e Huang, W. V., 1998, Virtual Manufacturing Information System Using Java and JDBC, Computers Ind. Engng, Vol. 35, No. 1-2, pp. 255-258.
- Prakash A, Shim HS: DistView: Support for Building Efficient Collaborative Applications using Replicated Objects, Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, October 1994.
- Prasad, A. V. S. R. K., Rao, P.N., Rao R.K., Optimal selection of process parameters for turning operations in a CAPP system, Int. J. Prod. Res., Taylor Francis, Volume 35, Number 6/June 1, 1997.
- Quatrani, T., Modelagem Visual com Rational Rose 2000 e UML, Rio de Janeiro, Ciência Moderna, 2001, 206 p.
- Reddy S.V.B., Shunmugam, M.S., Narendran, T.T., Optimal sub-division of the depth of cut to achieve minimum production cost in multi-pass turning using genetic algorithm, Journal of Materials Processing Technology, 79, pp. 101-108, 1998.
- Rezende, D. F., Planejamento de Processo de Fabricação Assistido por Computador Através de um Sistema Especialista Baseado na Tecnologia de Features: Um Modelo de Desenvolvimento Voltado Para a Realidade Industrial, Dissertação de Mestrado, UFSC, 1996.
- Rho, H.M., Greelink, R., Van't Erve, A.H., Kals, H. J.J., 1992, An integrated cutting tool selection and operation sequencing method, Anais CIRP 41 (1), 1992, pp. 517-520.
- Rico, C. S., Mateos, S., Cuesta, E., Duarte, A., 1997, An Automatic CAPP System for Rotational Parts, 0-7803-4192-9/97, IEEE, 1997.
- Rico, C. S., Mateos, S., Cuesta, E., Duarte, A., 2000, Automatic selection of clamping surfaces in the turning process for rotational parts, Proc Instn Mech Engrs, Vol 214, Part B, pp. 693-708.
- Rockwell Automation, 2002, Making Sense of e-Manufacturing: A Roadmap for Manufacturers, Industry White Paper.
- Rocha, J., Ramos, C., Vale, Z., Process Planning using a genetic algorithm approach, Proceedings of the 1999 IEEE International Symposium on Assembly and task Planning, Porto, Portugal, July 1999.
- Rodrigues, K., Al-Ashaab, A., Knowledge web-based system architecture for collaborative product development, Computers in Industry, Volume 56, Issue 1, January 2005, Pages 125-140.
- Salomons, O.W. Computer Support in the Design of Mechanical Products. Twente, 1995. Tese (Doutorado em Projeto Mecânico-Ferramentas Computacionais de Suporte). University of Twente.

- Salomons O.W., Houten F.J.A.M. van, Kals H.J.J., Review of research in feature-based design, *Journal of Manufacturing Systems*, Vol.12, no.2, 1993, 113-132.
- Salomons O.W., Slooten F. van, Houten F.J.A.M. van, Observations from redesign and process planning practice, a study of human practice aimed at improving feature based CAD/CAPP, proceedings of the Twelfth International Conference on CAD/CAM, Computer Graphics and Computer Aided Technologies (MICAD'93), Vol. 1, Editions Hermes, Paris, 9-12 February, 1993, pp 115-129. (Also: internal report PT 452).
- Salomons O.W., Kappert J.H., Slooten F. van, Houten F.J.A.M. van, Kals H.J.J., Computer support in the (re)design of mechanical products, a new approach in feature based design, focusing on the link with CAPP, IFIP Transactions B-11, Knowledge Based Hybrid Systems, Mezger I., Bertek P. (eds.), Elsevier Science Publishers B.V. (North Holland), 1993, pp 91-103.
- Salomons O.W., Slooten F. van, Koning G.W.F. de, Houten F.J.A.M. van, Kals H.J.J., Conceptual graphs in CAD, *CIRP annals*, v. 43/1, August 1994, 125-128.
- Sandvik Coromant. Herramientas de Tornear: Productos para el Mecanizado del Metal, Catálogo de Productos. Dinamarca, 1993.
- Sandvik Coromant. Modern Metal Cutting - A Practical Handbook, 1. edição, Dinamarca, 1993.
- Scientific American Frontiers, 2000, 21st Century Medicine: Cybersurgery , http://www.pbs.org/safarchive/4_class/45_pguides/pguide_605/4565_cyber.html
- Schmid, C. 1998, The Virtual Lab VCLAB for Education on the Web, Proc. Of ACC , Philadelphia, págs. 1314-1318, <http://www.esr.ruhr-uni-bochum.de/VCLab>.
- Schutzer, K.; Schulz, H.; Weidmann, A., Design by Features: Feature Based Design Prismatic Parts. In: Proceedings of the Workshop on Flexibility through Integrated Design, Process Planning and Scheduling, CETIM, Selins, 1992.
- Schutzer, K.; Schulz, H.; Weidmann, A., Design by Features Based Design of Rotational Parts. In: Proceedings of the Workshop on Flexibility through Integrated Design, Process Planning and Scheduling, CETIM, Senlis, 1992.
- Shah, J. J., Mäntylä, M., Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications, John Wiley Sons Inc, New York, 1995.
- Shah, J. J., Dedhia H., Pherwani V. e Solkhan S., Dynamic Intergacing of Appllications to Geometric Modeling Services Via Modelar Neutral Protocol, *Computer-Aided Design*, 29(12), 811-824, 1997.
- Shah J.J., Rogers M.T., Functional requirements and conceptual design of the feature-based modelling system, *Computer Aided Engineering Journal*, Febr. 1988, 9-15.
- Shah J.J., Bhatnagar A.S., Group technology classification from feature-based geometric models, *Manufacturing Review*, Vol.2, No.3, September 1989, 204-213.

- Shah J.J., Philosophical development of form feature concept, CAM-I report P-90-PM-02 , 1990, 55-70.
- Shah, Jami J., Rogers, Marty T. and Urban, Susan D., Engineering data management: achieving integration through database technology, *Computing and Control Engineering Journal*, june 1993, pp. 119-126.
- Shah, J.J. e Mäntylä, M., *Advances in Feature Based Manufacturing*, Elsevier, 1994.
- Shah J.J., A heterogeneous, active database architecture for engineering data management, *Int. J. Computer Integrated Manufacturing*, Vol.7, No.5, 1994, 276-293.
- Shen, W. e Norrie, D.H. *Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey*. *Knowledge and Information Systems, an International Journal*, 1(2), 129-156, 1999.
- Shen, W., Norrie D.N., Barthés J.P.A, 2001, *Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing*, Taylor Francis, New York.
- Sheridan, T. B. *Telerobotics, Automation and Human Supervisory Control*. MIT Press, 1992.
- Shin, Y.C. e Joo, Y. S., Optimization of machining conditions with practical constraints, *International Journal Production Research*, Vol. 30, pp. 2907-2919, 1992.
- Shunmugam, M. S., Mahesh, P. e Reddy, S. V. B., 2002, A method of preliminary planning for rotational components with C-axis features using genetic algorithm, *Computers in Industry*, 1605, pp. 1-19.
- Shyu, V. P.J.M., Chen, Y. W. e Wang, K.K., 1987, A Mini CIM System for Turning, *Anais CIRP*, Vol. 35, pp. 277-280.
- Suh, S.H., Cheon, S.U., A Framework for an Intelligent CNC and Data Model, *Int. J. Adv. Manuf. Technol.*, 19, pp. 727-735, 2002.
- Suh, S.H., Lee, B. E., Chung, D.H., Cheon, S.U., Architecture and implementation of a shop-floor programming system for STEP-compliant CNC, *Computer-Aided Design*, 35, pp. 1069-1083, 2003.
- Suh, S.H., Lee, B. E., Chung, D.H., Cheon, S.U., et al., Developing an Integrated STEP-Compliant CNC Prototype, *Journal of Manufacturing Systems*, Vol. 21, No. 5, pp. 350-362, 2002.
- Suh, S.H., Cho, J.H., Hong, H.D., On the architecture of intelligent STEP-compliant CNC, *Int. J. Computer Manufacturing*, Vol. 15, No.2, pp. 168-177, 2002.
- Silva, A. D., *Uma Metodologia para Otimização Automática de Parâmetros de Usinagem*. Florianópolis, 1994. Tese (Doutorado em Fabricação Mecânica-Planejamento de Processos). Universidade Federal de Santa Catarina.
- Smart J : wxWindows Technical Notes , <http://www.wxwin.org>.

- Smith, C. S., The Manufacturing Advisory Service: Web Based Process and Material Selection, Tese de Doutorado, Mechanical Engineering, University of California, Berkeley, 1999.
- Smith, C. S., Wright, P. K., Cybercut: A World Wide Web Based Design to Fabrication Tool, 2000, <http://cybercut.berkeley.edu>.
- Spatial Technology, ACIS Geometric modeler, Boulder, Colorado, 1994.
- STEP, ISO TC184/SC4/WG5, Express language reference manual (ISO 10303, part 11), release draft, Spiby P. (ed.) , 1991.
- STEP, ISO TC184/SC4/WG5, Express-I language reference manual (ISO/WD 10303-12), working project draft, Wilson P. (ed.), 1992.
- STEP, ISO TC184/SC4/WG5, Industrial automation systems- product data representation and exchange, (ISO/WD 10303-part 48: Form features), 1992.
- Stork A, Jasnoch U: A Collaborative Engineering Environment , Proceedings of the TeamCAD97 Workshop on Collaborative Design, May 1997, Atlanta, pp. 25-33.
- Stork A, Von Lukas U, Schultz R: Enhancing a commercial 3D CAD system by CSCW functionality for enabling co-operative modelling via WAN , Proceedings of the ASME Design Engineering Technical Conferences, September 1998, Atlanta, CIE-5711.
- Su, D., Li, J., Xiong, Y., Zheng, Y., Application of Internet techniques into online collaborative design and manufacture, Computer Supported Cooperative Work in Design, 2005. Proceedings of the Ninth International Conference on Volume 2, 24-26 May 2005 Page(s):655 - 660 Vol. 2.
- Sundaram, R. M., 1986, Process Planning and Machining Sequence, Computers Industrial Engineering, Vol. 11, 1986, 27-31.
- Tan, F.P., Creese, R.C., A generalized multi-pass machining model for machining parameter selection in turning, Int. J. Prod. Rese, Vol. 33, No. 5, 1467-1487, 1995.
- Taylor, K. Trevelyan, J., 1995, A Telerobot on the World Wide Web, National Conference of the Australian Robot Association, Melbourne, July 5-7, <http://telerobot.mech.uwa.edu.au>.
- Tönshoff, H.K., Aurich, J.C., Baum, Th. Configurable Feature-Based CAD/CAPP System. Proceedings of the IFIP International Conference on Feature Modeling and Recognition in Advanced CAD/CAM Systems. Valenciennes, France, p.757-769, 1994.
- Tornincasa, S., Web3D Technology applications for distance training and learning: the Leonardo project WEBD, XII ADM International Conference - Grand Hotel - Rimini Italy - Sept. 5 5th-7th, 2001.
- Ulieru M., Norrie D., Kremer R., Shen W., A Multi-resolution collaborative architecture for web-centric global manufacturing, Information Sciences, 127, 3 - 21, 2000.

- Usher J. M., Bowden R. O., The application of Genetic Algorithms to operation sequencing for use in computer-aided process planning, *Computers ind. Engng*, Vol. 30, No. 4, 999-1013, 1996.
- Van't A.H., Generative Computer Aided Process Planning for Part Manufacturing - An Expert System Approach, PhD thesis, University of Twente, Enschede, 1985.
- Varvakis, G. J. R., Manufacturing Code Generation for Rotacional Parts in a Feature Based Product Modelling Environment, Tese de Doutorado, Loughborough University of Technology, 1991.
- Vin L.J. de, Vries J. de, Streppel A.H., Kals H.J.J., PART-S, a CAPP system for small batch manufacturing of Sheet metal components, *Manufacturing Systems*, (proceedings of the CIRP Seminars), Vol. 22, No. 2, 1993, 133 - 141.
- Wang H.P., Lin C.A., Automated Generation of NC Part Programs for Turned Parts based on 2-D drawing files, *The Int. J. Advanced Manufacturing Technology*, 1987, vol2, No.3, 23-35.
- Wang, F. C. e Wright, P. K., Web-Based CAD Tools for a Networked Manufacturing Service. Proceedings of DET'98 ASME Design Engineering Technical Conference, Atlanta, Georgia, CIE-5517, 1998.
- Wang M-T., An object-oriented feature-based CAD-CAPP-CAM integration framework, *Advances in Design Automation*, vol.1, ASME 1991, DE Vol.32-1, 109-116.
- Wang, M. e Walker, H., Creation of a intelligent process planning system with the relational DBMS software environment, *Computers in Industry*, vol. 13, 215-228, 1989.
- Wang, H.P. e Li, J.K., *Computer-Aided Process Planning*, *Advances in Industrial Engineering*, Vol. 13, Elsevier, 1991.
- Wang, H.P., Wisk, R.A., Intelligent Reasoning of Process Planning, *Computers in Industry*, 1987, No 8, 293-309.
- Wang, L., Orban, P., Cunningham, A., Lang, S., Remote real-time CNC machining for web-based manufacturing, *Robotics and Computer-Integrated Manufacturing*, Volume 20, Issue 6, December 2004, Pages 563-571.
- Wang, X., Da, Z.J., Balaji, A.K., et al., Performance-based optimal selection of cutting conditions and cutting tools in multipass turning operations using genetic algorithms, *International Journal of Production Research*, Taylor Francis, Volume 40, Number 9/June 01, 2002.
- Wang, X., Jawahir, I.S., Optimization of Multi-pass Turning Operations Using Genetic Algorithms for the Selection of Cutting Conditions and Cutting Tools with Tool-wear Effect, *IEEE (0-7803-7078-3/01)*, pp. 3093-3100, 2001.
- Waterman, Donald A. *A Guide to Expert Systems*. 1. ed. Massachusetts : Addison-Wesley, 1986.
- Weinman, W., 1997, *Manual de CGI*, Makron Books, São Paulo. Wolf, H. Froitzheim, K., 1997, *WebVideo - A Tool for WWW Based Teleoperation*,

- Wong C.L. et al., DMAP: A computer integrated system for design and manufacturing of axissymmetric parts, ASME Winter Meeting, Anaheim-USA, 1986.
- Wong, Y.S., Li, W.D., Fuh, J.Y.H., An Internet-enabled integrated system for co-design and concurrent engineering, *Computers in Industry*, Volume 55, Issue 1, September 2004, Pages 87-103.
- Wooldridge, M. and Jennings, N. - *Intelligent Agents: Theory and Practice*. Knowledge Engineering Review, 10(2), 115-152, 1995.
- Xu, X.W., He, Q., Striving for a total integration of CAD, CAPP, CAM and CNC, Robotics and Computer Integrated Manufacturing, Article in Press, accepted 7 August 2003.
- Yang, M. Y., Lee, S.C., A feature modification framework for the generation of alternative process plans, *Int. J. Prod. Res.*, Vol. 36, No. 7, pp. 1825-1840, 1998.
- Yarger, R. J., Reese, G., King, T., *MySQL e mSQL*, O'Reilly Associates, Inc. 101 Morris Street, CA 95472, First Edition, 1999.
- Ye, N., Gel, E.S., Li, X., Farley, T., Lai, Y-C., Web server QoS models: applying scheduling rules from production planning, *Computers Operations Research*, Volume 32, Issue 5, May 2005, Pages 1147-1164.
- Yellowley, I., Gunn, E. A., The optimal Subdivision of cut in Multipass Machining Operations, *International Journal of Production Research*, 27, 1573-1578, 1989.
- Yeo, S.H., A multipass optimization strategy for CNC lathe operations, *Int. J. Production Economics*, 40, pp. 209-218, 1995.
- Zhai, S. Milgram, P., 1991, A Telerobotic Virtual Control System, *Proc. SPIE*. Boston.
- Zhang, F., Zhang, Y.F., Nee, A.Y.C., Using genetic algorithms in process planning for job shop machining, *IEEE Transactions on Evolutionary Computations*, Vol. 1, N. 4, November, 1997.
- Zhang, S., Sheh, W., Ghenniwa, H., A review of Internet-based product information sharing and visualization, *Computers in Industry*, Volume 54, Issue 1, May 2004, Pages 1-15.
- Zhao Y., Ridgway K., Al-Ahmari A.M.A., Integration of CAD and a cutting tool selection system, *Computers Industrial Engineering*, 42, 17-34, 2002.
- Zhou S., Chin K.S., Yarlaga P. K.D.V., Internet-based intensive product design platform for product design, *Knowledge-Based System*, 16, 7-15, 2002.