

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

**COMPARAÇÃO DE DESEMPENHO ENTRE AS CONFIGURAÇÕES  
CELULAR, FUNCIONAL E CELULAR VIRTUAL ATRAVÉS DE  
SIMULAÇÃO E PROJETO DE EXPERIMENTOS**

**Tese submetida à**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA**

**para a obtenção do grau de**

**DOUTOR EM ENGENHARIA MECÂNICA**

**PAULO ANTONIO REAES**

**Florianópolis, setembro de 2008**

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

**COMPARAÇÃO DE DESEMPENHO ENTRE AS CONFIGURAÇÕES  
CELULAR, FUNCIONAL E CELULAR VIRTUAL ATRAVÉS DE  
SIMULAÇÃO E PROJETO DE EXPERIMENTOS**

**PAULO ANTONIO REAES**

**Esta tese foi julgada adequada para a obtenção do título de**

**DOUTOR EM ENGENHARIA**

**ESPECIALIDADE ENGENHARIA MECÂNICA  
Sendo aprovada em sua forma final**

---

**Eduardo Alberto Fancello, D.Sc. - Coordenador do Programa**

**BANCA EXAMINADORA**

---

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

---

**Prof. Abelardo Alves de Queiroz, Ph.D.**

---

**Prof. Paulo José de Freitas Filho, Dr.Eng.**

---

**Prof. Herman Augusto Lepikson, Dr.Eng.**

---

**Prof. Roberto Silvio Ubertino Rosso Jr., Ph.D.**

## SUMÁRIO

### LISTA DE FIGURAS

### LISTA DE TABELAS

### LISTA DE SÍMBOLOS, ABREVIACIONES E SIGLAS

### RESUMO

### ABSTRACT

<b>CAPÍTULO 1</b> .....	<b>16</b>
1.1 Definição do problema .....	17
1.2 Objetivo Geral .....	18
1.3 Objetivos Específicos .....	18
1.4 Justificativa.....	19
1.5 Suposições, Alcance e Limitações do Presente Trabalho.....	19
1.6 Estrutura da Tese .....	19
<b>CAPÍTULO 2</b> .....	<b>21</b>
2.1 Modelagem de configurações de tipos de processos de manufatura.....	21
2.1.1. Tipos de processos de manufatura .....	21
2.1.2. Principais tipos de leiautes .....	24
2.1.3. Padrões de referência e modelos empregados na comparação dessas configurações.....	29
2.2 Simulação das operações de manufatura .....	37
2.2.1. Técnica de simulação.....	37
2.2.3. Instrumentos de software de simulação disponíveis .....	41
2.3 Método empregado na geração da seqüência de produção .....	42
2.4 Emprego de DOE em combinação com simulação .....	46
2.4.1. Projeto de experimentos (DOE – <i>Design of Experiments</i> ).....	46
2.4.2. Emprego de software estatístico .....	49
2.5 Conclusões e Comentários .....	51
<b>CAPÍTULO 3</b> .....	<b>52</b>
3.1 Padrão e Modelos Adotados .....	52
3.2 Sequenciamento de Produção por Regras de Prioridade .....	55
3.3. O software de simulação .....	56
3.4 Metodologia para a avaliação dos resultados da simulação .....	57
3.4.1. O emprego do projeto de experimentos .....	57

3.4.2. Fatores e níveis considerados no projeto de experimentos. ....	57
3.4.3. Variáveis resposta utilizadas no projeto de experimentos.....	58
3.4.4. O software MINITAB para o projeto de experimentos .....	59
<b>CAPÍTULO 4</b> .....	<b>61</b>
4.1 Descrição do Padrão e Modelos Adotados .....	61
4.1.1 Descrição geral do padrão comum aos três modelos .....	61
4.1.2 Descrição específica do modelo da configuração <i>job shop</i> .....	65
4.1.3 Descrição específica do modelo da configuração celular virtual .....	67
4.1.4 Descrição específica do modelo da configuração celular tradicional .....	70
4.2 Descrição do Projeto de Experimentos Utilizado.....	73
4.2.1 Descrição dos fatores empregados no projeto de experimentos.....	73
4.2.2 Descrição das variáveis resposta empregados no projeto de experimentos....	75
4.2.3 Geração, coleta e transferência dos dados da simulação .....	76
4.2.4 Descrição do tratamento de dados realizado nas planilhas de resultados.....	83
4.2.5 Descrição do tratamento de dados realizado com o software Minitab .....	85
4.3 Recursos Empregados no Desenvolvimento da Tese.....	90
4.4 Conclusões.....	90
<b>CAPÍTULO 5</b> .....	<b>92</b>
5.1 Descrição Geral do Padrão Comum aos Três Modelos .....	92
5.1.1 Algumas variáveis do sistema .....	92
5.1.2 Os recursos e os conjuntos de recursos .....	94
5.1.3 Geração das demandas de cada tipo de peça .....	97
5.1.4 Geração dos lotes de peças e atribuição das seqüências de produção para cada peça .....	97
5.2 Descrição Específica do Modelo da Configuração <i>Job Shop</i> .....	99
5.2.1 Visão geral do modelo <i>job shop</i> .....	99
5.2.2 Animação do leiaute <i>job shop</i> .....	99
5.2.3 Lógica de um departamento.....	102
5.2.3.1 O bloco “Entrada” na lógica de um departamento .....	102
5.2.3.2 O bloco “Sistema Livre” na lógica de um departamento.....	103
5.2.3.3 O bloco “Blocos intermediários” na lógica de um departamento .....	103
5.2.3.4 O bloco “Obtida a máquina obtém operador interno” na lógica de um departamento.....	104

5.2.3.5 O bloco “Obtida a máquina busca qualquer operador” na lógica de um departamento.....	104
5.2.4 O trecho “Saídas” na configuração <i>job shop</i> .....	105
5.2.4.1 O bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos” .....	105
5.2.4.2 O bloco “Verificação de necessidade de <i>Warm up</i> ” .....	105
5.2.4.3 O bloco “Saída das Entidades”.....	108
5.3 Descrição Específica do Modelo Celular Tradicional .....	108
5.3.1 Visão geral do modelo celular tradicional.....	108
5.3.2 Animação do leiaute celular tradicional .....	110
5.3.3 Geração de operador sempre ocupado.....	110
5.3.4 Lógica de uma célula.....	112
5.3.5 Saídas do sistema.....	115
5.3.5.1 O bloco “Saída das Entidades”.....	116
5.3.5.2 O bloco “Transferência de dados para o Excel” .....	117
5.4 Descrição Específica do Modelo Celular Virtual.....	117
5.4.1 Visão geral do modelo celular virtual.....	118
5.4.2 Lógica de um departamento.....	119
5.4.2.1 O bloco “Entrada” na lógica de um departamento.....	119
5.4.2.2 O bloco “Operando com máquina já configurada”.....	120
5.4.2.3 O bloco “Operando com máquina não configurada”.....	123
5.4.3 O trecho “Saídas” .....	125
5.5 Conclusões.....	125
<b>CAPÍTULO 6 .....</b>	<b>127</b>
6.1 Verificação dos modelos .....	127
6.1.1 Verificação de características comuns a todos os modelos .....	127
6.1.1.1 Verificação da Geração dos Lotes de Peças e da Demanda das Peças ....	127
6.1.1.2 Verificação do roteiro de fabricação, dos tempos de movimentação, <i>setup</i> etc. ....	128
6.1.1.3 Verificação do tempo total médio de fabricação de um lote de peças.....	128
6.1.1.4 Verificação da flexibilidade dos operadores .....	129
6.1.2 Verificação de características específicas do modelo <i>job shop</i> .....	130
6.1.3 Verificação de características específicas do modelo celular virtual .....	131
6.1.3.1 Bloco “Entrada”: .....	131

6.1.3.2 Bloco “Operando com máquina já ajustada”:	131
6.1.3.3 Bloco “Operando com máquina não ajustada”:	132
6.1.4 Verificação de características específicas do modelo celular tradicional	132
6.1.4.1 A lógica de uma Célula tradicional	132
6.1.4.2 Verificação da Geração de Operador Sempre Ocupado	133
6.1.5 Verificação da necessidade de <i>warm-up</i>	134
6.2 Validação	134
6.3 Conclusões	135
<b>CAPÍTULO 7</b>	<b>136</b>
7.1 Análise dos resultados	136
7.2 Resultados envolvendo o tempo médio de fluxo	139
7.3 Resultados envolvendo o tempo médio de filas	140
7.4 Resultados envolvendo a ocupação dos operadores	141
7.5 Resultados envolvendo a produtividade	142
7.6 Conclusão	142
<b>CAPÍTULO 8</b>	<b>143</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>149</b>
<b>GLOSSÁRIO</b>	<b>153</b>
<b>APÊNDICE 1</b>	<b>156</b>
<b>APÊNDICE 2</b>	<b>159</b>
A2.1 Geração das demandas de cada tipo de peça	159
A2.2 Geração dos lotes de peças e atribuição das seqüências de produção para cada peça	160
A2.3 Descrição Detalhada do bloco “Entrada” na lógica de um departamento da Configuração <i>Job Shop</i>	162
A2.4 Descrição Detalhada do Bloco “Sistema Livre” na Lógica de um Departamento.	164
A2.5 Descrição Detalhada do Bloco “Blocos intermediários” na lógica de um departamento	167
A2.6 Descrição Detalhada do Bloco “Obtida a máquina obtém operador interno”	168
A2.7 Descrição Detalhada do Bloco “Obtida a máquina busca qualquer operador”	170
A2.8 Descrição Detalhada do Bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos”	174
A2.9 Descrição Detalhada do Bloco “Verificação de necessidade de Warm up”	175

A2.10 Descrição Detalhada do Bloco “Saída das Entidades” .....	175
A2.11 Descrição do Bloco “Geração de operador sempre ocupado” .....	177
A2.12 Descrição da Lógica de uma célula .....	177
A2.13 Descrição Detalhada do Bloco “Saída das Entidades” .....	180
A2.14 Descrição Detalhada do Bloco “Transferência de dados para o Excel” .....	181
A2.15 Descrição Detalhada do Bloco “Entrada” na lógica de um departamento.....	182
A2.16 Descrição Detalhada do Bloco “Operando com máquina já configurada”.....	184
A2.17 Descrição Detalhada do Bloco “Operando com máquina não configurada”..	186
A2.18 Descrição Detalhada do Bloco “Saídas de entidades” .....	189
<b>APÊNDICE 3.....</b>	<b>192</b>
A3.1 Descrição Detalhada da Verificação da Geração dos Lotes de Peças e da Demanda das Peças.....	192
A3.2 Descrição Detalhada do roteiro de fabricação, dos tempos de movimentação, setup etc. ....	196
A3.3 Descrição Detalhada da Verificação do tempo total médio de fabricação de um lote de peças.....	201
A3.4 Descrição Detalhada da Verificação da flexibilidade dos operadores.....	203
A3.4.1 Descrição detalhada da verificação da flexibilidade dos operadores no job shop .....	204
A3.4.2 Descrição detalhada da verificação da flexibilidade dos operadores ( <i>cross training</i> ) no celular tradicional .....	206
A3.4.3 Descrição detalhada da verificação da flexibilidade dos operadores ( <i>cross training</i> ) no celular virtual .....	211
A3.5 Verificação de Características Específicas do Modelo <i>Job Shop</i> .....	216
A3.5.1 Descrição detalhada do bloco “Entrada” do trecho Departamento dentro do modelo <i>job shop</i> .....	216
A3.5.2 Descrição detalhada do bloco “Sistema Livre” do trecho Departamento dentro do modelo <i>job shop</i> .....	218
A3.5.3 Descrição detalhada da verificação dos blocos “Obtida uma máquina obtém operador interno” ( $\eta = 2$ e $3$ ) e “Obtida uma máquina busca qualquer operador” ( $\eta = 2$ e $3$ ).....	220
A3.5.4 Descrição detalhada da verificação do “Priority” no <i>job shop</i> .....	222
A3.6 Descrição Detalhada do bloco “Entrada” do trecho Departamento dentro do Modelo Celular Virtual.....	226

A3.7 Descrição Detalhada da Verificação de Características Específicas do Modelo Celular Tradicional .....	229
A3.7.1 Descrição detalhada da verificação da lógica de uma Célula tradicional.....	229
A3.7.2 Descrição detalhada da verificação da geração de operador sempre ocupado .....	231
A3.8 Descrição Detalhada do Bloco de Verificação do Tempo Necessário de <i>warm-up</i> .....	232
A3.9 Descrição Detalhada da Validação .....	233
<b>APÊNDICE 4.....</b>	<b>241</b>
A4.1 Resultados envolvendo o tempo médio de fluxo.....	241
A4.1.1 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o job shop (JS) para $\eta = 1$ .....	241
A4.1.2 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 2$ .....	243
A4.1.3 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 3$ .....	244
A4.1.4 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$ .....	245
A4.1.5 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$ .....	247
A4.2 Resultados envolvendo o tempo médio de filas.....	248
A4.2.1 Diferença de tempo médio de filas entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 1$ .....	248
A4.2.2 Diferença de tempo médio de filas entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 2$ .....	250
A4.2.3 Diferença de tempo médio de filas entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 3$ .....	251
A4.2.4 Diferença de tempo médio de fila entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$ .....	252
A4.2.5 Diferença de tempo médio de fila entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$ .....	253
A4.3 Resultados envolvendo a ocupação dos operadores .....	254
A4.3.1 Diferença de ocupação dos operadores entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 1$ .....	254



A4.3.2	Diferença de ocupação dos operadores entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 2$ .....	256
A4.3.3	Diferença de ocupação dos operadores entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 3$ .....	257
A4.3.4	Diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$ .....	259
A4.3.5	Diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$ .....	260
A4.4	Resultados envolvendo a produtividade .....	261
A4.4.1	Diferença de produtividade entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 1$ .....	262
A4.4.2	Diferença de produtividade entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 2$ .....	263
A4.4.3	Diferença de produtividade entre o celular virtual (VCM) e o <i>job shop</i> (JS) para $\eta = 3$ .....	264
A4.4.4	Diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$ .....	265
A4.4.5	Diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$ .....	267

## LISTA DE FIGURAS

Figura 2.1 – Tipos de processos de manufatura e a relação Volume x Variedade - fonte: Slack et al. (2002).....	22
Figura 2.2 – Leiaute <i>Job Shop</i> – fonte: adaptado de Martins e Laugeni (2002).....	25
Figura 2.3 – Leiaute celular tradicional – fonte: adaptado de Martins e Laugeni (2002) .....	26
Figura 2.4 – Leiaute por produto – fonte: adaptado de Martins e Laugeni (2002).....	26
Figura 2.5 – Configuração do sistema com células virtuais .....	32
Figura 2.6 – <i>Flow shop</i> e célula virtual com 3 estágios e 2 máquinas por estágio - fonte: adaptado de Vakharia et al. (1999) .....	35
Figura 2.7 – Modelo <i>job shop</i> no padrão estabelecido por Jensen (2000).....	35
Figura 2.8 – Modelo híbrido no padrão estabelecido por Jensen (2000) .....	35
Figura 2.9 – Modelo celular tradicional no padrão estabelecido por Jensen (2000) .	36
Figura 2.10 – Modelos <i>Job Shop</i> e Celular Tradicional no padrão de Suresh e Slomp (2005) .....	36
Figura 2.11 – Etapas da simulação - fonte: Freitas Filho (2001).....	38
Figura 2.12 – Esquema do estudo de simulação aplicado a um experimento - fonte: adaptado de Sánchez (2007) .....	40
Figura 2.13 – Módulo de Entrada de Dados do software Minitab.....	50
Figura 2.14 – Resultados de uma análise de variância do software Minitab.....	50
Figura 4.1 Representação esquemática do modelo <i>Job Shop</i> .....	66
Figura 4.2 Fluxograma da lógica das operações dentro do departamento no <i>job shop</i> .....	67
Figura 4.3 Fluxograma da lógica das operações dentro do departamento no celular virtual.....	69
Figura 4.4 Representação esquemática do modelo da configuração celular tradicional .....	71
Figura 4.5 Fluxograma da lógica das operações dentro da célula na configuração celular tradicional.....	72
Figura 4.6 Tela do computador durante a simulação e coleta de dados .....	76
Figura 4.7 Software Arena com o arquivo “celular virtual_tempo de fluxo_24_10_07.doe” .....	77

Figura 4.8 Arquivo do software Microsoft Word .....	77
Figura 4.9 Software Microsoft Excel com o arquivo “transposer.xls” .....	78
Figura 4.10 O arquivo “Resultados_24_10_07_eta_2_3_tempo de fluxo.xls do Excel” .....	79
Figura 4.11 A tela do software Output Analyser .....	80
Figura 4.12 A tela expandida com os resultados calculados pelo Output Analyser ..	81
Figura 4.13 Os resultados do Output Analyser colados para um arquivo Word.....	81
Figura 4.14 Os dados do arquivo Word com vírgulas e a segunda coluna alinhada.	82
Figura 4.15 O texto convertido em tabela e a coluna selecionada para cópia .....	82
Figura 4.16 Dados da guia “minitab_eta_2_VCM-JS” .....	84
Figura 4.17 Tela do Minitab.....	87
Figura 4.18 Solicitação de dados para a Análise de Variância Balanceada.....	88
Figura 4.19 Os resultados da Análise de Variância na tela “session” .....	88
Figura 4.20 Representação com o gráfico “Scatterplots” com regressão linear.....	89
Figura 4.21 Representação com o gráfico “Scatterplots” com regressão quadrática	90
Figura 4.22 Representação com o gráfico “Scatterplots” com regressão cúbica .....	90
Figura 5.1 Tela do software Arena com algumas variáveis que definem os níveis dos fatores .....	93
Figura 5.2 Detalhes do módulo <b>Resource</b> em uma tela parcial do software Arena ...	95
Figura 5.3 Os recursos do conjunto “Depto 01” do módulo <b>Set</b> da configuração <i>job shop</i> .....	95
Figura 5.4 Os recursos do conjunto “operador_1” do módulo <b>Set</b> da configuração <i>job shop</i> .....	96
Figura 5.5 Conjunto de módulos responsáveis pela geração das demandas das peças .....	97
Figura 5.6 Conjunto de módulos responsáveis pela geração das peças da família	198
Figura 5.7 Conjunto de módulos responsáveis pela geração das peças “Peca_10”.	99
Figura 5.8 Visão geral do modelo da configuração <i>job shop</i> .....	99
Figura 5.9 Animação do leiaute <i>job shop</i> .....	100
Figura 5.10 Conjunto de módulos que formavam um departamento .....	101
Figura 5.11 Os blocos “intermediários” na lógica de um departamento .....	104
Figura 5.12 Módulos que compõem o bloco “Obtida a máquina obtém operador interno” .....	104
Figura 5.13 Os blocos “Obtida a máquina busca qualquer operador” .....	106

Figura 5.14 Os blocos no trecho “Saídas” na configuração <i>job shop</i> .....	107
Figura 5.15 O bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos” .....	108
Figura 5.16 O bloco “Verificação de necessidade de <i>Warm up</i> ” .....	108
Figura 5.17 O bloco “Saída das Entidades” .....	109
Figura 5.18 Visão geral do modelo celular tradicional.....	110
Figura 5.19 Animação do leiaute celular tradicional.....	111
Figura 5.20 Geração de operador sempre ocupado .....	112
Figura 5.21 Conjunto de módulos que formam uma célula tradicional.....	113
Figura 5.22 Lógica associada a cada uma das máquinas disponíveis em cada célula .....	114
Figura 5.23 Blocos associados a saídas do Sistema .....	116
Figura 5.24 O bloco “Saída das Entidades” .....	116
Figura 5.25 O bloco “Transferência de dados para o Excel” .....	117
Figura 5.26 O bloco “Visão geral do modelo celular Virtual” .....	117
Figura 5.27 A animação da configuração celular Virtual .....	118
Figura 5.28 O conjunto de módulos que formam um departamento .....	119
Figura 5.29 O bloco “Entrada” na lógica de um departamento.....	120
Figura 5.30 O bloco “Operando com máquina já configurada”.....	121
Figura 5.31 Trecho da máquina “Depto 01_1” no módulo “Operando com máquina já configurada” .....	122
Figura 5.32 O bloco “Operando com máquina não configurada” .....	124
Figura 5.33 Blocos associados às saídas do sistema.....	126
Figura 6.1 – Verificação da necessidade de <i>warm-up</i> .....	134
Figura A2.1 Caixa de diálogo do módulo <b>assign</b> “demandas1” .....	160
Figura A2.2 Detalhes do módulo <b>Sequence</b> em uma tela parcial do software Arena .....	162
Figura A2.3 O bloco “Entrada” dentro da lógica de um departamento .....	163
Figura A2.4 O bloco “Sistema Livre” na lógica de um departamento .....	165
Figura A2.5 Os blocos “Obtida a máquina busca qualquer operador” simplificado. ....	173
Figura A3.1 O módulo <b>ReadWrite</b> .....	193
Figura A3.2 O módulo <b>File</b> .....	194
Figura A3.3 A planilha Excel com os dados de demanda e geração de lotes.....	194

Figura A3.4 Telas da coleta de dados do <i>job shop</i> , do celular virtual e do celular tradicional .....	196
Figura A3.5 Dados do <i>job shop</i> transferidos para o Excel .....	198
Figura A3.6 Dados do celular tradicional transferidos para o Excel .....	200
Figura A3.7 Resumo dos três sistemas.....	200
Figura A3.8 Gráfico com as ocupações dos operadores .....	206
Figura A3.9 Gráfico com as ocupações dos operadores no celular virtual .....	215
Figura A3.10 Módulo ReadWrite dos os dados dos recursos .....	217
Figura A3.11 Módulo Record transferindo tamanho médio da fila “Nova1111” .....	219
Figura A3.12 O Programa Output Analyzer analisando o tamanho médio da fila “Nova1111” .....	219
Figura A3.13 Dados transferidos para verificar blocos dos departamentos no <i>job shop</i> .....	220
Figura A3.14 Dados reordenados para verificar blocos dos departamentos no <i>job shop</i> .....	221
Figura A3.15 Os quatro módulos <b>Seize</b> dos departamentos para $\eta=2$ no <i>job shop</i> .....	223
Figura A3.16 O atributo “Priority” no <i>job shop</i> .....	225
Figura A3.17 O critério SPT no <i>job shop</i> .....	225
Figura A3.18 Planilha “verificacao novo virtual.xls” com resultados da guia “Depto10” .....	227
Figura A4.1 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para $\eta = 1$ .....	242
Figura A4.2 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para $\eta = 2$ .....	243
Figura A4.3 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para $\eta = 3$ .....	244
Figura A4.4 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para $\eta = 1,2$ e 3 .....	245
Figura A4.5 – “Scatterplots” com regressão do tempo médio de fluxo VCM-CM para $\eta = 2$ .....	246
Figura A4.6 – “Scatterplots” com regressão do tempo médio de fluxo VCM-CM para $\eta = 3$ .....	247

Figura A4.7 – “Scatterplots” com regressão do tempo médio de fila VCM-JS para $\eta = 1$ .....	249
Figura A4.8 – “Scatterplots” com regressão do tempo médio de filas VCM-JS para $\eta = 2$ .....	250
Figura A4.9 – “Scatterplots” com regressão do tempo médio de filas VCM-JS para $\eta = 3$ .....	251
Figura A4.10 – “Scatterplots” com regressão do tempo médio de fila VCM-CM para $\eta = 2$ .....	252
Figura A4.11 – “Scatterplots” com regressão do tempo médio de fila VCM-CM para $\eta = 3$ .....	254
Figura A4.12 – “Scatterplots” com regressão da ocupação dos operadores VCM-JS para $\eta = 1$ .....	255
Figura A4.13 – “Scatterplots” com regressão da ocupação dos operadores VCM-JS para $\eta = 2$ .....	257
Figura A4.14 – “Scatterplots” com regressão da ocupação dos operadores VCM-JS para $\eta = 3$ .....	258
Figura A4.15 – <i>Scatterplot</i> com regressão da ocupação dos operadores (VCM-JS $\eta = 1, 2$ e $3$ ) .....	258
Figura A4.16 – “Scatterplots” com regressão da ocupação dos operadores VCM-CM para $\eta = 2$ .....	259
Figura A4.17 – “Scatterplots” com regressão da ocupação dos operadores VCM-CM para $\eta = 3$ .....	261
Figura A4.18 – “Scatterplots” com regressão da produtividade VCM-JS para $\eta = 1$ .....	262
Figura A4.19 – “Scatterplots” com regressão da produtividade VCM-JS para $\eta = 2$ .....	264
Figura A4.20 – “Scatterplots” com regressão da produtividade VCM-JS para $\eta = 3$ .....	265
Figura A4.21 – “Scatterplots” com regressão da produtividade VCM-CM para $\eta = 2$ .....	266
Figura A4.22 – “Scatterplots” com regressão da produtividade VCM-CM para $\eta = 3$ .....	267

## LISTA DE TABELAS

Tabela 2.1 – Relações possíveis entre tipos de processo de manufatura e tipos básicos de leiaute - fonte: adaptado de Slack et al. (2002) .....	27
Tabela 2.2 – Padrões de referência e modelos empregados.....	30
Tabela 2.3 – Estações de trabalho consideradas neste exemplo de modelo, e as respectivas máquinas que as compões - Adaptado: Mak e Wang (2002) .....	31
Tabela 2.4 - Distâncias entre estações de trabalho (ET) - fonte: Mak e Wang (2002)	31
Tabela 2.5 - Lista de Pedidos de Produção - fonte: Adaptado de Mak e Wang (2002) .....	32
Tabela 2.6 – Exemplo de modelo com 5 células tradicionais – fonte: Morris e Tersine (1990) .....	33
Tabela 2.7 – Exemplo de modelo com oito departamentos funcionais – fonte: Morris e Tersine (1990) .....	34
Tabela 2.8 – Exemplos de software de simulação - fonte: adaptado de Maria (1997)	42
Tabela 2.9 - Seqüenciamento por data prometida - fonte: Slack et al. (2002) .....	43
Tabela 2.10 – Seqüenciamento por LIFO - fonte: Slack et al. (2002) .....	43
Tabela 2.11 – Seqüenciamento por FIFO - fonte: Slack et al. (2002) .....	44
Tabela 2.12 – Seqüenciamento por SPT - fonte: Slack et al. (2002) .....	44
Tabela 2.13 – Trabalhos que comparam células virtuais, tradicionais e <i>job shop</i> .....	45
Tabela 2.14 – Resultados do ensaio de dureza para o exemplo considerado - fonte: Montgomery (1997) .....	47
Tabela 2.15 – Análise de variância para o exepimento consiedrado - ensaio de dureza - fonte: Montgomery (1997) .....	48
Tabela 2.16 – Série ortogonal para 13 fatores e 3 níveis.....	49
Tabela 3.1 – Fatores e quantidade de níveis selecionados para os experimentos ...	58
Tabela 3.2 – Variáveis resposta selecionadas para os experimentos.....	58
Tabela 4.1 – Tipos de peças .....	61
Tabela 4.2 – Os roteiros de fabricação das 50 peças .....	62
Tabela 4.3 – Os três tipos adotados de variação de demanda anual de peças.....	63
Tabela 4.4 – Tempos (em horas) dos intervalos entre chegadas de lotes de peças	64
Tabela 4.5 – Tempos de <i>setup</i> (em horas) .....	64
Tabela 4.6 – Tempos (em minutos) de movimentação de lotes e de operadores.....	65

Tabela 4.7 – Habilidades de cada operador.....	65
Tabela 7.1 Síntese dos resultados.....	138
Tabela A1.1 – Tipos de peças.....	156
Tabela A1.2 – Os roteiros de fabricação das 50 peças.....	157
Tabela A1.3 – Balanceamento das Cargas.....	158
Tabela A3.1 – Tempos médios de fluxo no <i>job shop</i> por n° de estágios.....	201
Tabela A3.2 – Tempos médios nos três sistemas por n° de estágios.....	202
Tabela A3.3 – Operadores e tipos de máquinas utilizadas no <i>job shop</i> .....	204
Tabela A3.4 – Operadores e tipos de máquinas utilizadas no <i>job shop</i> .....	205
Tabela A3.5 Operadores e máquinas utilizadas no celular tradicional por ordem de saída.....	208
Tabela A3.6 Operadores e máquinas utilizadas no celular tradicional por ordem de solicitação.....	209
Tabela A3. 7 Operadores e departamentos utilizados no celular virtual por ordem da saída dos lotes .....	213
Tabela A3. 8 Operadores e departamentos utilizados no celular virtual .....	214
Tabela A3. 9 Operadores e departamentos utilizados no exemplo do <i>job shop</i> .....	216
Tabela A3.10 Disponibilidade de operadores e departamentos utilizadas no <i>job shop</i> .....	218
Tabela A3.11 Solicitação de operadores no <i>job shop</i> .....	224
Tabela A4.1 Análise de variância tempo médio de fluxo VCM-JS para $\eta = 1$ .....	241
Tabela A4.2 Análise de variância tempo médio de fluxo VCM-JS para $\eta = 2$ .....	243
Tabela A4.3 Análise de variância tempo médio de fluxo VCM-JS para $\eta = 3$ .....	244
Tabela A4.4 Análise de variância tempo médio de fluxo VCM-CM para $\eta = 2$ .....	245
Tabela A4.5 Análise de variância tempo médio de fluxo VCM-CM para $\eta = 3$ .....	247
Tabela A4.6 Análise de variância tempo médio de filas VCM-JS para $\eta = 1$ ..	248
Tabela A4.7 Análise de variância tempo médio de filas VCM-JS para $\eta = 2$ ..	250
Tabela A4.8 Análise de variância tempo médio de filas VCM-JS para $\eta = 3$ ..	251
Tabela A4.9 Análise de variância tempo médio de fila VCM-CM para $\eta = 2$ ..	252
Tabela A4.10 Análise de variância tempo médio de fila VCM-CM para $\eta = 3$ ..	253
Tabela A4.11 Análise de variância da ocupação dos operadores VCM-JS para $\eta = 1$ .....	255
Tabela A4.12 Análise de variância da ocupação dos operadores VCM-JS para $\eta = 2$ .....	256



Tabela A4.13 Análise de variância da ocupação dos operadores VCM-JS para eta ( $\eta$ ) = 3 .....	257
Tabela A4.14 Análise de variância da ocupação dos operadores VCM-CM para eta ( $\eta$ ) = 2 .....	259
Tabela A4.15 Análise de variância da ocupação dos operadores VCM-CM para eta ( $\eta$ ) = 3 .....	260
Tabela A4.16 Análise de variância da diferença de produtividade VCM-JS para eta ( $\eta$ ) = 1 .....	262
Tabela A4.17 Análise de variância da produtividade VCM-JS para eta ( $\eta$ ) = 2.....	263
Tabela A4.18 Análise de variância da produtividade VCM-JS para eta ( $\eta$ ) = 3.....	264
Tabela A4.19 Análise de variância da produtividade VCM-CM para eta ( $\eta$ ) = 2 .....	266
Tabela A4.20 Análise de variância da produtividade VCM-CM para eta ( $\eta$ ) = 3 .....	267

## LISTA DE SÍMBOLOS, ABREVIações E SIGLAS

**ANOVA** - análise de variância.

**CEP** - controle estatístico de processo

**CL** – leiaute celular (*cellular layout*)

**CNC** - Controle Numérico Computadorizado

**COMP** – regra que libera mais facilmente operadores para operarem em outros departamentos

$d_{anual}$  - média da demanda anual da peça  $i$ .

**DF** – graus de liberdade (*degree freedom*)

$d_i$  - demanda gerada para a peça  $i$

**DOE** – Projeto de Experimento (*Design of Experiments*)

**DRC** – sistema sujeito a duas restrições de recurso (*dual resource constrained*)

$\eta$  - ( $\eta$ ) nível de *cross training*. Esses valores podem variar de 1 a 3, ou seja, respectivamente um operador poderia operar de uma a três máquinas.

**F** – Distribuição de Snedecor

**F crítico** – valor crítico na função de Snedecor para um dado nível de significância estatística

**FCFS** – Primeiro a chegar primeiro a ser operado (*First Come, First Served*)

**FIFO** - primeiro a chegar, primeiro a sair (*First In First Out*)

**FL** – *Functional layout*

$g_l$  - graus de liberdade

**IDLE** - regra mais restritiva na liberação de operadores para operarem em outros departamentos

**JS** – *Job Shop*

**K** – número de fatores

**LIFO** - último a chegar, primeiro a sair (*Last In First Out*)

**LNQ** – Critério de prioridade baseado na maior quantidade de serviços aguardando na fila (*Longest Number of Jobs in Queue*)

**MPS** - Plano Mestre de Produção (*Master Production Scheduling*)

**MQ** - média quadrada

**MQerro** - média quadrada do erro

**MQfator** - média quadrada de um fator

**MRP** - Planejamento das Necessidades de Materiais (*Material Requirements Planning*)

**MRP II** - Planejamento dos Recursos de Manufatura (*Manufacturing Resource Planning*)

**MS** – média da diferença quadrática

**n** – número de experimentos

**P** – probabilidade de erro do fator não influenciar significativamente

**q** - tamanho do lote

**R#** - replicação número (#)

**SCM** – *System of Cellular Manufacture*

**SOT** – Prioridade ao que tem de processamento mais curto (*Shortest Operation Time First*)

**SPT** – Tempo de processamento mais curto (*Shortest Processing Time*)

**SQ** - quadrados dos desvios

**SS** – diferença quadrática

**VCM** – *Virtual Cellular Manufacture*

**$\delta$**  - fator de redução do *setup*

## RESUMO

A escolha adequada do sistema de gestão da produção e do arranjo físico no chão de fábrica é fundamental para a viabilidade de uma atividade fabril na atual economia globalizada e altamente competitiva. Produtos cada vez mais especializados, com ciclos de vida sucessivamente menores buscam atender as freqüentes mudanças de mercado, exigindo um sistema fabril ágil e flexível que ao mesmo tempo seja economicamente eficiente. Grande parte das empresas industriais opera no sistema de lotes de produção utilizando o leiaute *job shop* (funcional) ou o celular tradicional. Entretanto nesses arranjos há sempre uma solução de compromisso entre eficiência e flexibilidade. Buscando dar solução a esta dificuldade surge o conceito de manufatura celular virtual, que reúne a alta flexibilidade de roteiro de fabricação do sistema *job shop* com a eficiência em termos de *setup* do sistema celular tradicional. Entretanto, a configuração celular virtual também pode apresentar as desvantagens de ambos os sistemas, tais como maiores tempos de movimentação e a necessidade de dispor de capacidade adicional de recursos. Por tratar-se de uma concepção que dispõe de poucos trabalhos de pesquisa, e em particular que realizem comparações analíticas abrangendo simultaneamente vários cenários, a implementação da configuração celular virtual na indústria é ainda quase inexistente, apesar da vantagem competitiva que esta mudança de configuração pode contribuir para sistemas de manufatura. Preenchendo esta lacuna, este trabalho de tese busca contribuir para a solução do problema de falta de eficiência e flexibilidade, avaliando, via simulação e projeto de experimentos, o desempenho dos sistemas de manufatura com configuração *job shop*, celular tradicional em comparação ao sistema de manufatura celular virtual, identificando assim as condições em que a aplicação deste último tipo de configuração é vantajoso com relação aos demais. Estes sistemas serão estudados mediante o auxílio de modelos, com idênticas capacidades e sujeitos a uma mesma demanda, e cujas seqüências de produção serão geradas por critérios de prioridade. Um significativo número de fatores, níveis e replicações, serão empregados com objetivo de delimitar adequadamente o escopo em que as células virtuais efetivamente geram vantagens competitivas.

**PALAVRAS-CHAVE:** células virtuais, sistemas de manufatura, simulação, projeto de experimentos.

## **ABSTRACT**

The appropriate choice of the production management system and the physical layout of the shop floor are fundamental for the viability of an industrial activity in the current globalized and highly competitive economy. More and more specialized products, with successively shorter life cycles seek to respond to the frequent market changes, demanding an agile and flexible industrial system that at the same time is economically efficient. Great part of the industrial companies manufacture their products in batch-based manufacturing systems using the job shop (functional) or the traditional cellular layouts. However, in those arrangements there is always a compromise solution between efficiency and flexibility. Seeking to provide a solution to this difficulty, the concept of virtual cellular manufacturing appears, which encompasses the high routing flexibility of job shop production with the efficiency in terms of setup of the traditional cellular configuration. However, the virtual cell configuration can also present the disadvantages of both systems, such as longer movement times and the need for additional resource capacity. Since few research works have been carried out on this concept, particularly attempting to accomplish analytical comparisons of several scenarios simultaneously, the implementation of the virtual cellular configuration in industry is still almost inexistent, despite the competitive advantage that this configuration change may provide to manufacturing systems. In order to fill this gap, this thesis seeks to contribute to the investigation of the problem of lack of efficiency and flexibility, by evaluating, through simulation and design of experiments, the performance of job shop and cellular traditional manufacturing systems in comparison with the a virtual cell manufacturing system, identifying the conditions in which the application of this latter configuration is advantageous compared with the others. These systems will be studied through the aid of models with identical capacities and subject to the same demand, and their production sequences are generated by priority criteria. A significant number of factors, levels and replications will be used with the objective of delimiting appropriately the scope in which virtual cells effectively lead to competitive advantages.

**KEY WORDS:** virtual cells, manufacturing systems, simulation, design of experiments

## CAPÍTULO 1

### INTRODUÇÃO

A escolha adequada do tipo de processo de produção e da configuração do leiaute no chão de fábrica é fundamental para a viabilidade de uma atividade fabril na atual economia globalizada e altamente competitiva.

Os tipos de processos, tratados e descritos de maneira abrangente por Slack, Chambers e Johnston (2002), podem ser classificados como pertencentes a cinco tipos principais: projeto, *jobbing*, lotes, em massa e contínuo. E, segundo os mesmos autores, esses sistemas adaptam-se a quatro tipos principais de configurações com seus respectivos arranjos físicos ou leiautes: posicional, por processo (também denominado *job shop* ou funcional), celular e por produto, definidos pela relação “volume x variedade” dos produtos a serem fabricados. Esta mesma relação entre volume e variedade irá determinar também possíveis economias de escala, níveis de flexibilidade de produção, *lead times* (tempos de atravessamento), níveis de gestão etc. Dentre das configurações apresentados acima, aqueles que são mais amplamente utilizados em sistemas de manufatura são os que operam com lotes de produção, utilizando como leiaute a manufatura celular ou o *job shop*.

Drolet et al. (1996) destacam as vantagens de cada uma dessas configurações. A manufatura celular exige muito menos controle e manipulação de materiais, com redução do tempo de *setup* e de fluxo, com aumento da qualificação do operador e melhoria nas relações humanas. Utiliza-se para isso a Tecnologia de Grupo (*Group Technology - GT*), que possibilita a agregação de configuração de máquinas sob a forma de células de trabalho, capazes de desempenhar todas as atividades para processar completamente grupos de peças, chamados de famílias de peças.

Entretanto, o leiaute *job shop* possibilita grande flexibilidade do roteiro de fabricação e melhor utilização das máquinas, além de criar uma sinergia proveniente do conjunto de máquinas (*pooling synergies*).

Tanto a redução dos tempos de fluxo e de *setup* como a flexibilidade de roteiro são características muito importantes para a viabilidade do negócio. Buscando aliar essas duas características em uma única configuração, surge o conceito das células virtuais. De acordo com Mclean et al. (1982), que são os criadores deste conceito, uma célula virtual não é identificável por um agrupamento físico de estações de trabalho, mas por arquivos de dados e processos dentro de um controlador. Ou seja, máquinas de diferentes departamentos são temporariamente associadas logicamente, constituindo e funcionando como células de manufatura para atenderem a produção de famílias de peças como no sistema celular tradicional, ainda que fisicamente estejam dispostas como no leiaute *job shop*.

Kannan (1998) destaca que a manufatura celular virtual é constituída de células lógicas baseadas em regras que programação de produção (*schedule*) por famílias de peças que permitem manter o leiaute por processo. Elas envolvem a alocação de máquinas por família de peças em vez de tarefas individuais. Ou seja, logicamente ela opera como uma célula tradicional, voltada à tecnologia de grupo. Como as células são virtuais e não físicas, a maior vantagem dessa configuração é a possibilidade de mudanças freqüentes do leiaute virtual (sem os custos ou rupturas decorrentes das mudanças de leiaute físico), que frequentemente é o maior desestímulo para o emprego da manufatura celular. Além disso, permite-se mudanças de leiaute para se ajustar e adaptar às variações de mix de produtos (pelas oscilações normais de demanda nas economias de mercado aberto), e portanto o uso do planejamento de produção orientado às famílias de peças no leiaute funcional mantém a sinergia do conjunto de máquinas típica do *job shop*.

Novos atributos e conceitos foram agregados às células virtuais por Drolet et al., (1996), Sarker e Li (2001), Ko e Egbelu (2003) entre outros, tais como a conceituação das máquinas como entidades lógicas, a configuração do controle dessas células, o compartilhamento de máquinas por células distintas, o gerenciamento das informações e a eficiência das células virtuais na flexibilidade das dimensões das instalações. Dessa forma, o conceito das células virtuais tornou-se bastante robusto e operacional.

Paradoxalmente às significativas vantagens competitivas potenciais desse conceito de configuração de manufatura, observam-se poucos trabalhos de pesquisa que avaliam esses potenciais e, até o momento, uma ausência de comprovação científica de que essas vantagens possam ser realmente obtidas na prática para alguns cenários reais. Quase como uma conseqüência natural, verifica-se também a inexistência do emprego dessa configuração nas empresas. Verifica-se ainda que poucas pesquisas foram realizadas com essas células usando-se técnicas de simulação e de projeto de experimentos (DOE – *Design of Experiments*).

Em síntese, o tema dessa tese é sobre a avaliação dos cenários em que ocorrem vantagens competitivas no emprego da manufatura celular virtual em comparação com as configurações *job shop* e celular tradicional.

## **1.1 Definição do problema**

Uma vez que há avanços conceituais na configuração celular virtual, que teoricamente busca reduzir os tempos de fluxo e de *setup* associados à flexibilidade de roteiro, questiona-se nesse trabalho se há cenários, e neste caso quais seriam, em que esta configuração gera vantagens competitivas sobre as configurações *job shop* e celular tradicional. Por outro lado,

quais seriam os modelos e instrumentos adequados para representar as características dessas três configurações de maneira que elas pudessem ser comparadas? E ainda, sob quais parâmetros seriam realizadas essas avaliações?

## 1.2 Objetivo Geral

Neste trabalho de tese busca-se desenvolver modelos e ferramentas que permitam comparar, via simulação e projeto de experimentos, os sistemas de manufatura com configuração *job shop*, celular tradicional e o sistema de manufatura celular virtual, visando avaliar as vantagens da aplicação deste último tipo de configuração em comparação com os outros.

## 1.3 Objetivos Específicos

Os objetivos específicos são:

- Modelar, através de software de simulação três sistemas da manufatura: o primeiro com tipo de processo por lote com configuração *job shop*, outro com configuração celular tradicional e o último utilizando um sistema de manufatura celular virtual. O software utilizado foi o Arena versão 9.0.
- Utilizar um sistema fabril que envolvesse diferentes tipos de processos, por exemplo, torneamento, furação e fresamento. Tal sistema de manufatura deveria conter pelo menos três tipos de máquinas distintos, relativos a três distintos processos.
- Seqüenciar a produção (*schedule*) através de critérios de prioridade tais como Data Prometida, LIFO (*Last In First Out*), FIFO (*First In First Out*) e Menor Tempo de Processamento (SPT - *Shortest Processing Time*).
- Empregar projeto de experimentos (DOE) onde sejam identificados e validados os fatores e níveis mais significativos, assim como as variáveis resposta que melhor contribuem para avaliar o desempenho do sistema. Para esta finalidade, foi usado o software MINITAB versão 14 ou superior.
- Empregar variados fatores, dentre eles a flexibilidade dos operadores, os tamanhos de lote para cada peça, as variações no tempo de *setup* de novas ordens de produção e os tempos aleatórios de chegada de novos pedidos.
- Empregar diversas variáveis resposta, entre elas o tempo de fluxo médio.
- Comparar os resultados dos leiautes propostos nas diversas situações descritas acima.



#### **1.4 Justificativa**

Esta pesquisa se justifica por vários motivos. Primeiramente pela vantagem competitiva que pode ser vislumbrada pela mudança para a configuração celular virtual. Além disso, há poucos trabalhos envolvendo as células virtuais, e dentre estes trabalhos poucos deles aplicam as técnicas de simulação associadas à avaliação de resultados através de projeto de experimentos (DOE). A significativa quantidade de fatores (e de seus níveis de variação) considerados neste trabalho amplia o escopo de estudo dos poucos trabalhos anteriores. Finalmente, por apresentar uma nova característica associada às células virtuais, confere-se à presente pesquisa uma atualidade e abrangência significativa, permitindo chegar a conclusões importantes sobre os cenários pesquisados.

#### **1.5 Suposições, Alcance e Limitações do Presente Trabalho**

Para viabilizar esta pesquisa, a avaliação será realizada mediante a simulação de modelos que representem as três configurações. Para a análise dos resultados da simulação serão usados métodos estatísticos. Por outro lado, para dar maior abrangência e simplificação ao estudo serão consideradas apenas máquinas convencionais (não máquinas CNC) tais como tornos, fresadoras etc.

Para estas pesquisas e análises foram utilizadas técnicas de simulação e de projeto de experimentos (DOE – *Design of Experiments*). Como fatores de controle foram empregados os tamanhos de lote, a flexibilidade dos operadores, os tempos de preparação, a variabilidade de demanda e dos intervalos de chegadas de lotes de peças. Na modelagem como um todo, outros fatores foram utilizados para retratar de forma mais rica e completa a realidade dos cenários industriais, tais como critérios de prioridade na programação (*schedule*) ou sequenciamento das tarefas processadas, a quantidade de processos e de máquinas, os tempos de espera, movimentação etc.

Dessa forma se pôde avaliar o escopo real das vantagens competitivas dessa concepção alternativa, buscando estimular a sua efetiva aplicação nas empresas, em particular nos cenários em que comprovadamente essas vantagens sejam bastante significativas e inquestionáveis.

#### **1.6 Estrutura da Tese**

A presente tese está estruturada em oito capítulos. Este primeiro capítulo introduz o contexto dessa pesquisa, destacando a importância do tema, e finalmente, destaca o

objetivo geral e os objetivos específicos. O segundo capítulo aborda o estado da arte nas pesquisas que avaliam as células virtuais, sua gestão e desempenho em relação a outros sistemas e configurações. O terceiro capítulo descreve a metodologia aplicada visando atingir os objetivos traçados, que consiste no uso de simulação e projeto de experimentos. O capítulo quatro trata da apresentação do modelo em termos conceituais. O quinto capítulo descreve o modelo físico, suas características, especificações e limitações. O capítulo sexto aborda os métodos de verificação e validação empregados no modelo para certificar a idoneidade dos dados obtidos. O sétimo capítulo contém a análise dos resultados obtidos nos experimentos. E finalmente, o oitavo capítulo descreve as conclusões obtidas na pesquisa.

## **CAPÍTULO 2**

### **REVISÃO DA LITERATURA**

Este capítulo aborda o estado da arte dos temas relacionados com a tese. No capítulo anterior ficou definido como objetivo da tese implementar modelos e ferramentas que permitam comparar, via simulação combinada com projeto de experimentos, o desempenho das operações de manufatura nas configurações *job shop*, celular tradicional e o sistema de manufatura celular virtual, visando avaliar as vantagens da aplicação deste último tipo de configuração em comparação com os demais. Assim, no contexto deste trabalho foram identificadas quatro áreas em que devem ser abordados os respectivos estados da arte: a modelagem de leiautes de tipos de processos de manufatura, a simulação das operações de manufatura nas configurações especificadas, o seqüenciamento de produção através de regras de prioridade, e os tratamentos dos resultados através de projeto de experimentos e software estatístico.

Neste sentido, o item 2.1 descreve o estado da arte com relação à modelagem de diferentes configurações de processos de manufatura. E para esta abordagem torna-se necessário descrever os tipos de processos de manufatura, os leiautes empregados e os modelos que estão sendo utilizados. Já o item 2.2 aborda a simulação das operações de manufatura através de software, e para tanto é feita uma breve revisão dos processos de simulação e dos instrumentos de software disponíveis. O item 2.3 trata do seqüenciamento de produção através de regras de prioridade. Por último o item 2.4 descreve a aplicação de projeto de experimentos como instrumento no tratamento estatístico de dados obtidos por simulação, e destaca os benefícios do emprego de softwares estatísticos.

#### **2.1 Modelagem de configurações de tipos de processos de manufatura**

Neste item é feita uma revisão do estado da arte dos tipos de processos de manufatura, dos leiautes empregados nesses sistemas e, finalmente, dos modelos empregados na literatura para representar estas configurações e permitir comparações entre elas.

##### **2.1.1. Tipos de processos de manufatura**

Os principais tipos de processos de manufatura foram tratados e descritos com profundidade por Slack et al. (2002), que os classificam como pertencentes a cinco tipos distintos: projeto, *Jobbing*, lotes, em massa e contínuo. Por outro lado, Martins e Laugeni (2002) empregam uma classificação mais simples e são menos extensos e aprofundados nesta abordagem. Estes últimos utilizam o termo “tipos de operações industriais” em vez de “tipos de processos de manufatura”, e consideram a existência de quatro tipos de operações

industriais, a saber, processos contínuos, linha de montagem, *job shop* e projetos. Pode-se verificar que Martins e Laugeni (2002) empregam o termo *job shop* para um tipo de operação (ou de processo) e não como sinônimo de leiaute *job shop*. Por esses motivos, neste projeto de tese utiliza-se a classificação e nomenclatura empregadas por Slack et al. (2002).

Os fatores que mais determinam estes tipos de processos de manufatura são o volume e a variedade de produtos. A figura 2.1 ilustra o relacionamento desses dois fatores e os cinco tipos de processos de manufatura.

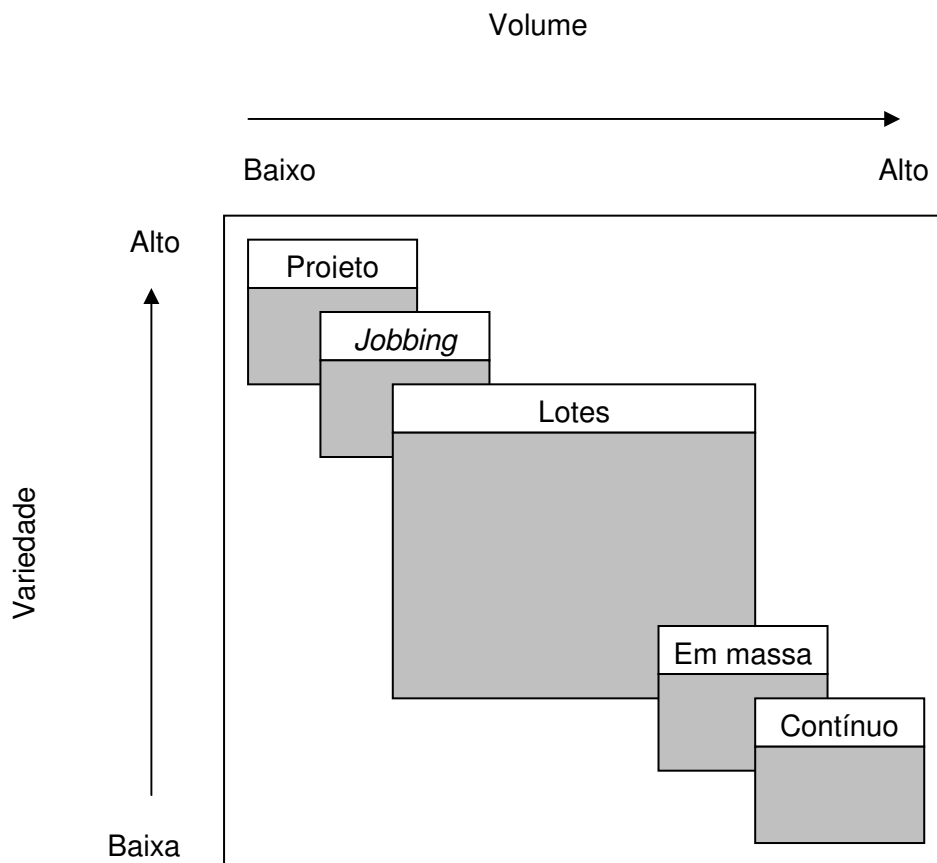


Figura 2.1 – Tipos de processos de manufatura e a relação Volume x Variedade - fonte: Slack et al. (2002)

Pode-se verificar que para volumes baixos de produção, haverá uma maior variedade (*mix*) de produtos. Por outro lado, no caso de volumes altos de produção, haverá uma baixa variedade de produtos, para viabilizar economia de escala. A construção de uma turbina única, específica, de um novo projeto, caracteriza o primeiro caso, isto é, baixo volume e elevada variedade, enquanto a fabricação de um barbeador descartável é um exemplo do segundo caso, que corresponde a um enorme volume e baixa variedade. Além disso, o volume e a variedade influenciam diretamente a especificação de outras importantes características industriais, tais como padronização de processos, localização de

equipamentos, fluxo de informações e materiais, tecnologia de processo, custo unitário, logística etc.

Cada um desses tipos de processos de manufatura tem características bem específicas. O tipo de processo de manufatura de projeto opera com baixíssimo volume de produção, onde cada item necessita de longos períodos para a sua produção. Os itens produzidos atendem a pedidos sob encomenda, logo com alta variedade, normalmente apresentando grandes intervalos entre pedidos. O uso dos equipamentos e das operações é extremamente variado.

O tipo de processo de manufatura de *jobbing* se caracteriza por estar voltado à produção de itens muito variados e em baixa quantidade. Ao contrário do tipo de projeto, os pedidos não são normalmente inéditos, e com freqüência o pedido compreende dois ou mais itens iguais, constituindo um pequeno lote, que normalmente não são novamente solicitados. Na prática, o *jobbing* corresponde ao caso intermediário entre o processo de projeto e por lote, sendo que por vezes seus pedidos se assemelham ao primeiro tipo, e outras vezes ao outro. Assim, os equipamentos costumam ser utilizados por vários pedidos porque existe alguma semelhança entre os pedidos.

O tipo de processo de manufatura por lote é muito freqüente nas instalações industriais porque apresenta uma relação equilibrada entre volume e variedade, conseguindo atender a uma gama maior de clientes. Os mesmos lotes de pedidos costumam se repetir e alternar, e devido a isso alguns autores o classificam sob o nome de produção repetitiva por lote (TUBINO, 1997) em contraste à produção sob encomenda. Dessa forma, obtém-se redução de custos, não ao mesmo nível de uma produção em massa ou contínua, mas bastante significativa com relação ao processo por *jobbing* ou de projetos. Freqüentemente esta é a solução de compromisso entre uma demanda que não justifica uma instalação dedicada, mas que freqüentemente consegue suprir a demanda a preços competitivos por contar sempre com alguma escala de produção.

O quarto tipo de processo de manufatura é o de produção em massa, o qual apresenta volumes de produção muito altos e baixa variedade. Caracteriza-se também por sua alta previsibilidade. O planejamento de produção é muito mais simples do que o de lotes, sendo mais complexo apenas em relação ao contínuo. Ele é tipicamente utilizado pela indústria automobilística.

O último tipo de processo de manufatura é o contínuo, o qual apresenta os mais elevados volumes de produção e os mais baixos de variedade. Em alguns casos as instalações são tão dedicadas e especializadas ao produto processado que a produção é literalmente contínua, sem paradas, vinte e quatro horas por dia, sete dias na semana, como

é o caso de refinarias e siderúrgicas. Usualmente são planejadas para uma operação específica, de alto investimento, como no caso de celulose.

### 2.1.2. Principais tipos de leiautes

O leiaute de manufatura estabelece a localização física dos recursos transformadores, ou seja, instalações, máquinas, equipamentos, pessoal etc. Estabelece também como os recursos transformados, isto é, matéria prima, materiais em processo, peças etc, devem fluir durante as operações. Ele busca privilegiar o fluxo, reduzindo os custos e aumentando a eficácia geral da produção. Segundo Slack et al. (2002), a importância do leiaute para os sistemas produtivos decorre dos seguintes motivos, dentre outros:

- Qualquer mudança de leiaute é freqüentemente uma atividade difícil, de longa duração e custo, devido às dimensões físicas dos recursos de transformação movidos.
- Refazer o leiaute de uma operação existente em geral acarreta ruptura de fornecimento, levando à insatisfação dos consumidores internos e externos, e a perdas na produção.
- Erros no projeto de leiaute geram fluxos longos e confusos, estoques desnecessários, filas, tempos de processamento longos, falta de flexibilidade na linha, altos custos etc.

Por isso, uma vez definido o tipo de processo de manufatura, em função da relação volume variedade, conforme descrito no item anterior, a próxima decisão deve ser o tipo de leiaute.

Slack et al. (2002) definem quatro tipos tradicionais de leiaute: posicional, por processo, celular tradicional e por produto. Martins e Laugení (2002) empregam uma classificação idêntica, basicamente se diferenciando apenas na nomenclatura, uma vez que, em vez de leiaute por produto, eles empregam o termo leiaute em linha. O primeiro, o posicional, também conhecido como leiaute de posição fixa, é aquele cuja característica principal é a de que os recursos transformadores se movem em torno dos recursos transformados, exatamente o contrário do que ocorre nos demais leiautes, normalmente porque nesse caso o produto é muito grande para ser movido de forma conveniente. Um estaleiro ou um prédio são exemplos desse tipo.

O segundo é o leiaute por processo, conhecido também como funcional ou ainda *job shop*, onde processos similares são localizados juntos um do outro. Este leiaute é apropriado para sistemas de manufatura que devem processar uma gama de produtos com

necessidades variadas de seqüências de processo, e por este motivo o fluxo de operação dos materiais é bastante complexo. Um exemplo de aplicação é a usinagem de peças de lotes pequenos e pouco freqüentes. A figura 2.2 auxilia a compreensão do conceito desse leiaute através de um exemplo simples envolvendo quatro setores que reúnem processos idênticos, torno revólver (TR), plainas (P), retíficas (R), e furadeiras (F), onde o produto passa por apenas dois desses setores.

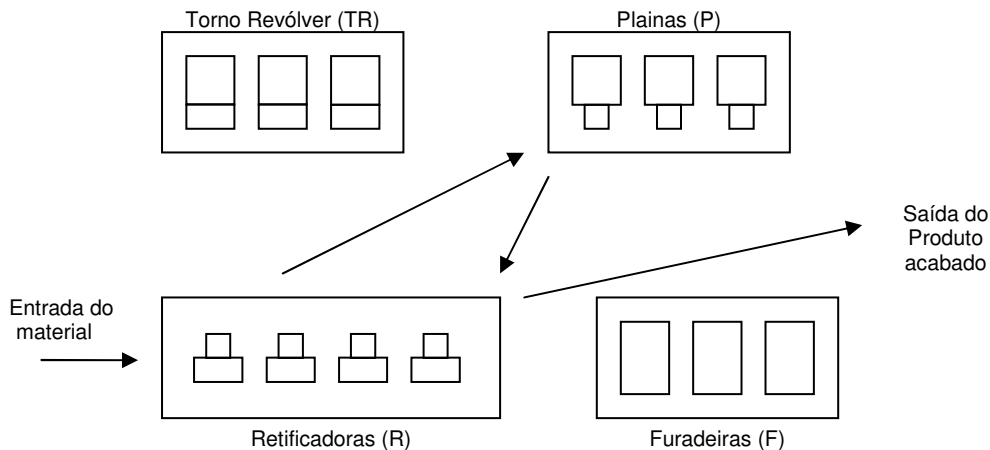


Figura 2.2 – Leiaute *Job Shop* – fonte: adaptado de Martins e Laugeni (2002)

O terceiro tipo de leiaute é o celular tradicional, constituído basicamente por células de manufatura. As células têm como característica a presença de recursos transformadores variados, isto é, com processos distintos, voltadas a operar sobre famílias de peças, as quais são formadas normalmente por peças que têm características semelhantes de processamento. As peças processadas em uma célula seguem para uma outra célula. Desta forma, diminui-se o fluxo de materiais típico do leiaute por processo e as reprogramações de *setup*. Como exemplo deste tipo de leiaute tem-se as indústrias de componentes de computador. A figura 2.3 ilustra o conceito de leiaute celular mediante um exemplo envolvendo duas células (1 e 2) que produzem respectivamente duas famílias de peças (A e B).

O último tipo, segundo Slack et al. (2002), é o leiaute por produto, ou em linha, segundo Martins e Laugeni (2002). Sua característica principal é procurar distribuir os recursos produtivos orientados ao melhor fluxo do produto, seus componentes, sub-conjuntos e conjuntos, pela planta industrial. Haverá assim um roteiro de fabricação predefinido e otimizado, e por isso é freqüente chamar-se este leiaute de arranjo em fluxo. Um exemplo típico deste tipo de leiaute são as montadoras automotivas. A figura 2.4 ilustra este tipo de arranjo, com a representação de três linhas de produção independentes.

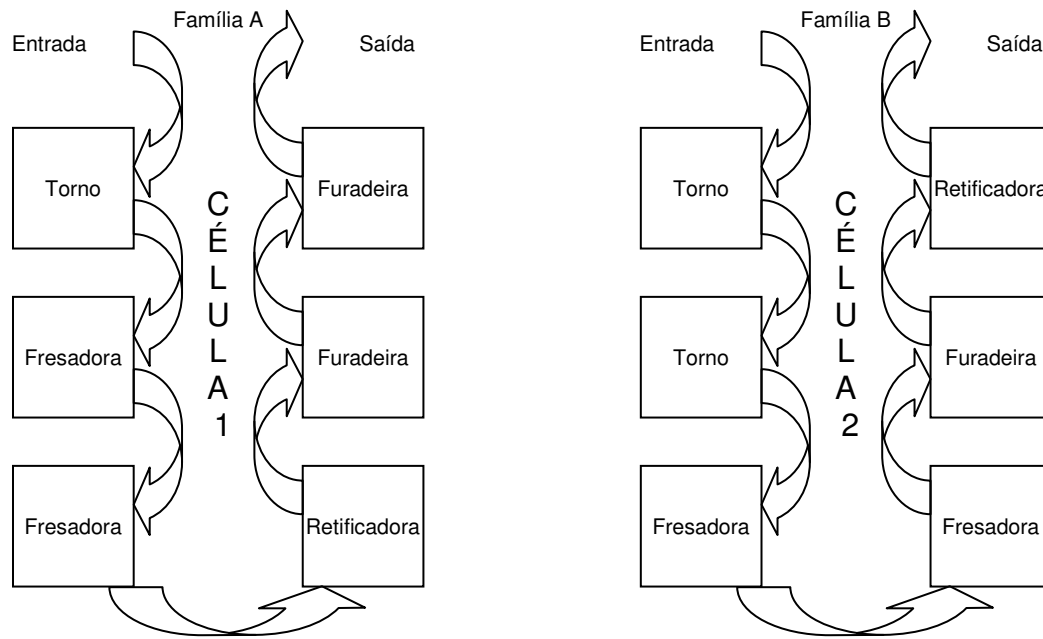


Figura 2.3 – Layout celular tradicional – fonte: adaptado de Martins e Laugeni (2002)

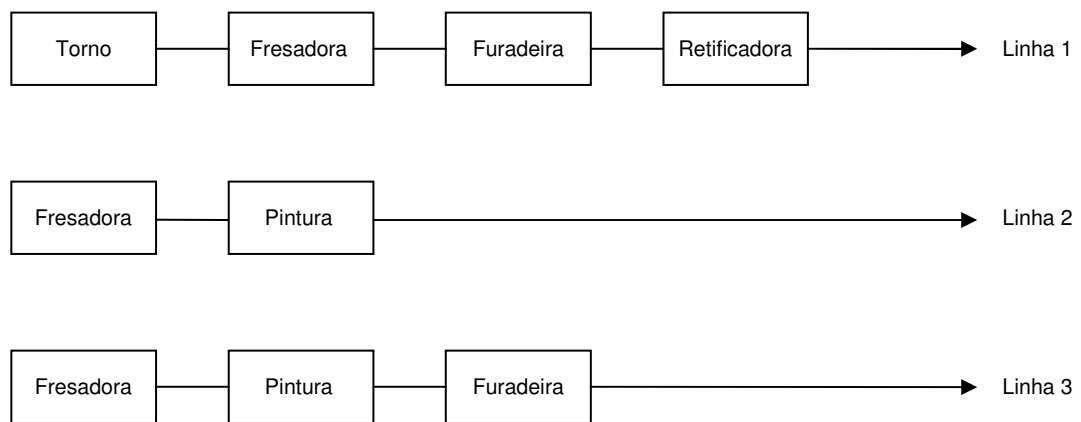


Figura 2.4 – Layout por produto – fonte: adaptado de Martins e Laugeni (2002)

O tipo de produção e processo de manufatura afeta significativamente o tipo de layout, mas é importante frisar que esta relação não é unívoca. A tabela 2.1 indica as possibilidades de layouts para cada tipo de processo. Assim, por exemplo, a produção por lote tradicionalmente admite tanto o layout por processo como o celular tradicional.



Tabela 2.1 – Relações possíveis entre tipos de processo de manufatura e tipos básicos de leiaute - fonte: adaptado de Slack et al. (2002)

<i>TIPOS DE PROCESSO DE MANUFATURA</i>	<i>TIPOS BÁSICOS DE LEIAUTE</i>
Processo por Projeto	Leiaute Posicional
Processo Tipo <i>Jobbing</i>	Leiaute por Processo
Processo Tipo Lote	Leiaute celular
Processo em Massa	Leiaute por Produto
Processo Contínuo	

Na literatura não se encontra nessa classificação as células virtuais que, como foi abordado no capítulo 1, foram criadas por MCLEAN et al. (1982) objetivando a redução dos tempos de fluxo e de *setup* em conjunto com o aumento da flexibilidade de roteiro. Como mencionado anteriormente, as células virtuais não são identificáveis por pertencer a um agrupamento físico de estações de trabalho, mas como arquivos de dados e processos dentro de um controlador. Fisicamente os recursos de cada célula virtual estão separados entre si, e por outro lado próximos de recursos de outras células e com esses justapostos, de forma semelhante ao *job shop*, onde recursos de processos idênticos ficam juntos. Mas operam virtualmente como células pois os seus recursos são do ponto de vista lógico – ainda que não fisicamente – agrupados em forma de célula, ou seja, os recursos são dedicados à manufatura de uma família de peças, como as células tradicionais. Entretanto, como este agrupamento é lógico, assim que deixa de haver na fila lotes da família de peças que causou a formação da célula virtual, esses recursos são desagrupados logicamente, podendo cada recurso individualmente tornar a se agrupar com outros recursos para constituir uma nova célula virtual preparada para atender outra família de peças que de fato tem lotes na fila aguardando para serem processados.

A escolha do leiaute a ser utilizado em um sistema de manufatura é bastante influenciada pela relação volume x variedade. Assim, à medida que a variedade de produtos diminui e o volume aumenta, a escolha do leiaute desloca-se do *job shop* para o celular tradicional, e a seguir, deste último para o leiaute por produto. Por outro lado, quando a demanda do produto não exige um processo de fabricação contínuo, ou seja, não justifica um leiaute dedicado (leiaute por produto), a escolha seguinte recai em um processo de fabricação por lote, que, conforme mostrado na tabela 2.1, admite os outros dois leiautes citados anteriormente, *job shop* e o celular tradicional. Como o sistema mais amplamente utilizado nas empresas atualmente é o que opera com lotes de produção, são muito freqüentes as instalações industriais que utilizam uma combinação desses leiautes, ou seja, o leiaute misto.

Como o objetivo deste trabalho de tese é comparar os sistemas de manufatura com configuração *job shop*, celular tradicional e o sistema de manufatura celular virtual, torna-se interessante destacar as características específicas desses três tipos de configuração, apontadas por Martins e Laugeni (2002) e Slack et al. (2002).

As características do leiaute *job shop* são as seguintes:

- As máquinas são agrupadas por similaridade de processo.
- Cada produto pode ter um roteiro de fabricação diferente. O fluxo é complexo e difícil de controlar.
- Tempos altos de movimentação.
- Alta variedade e baixa quantidade de produtos.
- Alta flexibilidade.
- Baixa utilização dos equipamentos.
- Tendência a formar estoques em processo e filas.
- Tempos de *setup* elevados.

As características do leiaute celular tradicional são as seguintes:

- Máquinas de processos distintos são agrupadas em células.
- O roteiro de fabricação é mais simples, mais fácil de controlar.
- Tempos baixos de movimentação.
- Baixa variedade e alta quantidade de produtos.

- Baixa flexibilidade. Cada célula processa uma família de peças. Os produtos que passam por essas células são pré-selecionados. Refazer o leiaute da célula é caro.
- Baixa utilização dos equipamentos, requerendo capacidade adicional.
- Redução de estoques em processos e filas.
- Tempos de *setup* reduzidos.

De acordo com a literatura, as características da configuração celular virtual são as seguintes:

- As máquinas estão distribuídas indistintamente pela planta. Entretanto, ao longo do tempo, máquinas de processos distintos são agrupadas periodicamente em células virtuais, isto é, sem serem movidas de suas posições físicas e sem custos adicionais.
- Cada produto pode ter um roteiro de fabricação diferente. O fluxo é complexo.
- Tempos altos de movimentação.
- Permite operar com ótimo desempenho mesmo com características de produção de alta variedade e baixa quantidade de produtos.
- Alta flexibilidade.
- Baixa utilização dos equipamentos.
- Tempos de *setup* reduzidos.
- Reúne características muito positivas de cada uma das configurações anteriores, tais como redução dos tempos de fluxo e de *setup*, com a conseqüente redução de estoques em processos e filas, assim como flexibilidade de roteiro.
- Entretanto, reúne também características negativas de cada uma das configurações anteriores, tais como necessitar percursos maiores para movimentar as peças, além de requerer uma capacidade adicional de recursos.

### 2.1.3. Padrões de referência e modelos empregados na comparação dessas configurações

Este tópico descreve os padrões de referência e modelos empregados na literatura para representar e comparar a configuração celular virtual com o celular tradicional e o *jobshop*. O termo modelo de configuração nesta tese é empregado para significar a especificação e

representação utilizada em uma simulação de uma configuração individual e específica, seja celular, celular virtual ou *job shop*. Já o termo padrão de referência é utilizado para significar as características e definições estabelecidas pelos autores de trabalhos de pesquisa que comparam a configuração celular virtual com as demais configurações através de um ou mais modelos de configuração.

Tabela 2.2 – Padrões de referência e modelos empregados

<i>PADRÕES</i>	<i>MODELOS DE CONFIGURAÇÕES</i>
DROLET (1989); MAK e WANG (2002)	JS e VCM: 9 estações de trabalho com 2-3 máquinas iguais (total de 21 máquinas)
MORRIS e TERSINE (1990); KANNAN e GHOSH (1996); KANNAN (1997a); KANNAN (1998).	VCM: 8 departamentos com 3-4 máquinas iguais (total de 30 máquinas) SCM: 5 células com 4-8 máquinas diferentes (total de 30 máquinas)
KANNAN (1997b)	JS, VCM1 e VCM2: 8 departamentos com 3-4 máquinas iguais (total de 30 máquinas)
VAKHARIA et al. (1999)	<i>Flow Shop</i> e <i>Célula Virtual</i> com muitas variações de números de estágios, máquinas etc.
JENSEN (2000)	JS: 4 departamentos em linha com 4 máquinas iguais (total de 16 máquinas) Híbrido: 4 células em linha, duas a duas, formando 2 linhas (total de 16 máquinas) SCM: 4 células independentes com 4 máquinas cada (total de 16 máquinas)
SURESH e SLOMP (2005)	VCM e FL: 10 departamentos com 3-4 máquinas (total de 31 máquinas) SCM: 5 células com 6-7 máquinas diferentes (total de 31 máquinas)

Suresh e Slomp (2005), na revisão da literatura que realizam, destacam os progressos e o rigor metodológico nas pesquisas realizadas nos últimos anos com emprego da simulação, que comparam organizações celulares tradicionais, celulares virtuais e funcionais. Para tanto, os padrões e modelos empregados devem representar fielmente as características essenciais dos sistemas que representam. Os modelos, dentro do mesmo padrão de referência, devem estar balanceados, isto é, com idênticas demandas e capacidades, para que a comparação faça sentido. Da mesma forma, é particularmente interessante a utilização de algum padrão e modelos já empregados em outros trabalhos similares para que os resultados e as conclusões das pesquisas possam também ser relacionados. Objetivando visualizar os padrões e modelos empregados em pesquisas que comparam o desempenho de sistemas que empregam configuração celular tradicional,

celular virtual ou *job shop*, essas informações são apresentadas de forma sintética na tabela 2.2.

O primeiro padrão de referência desta tabela é o utilizado por Drolet (1989) e posteriormente por Mak e Wang (2002). Este padrão de referência é descrito a seguir, com mais detalhes que os demais, de forma a introduzir e ilustrar o emprego de modelos como representação de configurações. Neste primeiro padrão de referência são representados o *job shop* e o celular virtual (VCM). Ambos os cenários dispõem de nove estações de trabalho (ET) de quatro tipos (A, B, C e D) contendo de duas a três máquinas por estação de trabalho, num total de vinte e uma máquinas, conforme mostrado na tabela 2.3. As estações encontram-se distribuídas no chão de fábrica afastadas entre si segundo as distâncias apresentadas na tabela 2.4.

Tabela 2.3 – Estações de trabalho consideradas neste exemplo de modelo, e as respectivas máquinas que as compõem - Adaptado: Mak e Wang (2002)

<i>Tipo de (ET)</i>	<i>Nº (ET)</i>	<i>Nº de Máquinas por (ET)</i>
A	1, 2	2
B	3, 4, 5	3
C	6, 7	2
D	8, 9	2

Tabela 2.4 - Distâncias entre estações de trabalho (ET) - fonte: Mak e Wang (2002)

<i>De: (ET)/ Para: (ET)</i>	1	2	3	4	5	6	7	8	9
1	–	6	7	3	4	4	5	5	4
2	6	–	3	3	6	2	3	1	4
3	7	3	–	4	6	5	2	2	5
4	3	3	4	–	3	3	2	2	3
5	4	6	6	3	–	6	5	5	2
6	4	2	5	3	6	–	5	3	6
7	5	3	2	2	5	5	–	2	3
8	5	1	2	2	5	3	2	–	3
9	4	4	5	3	2	6	3	3	–

Os dez pedidos de produção, com seus respectivos roteiros de fabricação, tempos de processos e tamanhos de lotes encontram-se na tabela 2.5. Por outro lado, de forma apenas ilustrativa, pensando em facilitar a visualização e comparação, foi adicionada a figura 2.5

com um possível arranjo físico. Nesta figura, a letra dentro de cada círculo representa o tipo de estação de trabalho - ET (A, B, C ou D), enquanto o número dentro do quadrado representa o número da ET (1 até 9). E cada retângulo preenchido na cor cinza representa cada uma das máquinas que cada ET contém (2 ou 3 unidades por ET).

Tabela 2.5 - Lista de Pedidos de Produção - fonte: Adaptado de Mak e Wang (2002)

<i>Pedidos</i>	<i>Roteiro de Produção</i>	<i>Tempos de Processo (min)</i>	<i>Volume de Produção (unidades)</i>
1	C ⇒ A	5 – 4	13
2	A ⇒ B ⇒ C	7 – 9 – 2	8
3	A ⇒ B ⇒ C	5 – 10 – 7	17
4	B ⇒ C ⇒ D	9 – 4 – 4	21
5	B	5	78
6	C ⇒ B	4 – 9	32
7	D ⇒ B	6 – 7	35
8	A ⇒ B ⇒ D	8 – 14 – 10	15
9	A ⇒ D	14 – 5	32
10	B ⇒ C	6 – 8	48

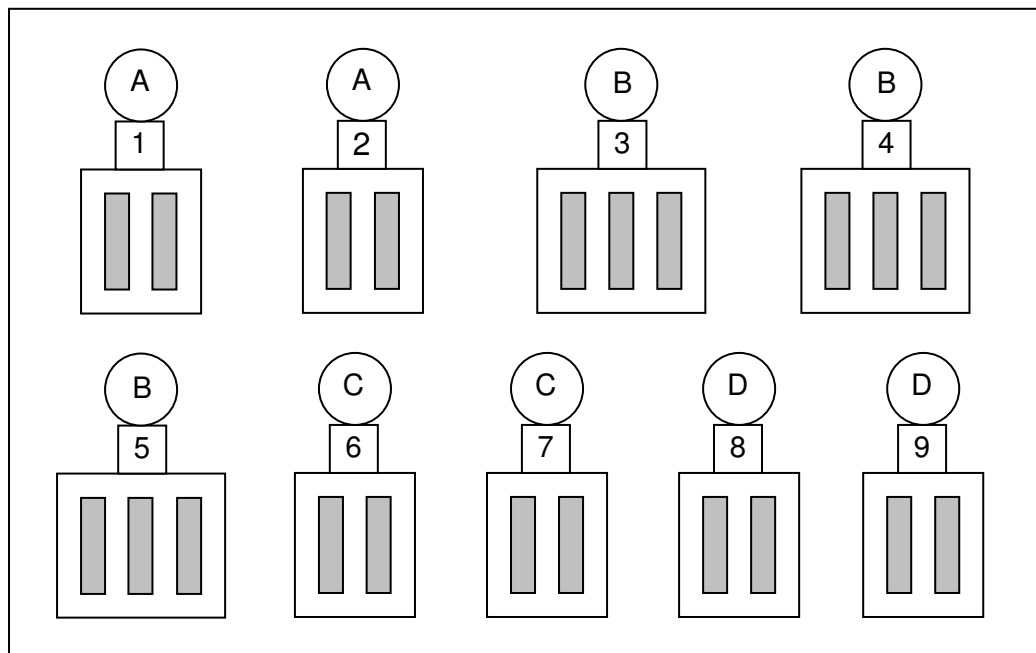


Figura 2.5 – Configuração do sistema com células virtuais

O padrão de referência seguinte, desenvolvido por Morris e Tersine (1990), e também utilizado por Kannan e Ghosh (1996), Kannan (1997a, 1997b) e Kannan (1998), é composto por oito departamentos funcionais no modelo de configuração celular virtual e cinco células no modelo de configuração celular tradicional. O autores empregam o termo departamento *job shop* com o mesmo significado que Suresh e Slomp (2005), ou seja, tal departamento é composto por um conjunto de recursos, tais como máquinas, equipamentos ou estações de trabalho, fisicamente agrupadas, que realizam processos similares. Por sua vez, uma ET corresponde a postos de trabalho constituídos por bancadas de ajustagem, montagem etc.

Tabela 2.6 – Exemplo de modelo com 5 células tradicionais – fonte: Morris e Tersine (1990)

<i>Células</i>	<i>Número da Máquina</i>
1	18, 25, 13, 3, 23, 10, 16
2	26, 2, 15, 7, 17, 4, 20, 12
3	22, 8, 28, 24, 9, 21
4	29, 14, 6, 19, 27
5	11, 1, 30, 5

A tabela 2.6 especifica a configuração do modelo proposto por Morris e Tersine (1990), que é composto por cinco células tradicionais, enquanto a tabela 2.7 apresenta a configuração do modelo composto de oito departamentos funcionais, cada um contendo três a quatro máquinas idênticas. Com essas duas tabelas tem-se uma visualização básica do padrão de referência e modelos empregados nesses trabalhos, especialmente no aspecto físico de leiaute. É ilustrativo observar como o padrão e modelos estão bem concebidos e balanceados, uma vez que as células têm no máximo uma máquina de cada tipo de processo. Assim, a célula maior, que é a segunda da tabela 2.6, é composta pelas máquinas 26, 2, 15, 7, 17, 4, 20, 12, que pertencem respectivamente aos processos 1, 2, 3, 4, 5, 6, 7 e 8 da tabela 2.7, ou seja, a máquina 26 é do tipo de processo do departamento 1, a máquina 2 é do tipo de processo do departamento 2 e assim sucessivamente. E cada modelo tem potencialmente a mesma capacidade, pois são compostas exatamente pelas mesmas máquinas.

O terceiro padrão de referência, empregado por Kannan (1997b), emprega apenas um dos modelos de leiaute do padrão anterior, o *job shop*, que pode operar dessa forma ou como célula virtual. Kannan (1997b) utilizou em seu modelo os dados da tabela 2.7.

Tabela 2.7 – Exemplo de modelo com oito departamentos funcionais – fonte: Morris e Tersine (1990)

<i>Departamento</i>	<i>Número da Máquina</i>
1	8, 18, 19, 26
2	2, 25, 27, 28
3	11, 13, 15, 24
4	1, 3, 7, 9
5	17, 21, 23, 30
6	4, 5, 10, 29
7	14, 16, 20
8	6, 12, 22

O padrão de referência empregado por Vakharia et al. (1999) tem sua aplicação para leiaute por produto (*flow shop*) e célula virtual, dentro de uma concepção *flow shop*, com muitas variações de números de estágios, máquinas etc. A figura 2.6, que representa uma das situações consideradas no trabalho de Vakharia et al. (1999), mostra que o leiaute da célula virtual corresponde a uma aplicação diferente do conceito, correspondendo à tentativa de aumentar a flexibilidade do leiaute por produto (*flow shop*). Como no presente trabalho não será feita uma comparação com o layout por produto (*flow shop*), devido à sua reduzida flexibilidade, não serão apresentados os resultados obtidos no trabalho de Vakharia et al. (1999).

O padrão de referência seguinte é empregado por Jensen (2000) para modelar e comparar um leiaute *job shop* com uma configuração híbrida e também com um leiaute celular tradicional. O primeiro, o *job shop*, é composto por quatro departamentos em linha com quatro máquinas iguais em cada, num total de 16 máquinas, conforme se pode ver na figura 2.7. Pode-se perceber que este leiaute não corresponde exatamente ao *job shop* típico, pois este não costuma ter restrições de fluxo, como neste modelo do Jensen (2000), sendo de fato um modelo *flow shop*. A configuração híbrida, representada na figura 2.8, é composta por quatro células em linha, duas a duas, formando duas linhas, tendo no total também 16 máquinas. Finalmente o celular tradicional é ilustrado na figura 2.9, o qual é formado por quatro células independentes com quatro máquinas cada, totalizando mais uma vez 16 máquinas. Esta descrição, associada às suas respectivas representações gráficas, permite auxiliar o entendimento dos modelos de configuração que compõem este padrão de referência, particularmente quanto às suas características físicas. Deve-se destacar também a inclusão dos operadores das máquinas nestes modelos.



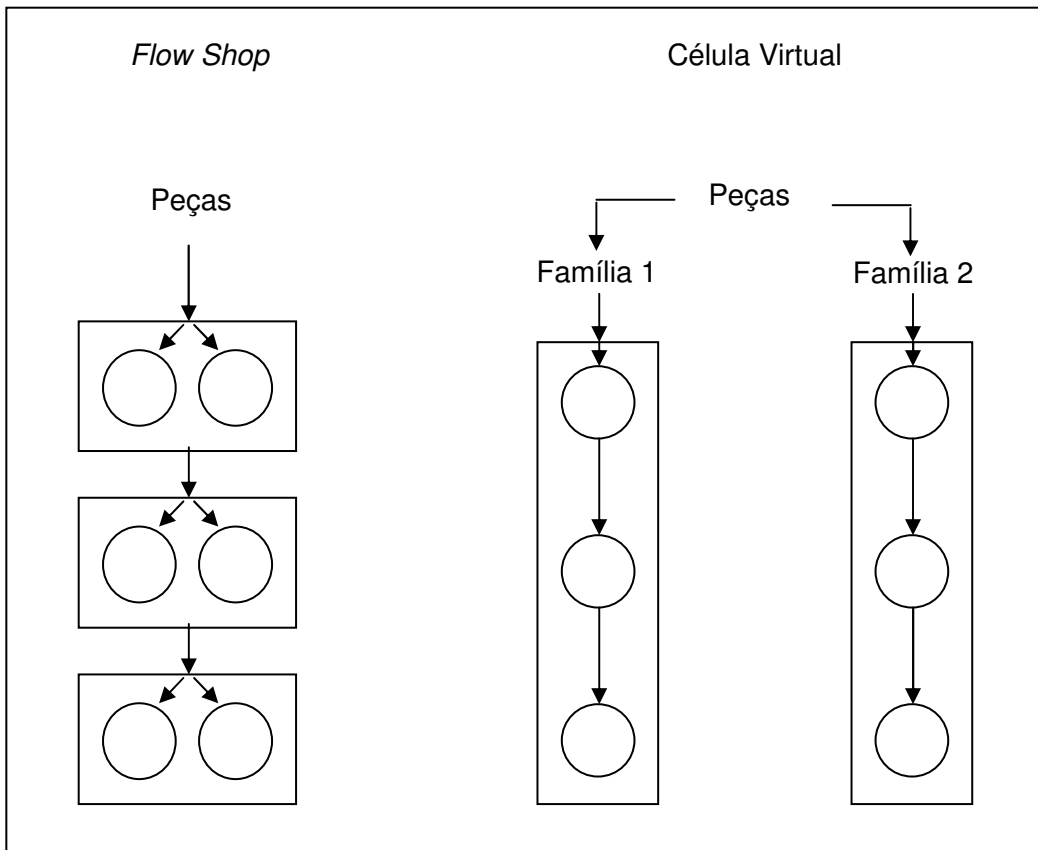


Figura 2.6 – *Flow shop* e célula virtual com 3 estágios e 2 máquinas por estágio - fonte: adaptado de Vakharia et al. (1999)

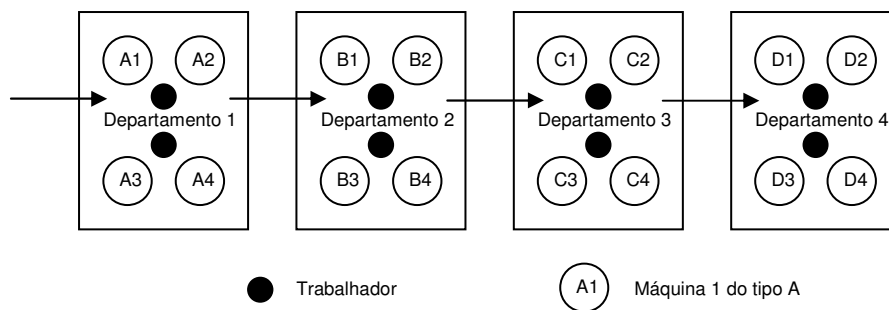


Figura 2.7 – Modelo *job shop* no padrão estabelecido por Jensen (2000)

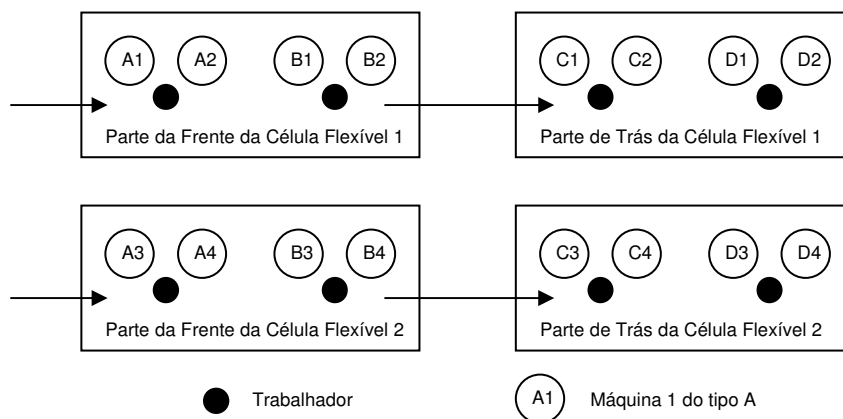


Figura 2.8 – Modelo híbrido no padrão estabelecido por Jensen (2000)

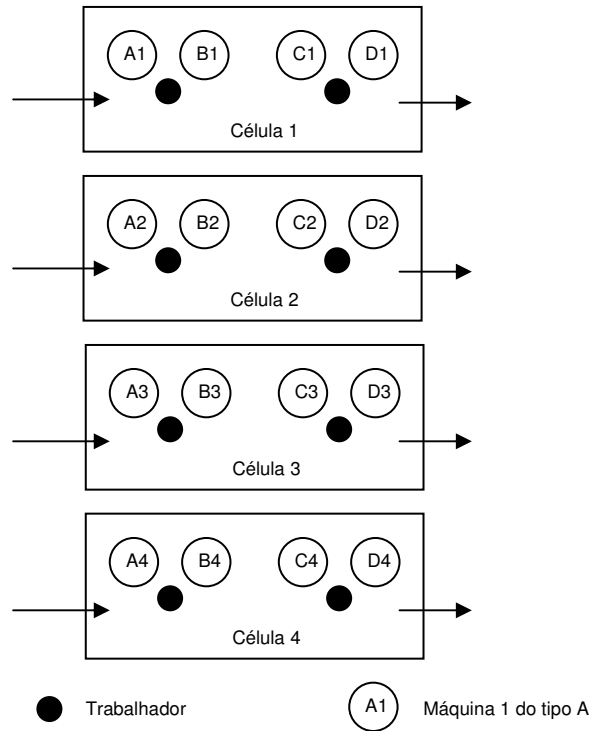


Figura 2.9 – Modelo celular tradicional no padrão estabelecido por Jensen (2000)

O último padrão de referência é o utilizado por Suresh e Slomp (2005), que é aplicado para comparar os leiautes *job shop*, células virtuais e células tradicionais. De forma idêntica ao padrão anterior, este padrão também leva em conta os operadores das máquinas. A figura 2.10 mostra os modelos de leiautes empregados neste padrão, onde os números dentro dos arranjos representam as máquinas e as letras os operadores. O modelo de leiaute à esquerda na figura 2.10 é típico do *job shop*, sendo constituído por dez departamentos, cada qual com três ou quatro máquinas, totalizando 31 máquinas. Este modelo de leiaute é empregado para a simulação tanto da configuração *job shop* como das células virtuais. O modelo à direita na figura 2.10 corresponde a um leiaute celular tradicional, constituído por cinco células, cada uma com seis ou sete máquinas diferentes, totalizando também 31 máquinas.

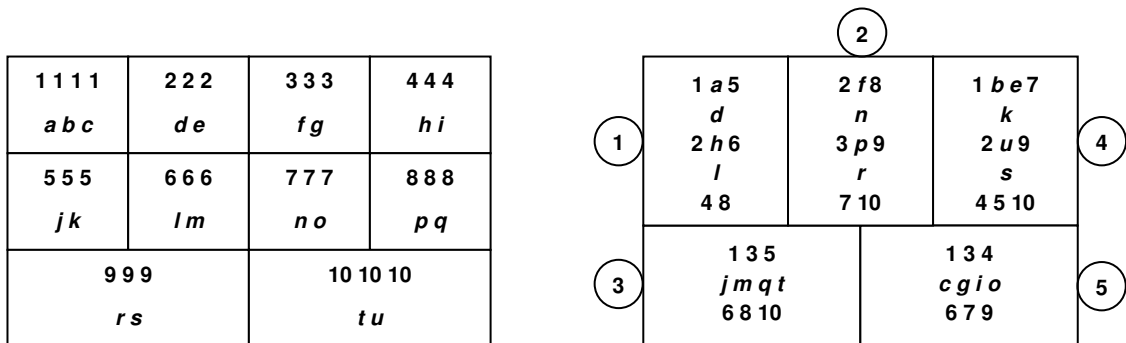


Figura 2.10 – Modelos *Job Shop* e Celular Tradicional no padrão de Suresh e Slomp (2005)

## 2.2 Simulação das operações de manufatura

Grande parte dos trabalhos de pesquisa que realizam experimentos em ambientes de gestão industrial emprega a simulação como ferramenta para facilitar esta tarefa. Este item descreve a técnica de simulação, o emprego da simulação combinada com o projeto de experimentos, além de mencionar os softwares de simulação disponíveis.

### 2.2.1. Técnica de simulação

Segundo Maria (1997), a simulação de um sistema é a execução de um modelo do sistema, entendido como uma representação da construção e funcionamento desse sistema. A simulação de um modelo permite entender a dinâmica de um sistema assim como analisar e prever o efeito de mudanças que se introduzam no mesmo. É uma representação próxima da realidade, e será tanto mais real quanto mais características significativas do sistema seja capaz de representar. Por outro lado, o modelo deve ser simples, de forma que não se torne demasiado complexo para se construir, mas ao mesmo tempo o modelo deve ser o mais fiel possível ao sistema real. O uso de modelos traz enormes benefícios no planejamento e análise de um sistema de gestão de manufatura. Segundo Freitas Filho (2001) as principais vantagens e desvantagens do uso da simulação são:

#### Vantagens:

- Um modelo criado pode ser utilizado inúmeras vezes;
- Não depende da disponibilidade do sistema real para a realização de ensaios, nem incorre em custos para executar os experimentos.
- Pode conter simplificações com relação aos sistemas reais para que facilitem a modelagem e torne mais fácil a compreensão dos parâmetros abordados, desde que os elementos desprezados não interajam com os parâmetros estudados com relação aos resultados a serem avaliados.
- A simulação é mais fácil de aplicar que os modelos analíticos, que por sua vez devem ainda ser muito simplificados para tornarem-se viáveis matematicamente.
- Podem ser quase tão detalhados como os sistemas reais, permitindo realizar ensaios num cenário muito similar ao real sem o ônus e o desgaste de ter de interferir na rotina dos sistemas reais. Permite também um estudo detalhado de todas as operações e características do sistema.

#### Desvantagens:

- A construção de modelos exige treinamento e experiência prévia.

- Nem sempre a variabilidade de um sistema é bem captada e modelada, podendo levar a resultados equivocados. Entretanto, algumas soluções, tais como introdução de geradores de números aleatórios, podem ajustar o modelo de forma a representar bem a variabilidade.
- A construção de modelos consome muito tempo. Tentativas de reduzir este tempo por via de simplificação do modelo podem levar a resultados insatisfatórios.

Para o desenvolvimento e conclusão do modelo de simulação, segundo Freitas Filho (2001), quatro etapas são necessárias: planejamento, modelagem, experimentação e conclusão. A figura 2.11 ilustra essas diversas etapas, sendo que para efeito de aprofundamento são abordadas neste item apenas dois componentes da etapa de modelagem, por serem os mais significativos e demandarem maior esforço e mais tempo de desenvolvimento, que são a tradução do modelo e a verificação e validação do modelo.

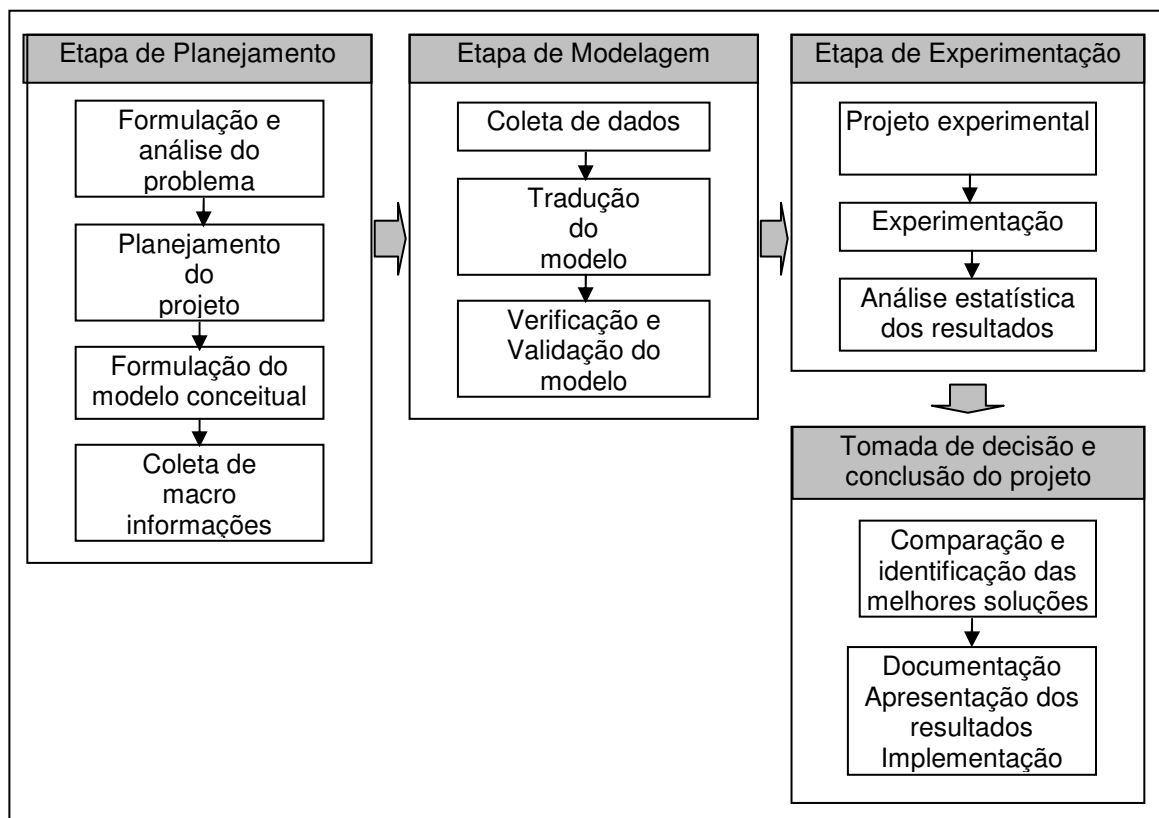


Figura 2.11 – Etapas da simulação - fonte: Freitas Filho (2001)

A tradução do modelo consiste em codificar o modelo na linguagem de simulação adequada. Além da linguagem é necessário o conhecimento de conceitos associados aos

sistemas e à concepção de cada linguagem de simulação em particular, tais como variáveis de estado, eventos, entidades, atributos, *templates*, recursos e filas entre outros.

Uma vez concluído o modelo, é necessário efetuar a sua verificação e validação. A técnica de verificação procura garantir que o programa computacional esteja isento de erros de sintaxe e de lógica, enquanto a técnica de validação tem como objetivo assegurar que o modelo tenha um comportamento semelhante ao do sistema modelado, permitindo a realização de inferências estatísticas válidas e confiáveis a partir do modelo. Segundo Maria (1997), a validação do modelo é uma questão muito importante que conta com diversas técnicas, dentre as quais se inclui a simulação do modelo sob condições de entrada conhecidas, para comparar as saídas do modelo com os dados reais conhecidos. Freitas Filho (2001) indica as seguintes técnicas de verificação e validação do modelo:

#### Técnicas de Verificação:

- Uso de modelos determinísticos
- Variações sobre os dados de entrada
- Uso de rotinas de rastreamento (*trace*)
- Testes de continuidade
- Testes de degenerescência<sup>1</sup>
- Execução de casos simplificados
- Verificação de consistência
- Independência de sementes geradoras de números aleatórios
- Rotinas de verificação

#### Técnicas de Validação:

- Conhecimento e intuição de especialistas
- Medições obtidas em sistemas reais
- Resultados Teóricos

Sabendo-se que experimentar com o sistema real é dispendioso, Freitas Filho (2001) cita que pode-se reduzir sobremaneira o tempo necessário para a execução de um

---

<sup>1</sup> Verificação do funcionamento de um modelo quando, sobre os parâmetros, aplicam-se valores extremos (menores ou maiores) que os permitidos (FREITAS FILHO 2001).

experimento completo de simulação. Law (2008) indica sete etapas para a validade de um experimento levado a cabo via simulação, as quais são listadas abaixo:

1. Formular o problema
2. Coletar informações/dados e formular um documento com os pressupostos
3. Validar o documento com os pressupostos
4. Programar o modelo
5. Validar o modelo
6. Desenvolver, executar e analisar os experimentos
7. Documentar e apresentar os resultados

Sánchez (2007) sintetiza a interação do experimento com o sistema real no esquema apresentado na figura 2.12.

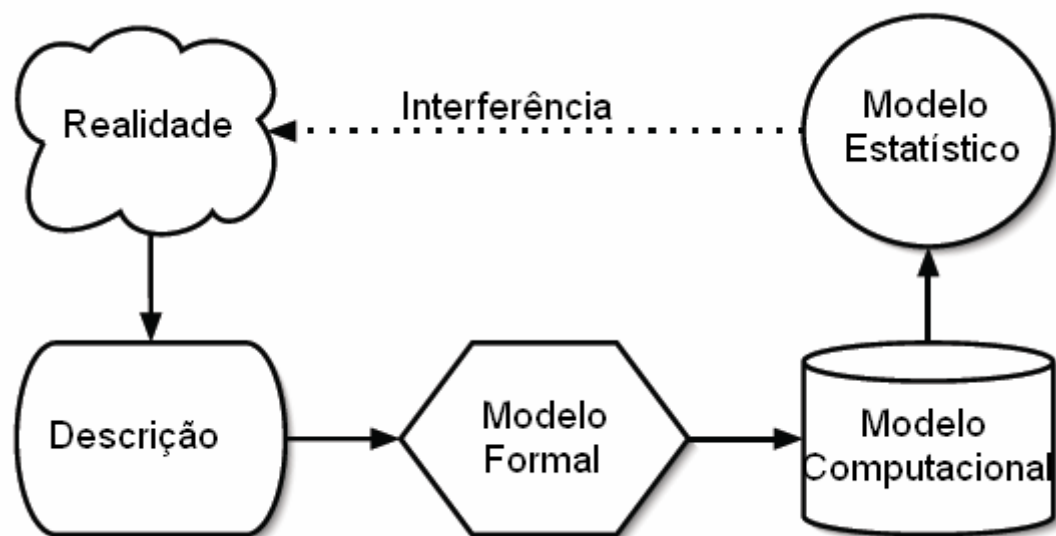


Figura 2.12 – Esquema do estudo de simulação aplicado a um experimento - fonte: adaptado de Sánchez (2007)

No caso do passo 6 (isto é, desenvolver, executar e analisar os experimentos), deve-se identificar quais os fatores e variáveis resposta a simulação pode auxiliar a estudar. Law e McComas (1997) respondem a esta questão afirmando que a simulação deve abordar em primeiro lugar as necessidades de recursos, entre eles: (a) a quantidade e os tipos de máquinas necessários para uma determinada situação; (b) quantidade, tipo e localização de equipamentos de transporte; (c) localização e dimensões de áreas de armazenamento; (d) avaliação da mudança de volumes e “mix de produção”, ou seja, a combinação concreta das quantidades e dos tipos de produtos a serem fabricados. Em segundo lugar, com relação às

medidas de desempenho a serem empregadas, Law e McComas (1997) e Carson II indicam: (a) a análise de “*throughput*”, ou seja, o tempo total para produzir toda a ordem de produção; (b) a análise do período de tempo dos produtos no sistema durante a simulação; e (c) a análise de gargalos de produção. Por último, eles afirmam que também devem ser avaliadas pela simulação as estratégias operacionais, tais como o gerenciamento da seqüência de produção; políticas de estoque; planos de confiabilidade na área de manutenção de máquinas e equipamentos; controle de qualidade etc. Segundo Law e McComas (1997), a simulação é uma técnica segura e eficaz para se estudar o efeito dos principais parâmetros envolvidos nos sistemas de manufatura.

### 2.2.3. Instrumentos de software de simulação disponíveis

O emprego de modelos de simulação para análise de sistemas reais é muito antigo. Inicialmente a abordagem era exclusivamente através de algoritmos matemáticos, particularmente no âmbito da pesquisa operacional. A teoria das filas e o método de Monte Carlo são exemplos dessas aplicações. O emprego generalizado da informática possibilitou o desenvolvimento de diversas ferramentas de software aplicadas à simulação.

Law e McComas (1997) abordam os diferentes instrumentos de software de simulação disponíveis. Os softwares tradicionais de simulação são linguagens especializadas voltadas para modelar e simular sistemas. Eles permitem desenvolver programas especializados para cada aplicação particular. Esses programas de simulação, como os demais programas de linguagens tradicionais, se constituem de linhas de comando ou instruções codificadas segundo uma sintaxe específica. Assim, o emprego dessas linguagens no processo de modelagem exige um profundo conhecimento da sintaxe e grande experiência em programação. Um exemplo desse tipo de software é a linguagem de simulação Siman (PEGDEN et al. 1995).

Entretanto, nos últimos anos foram desenvolvidos softwares mais amigáveis, com interface gráfica para a construção de modelos. Muitos deles são orientados especificamente à simulação de manufatura, exigindo pouca ou nenhuma programação, antes empregam ícones que representam recursos (trabalhadores, máquinas etc), entidades (clientes, peças etc), filas etc. Dentre estes destacam-se os softwares AutoMod, ProModel, Taylor, e Witness. O software Arena, apesar de ser habitualmente classificado no grupo das linguagens de simulação, por estar efetivamente alicerçado na linguagem Siman e por ser de uso genérico, possui também uma interface gráfica que dispensa a programação. A tabela 2.8 apresenta alguns exemplos desses dois tipos de software de simulação.

Tabela 2.8 – Exemplos de software de simulação - fonte: adaptado de Maria (1997)

TIPO	EXEMPLOS
Linguagens de Simulação	Arena (anteriormente SIMAN). AweSim! (anteriormente SLAM II), Extend, GPSS, Micro Saint, SIMSCRIPT, SLX <i>Software Orientado a Objeto:</i> MODSIM III, SIMPLE-++ <i>Software de Animação:</i> Proof Animation
Simuladores	<i>Orientados à Manufatura.</i> AutoMod. Extend+MFG, FACTOR/AIM, ManSim/X, MP\$IM, ProModel, QUEST, Taylor II WITNESS <i>Orientados à Comunicações/computadores.</i> COMNET III, NETWORK II.5, OPNET Modeler, OPNET Planner, SES/Strategizer, SES/workbench <i>Orientados aos Negócios:</i> BP\$IM, Extend+BPR, ProcessModel, ServiceModel, SIMPROCESS, Time machine <i>Orientado à Medicina:</i> MedModel

### 2.3 Método empregado na geração da seqüência de produção

Dentre os trabalhos de pesquisa que desenvolvem comparações entre os tipos de processos de manufatura mediante simulação combinada com projeto de experimentos, têm-se os publicados por Kannan e Ghosh (1996), Kannan (1997a, 1997b e 1998) e Suresh e Slomp (2005), os quais utilizam regras de prioridade para gerar a seqüência de produção. Por esse motivo, esse tema é abordado a seguir.

Segundo Slack et al. (2002), o seqüenciamento de produção, é o estabelecimento da seqüência em que as tarefas deverão ser executadas. Conjuntos predefinidos de regras podem ser utilizados para estabelecer a prioridade em que essas tarefas devem ser executadas. Dentre esses conjuntos de regras pode-se citar: Data Prometida, LIFO, FIFO e operação mais longa ou tempo total mais longo da tarefa.

Para ilustrar estas regras, a título de exemplo, considerem-se cinco tarefas segundo a ordem de chegada: A, B, C, D e E, com tempos de processamento respectivamente 5, 3, 6, 2 e 1 dias e prazo de entrega respectivamente de 6, 5, 8, 7 e 3 dias. As tabelas 2.9 a 2.12 mostram a aplicação das quatro regras de sequenciamento citadas.

A regra Data Prometida (“*due date*”) consiste em priorizar a seqüência de execução das tarefas pendentes em função das datas comprometidas para a entrega das tarefas. A tabela 2.9 ilustra a aplicação desta regra para gerar uma seqüência.



Tabela 2.9 - Seqüenciamento por data prometida - fonte: Slack et al. (2002)

<i>Seqüência de tarefas</i>	<i>Tempo de Processamento (dias)</i>	<i>Início (dia)</i>	<i>Término (dia)</i>	<i>Data Prometida (dia)</i>	<i>Atraso (dias)</i>
E	1	0	1	3	0
B	3	1	4	5	0
A	5	4	9	6	3
D	2	9	11	7	4
C	6	11	17	8	9
Atraso total =					16

O regra LIFO (“*Last In First Out*”, ou último a chegar, primeiro a sair) consiste em priorizar a ordem de execução das tarefas pendentes em função da ordem inversa da chegada. A tabela 2.10 ilustra a aplicação desta regra.

Tabela 2.10 – Seqüenciamento por LIFO - fonte: Slack et al. (2002)

<i>Seqüência de trabalhos</i>	<i>Tempo de Processamento (dias)</i>	<i>Início (dia)</i>	<i>Término (dia)</i>	<i>Data Prometida (dia)</i>	<i>Atraso (dias)</i>
E	1	0	1	3	0
D	2	1	3	7	0
C	6	3	9	8	1
B	3	9	12	5	7
A	5	12	17	6	11
Atraso total =					19

A regra FIFO (“*First In First Out*”, ou primeiro a chegar, primeiro a sair, ou ainda FCFS – “*First Come, First Served*”) é uma regra que consiste em priorizar a ordem de execução das tarefas pendentes em função da ordem de chegada (ver tabela 2.11).

Tabela 2.11 – Seqüenciamento por FIFO - fonte: Slack et al. (2002)

<i>Seqüência de trabalhos</i>	<i>Tempo de Processamento (dias)</i>	<i>Início (dia)</i>	<i>Término (dia)</i>	<i>Data Prometida (dia)</i>	<i>Atraso (dias)</i>
A	5	0	5	6	0
B	3	5	8	5	3
C	6	8	14	8	6
D	2	14	16	7	9
E	1	16	17	3	14
Atraso total:					32

A regra Operação mais curta (ou tempo total mais curto da primeira tarefa), também conhecida como Menor Tempo de Processamento (SOT – “*Shortest Operation Time First*” ou SPT - “*Shortest Processing Time*”), consiste em priorizar a execução das tarefas pendentes que tenham um menor tempo de processo. Esta regra aumenta o fluxo de produção, se este for medido como o número de ordens completas em um determinado período, e eventualmente aumenta o fluxo de caixa por acelerar o faturamento. A tabela 2.12 ilustra esta seqüência.

Tabela 2.12 – Seqüenciamento por SPT - fonte: Slack et al. (2002)

<i>Seqüência de trabalhos</i>	<i>Tempo de Processamento (dias)</i>	<i>Início (dia)</i>	<i>Término (dia)</i>	<i>Data Prometida (dia)</i>	<i>Atraso (dias)</i>
E	1	0	1	3	0
D	2	1	3	7	0
B	3	3	6	5	1
A	5	6	11	6	5
C	6	11	17	8	9
Atraso total =					15

Essas quatro regras de prioridade são as mais utilizadas, mas como há muitas outras, com o intuito de evitar que o escopo deste tópico fique limitado, serão citadas brevemente ainda mais duas importantes regras, a saber, Folga da Ordem (“*Remaining Slack Time*”) e Quociente da Folga sobre as Operações Restantes. Segundo Ferreira (2000), a regra baseada na Folga da Ordem corresponde à diferença entre o tempo restante até a data prometida para a entrega do serviço (“*due date*”) e o tempo de processamento (incluindo

*setup*) necessário para a conclusão da ordem; quanto menor a folga maior a prioridade. Ainda segundo este mesmo autor, a regra baseada no Quociente da Folga sobre as Operações Restantes é semelhante à anterior, mas ao invés de considerar apenas a diferença entre folga e processamento, ela considera o quociente entre a folga e a quantidade de operações restantes. Quanto menor o quociente, isto significa que a ordem está atrasada. Portanto, a prioridade é dada para as ordens com o menor quociente.

A Tabela 2.13 apresenta as regras de prioridade empregadas em trabalhos de pesquisa que desenvolvem comparações entre tipos de processos de manufatura com configurações distintas. Dentre essas regras, algumas se referem diretamente ao seqüenciamento de tarefas a serem processadas, ou seja, estabelece qual é o próximo lote de peças a serem processadas em uma determinada máquina ou célula. Outras regras fazem isto de maneira indireta, ao estabelecerem uma associação temporária entre máquinas disponíveis e famílias de peças a serem processadas, ou seja, cria-se uma nova célula virtual voltada a atender vários lotes de peças que estão na fila, todos classificados como da mesma família, mas a definição de qual desses lotes será processado primeiro será definido por outra regra de prioridade.

Tabela 2.13 – Trabalhos que comparam células virtuais, tradicionais e *job shop*

<i>Trabalho</i>	<i>Sistemas Comparados</i>	<i>Regras de Prioridade Empregadas</i>
KANNAN e GHOSH 1996	Celular Virtual Celular Tradicional <i>Job Shop</i>	Tarefa idêntica à anterior. Folga da Ordem. Número de tarefas iguais na fila. Tarefas que exigem células menores.
KANNAN 1997	Celular Virtual Celular Tradicional	Tarefa pertencente a uma família que ainda não tem acesso ao tipo de máquina liberada.
KANNAN 1997b	Celular Virtual <i>Job Shop</i>	Tarefa idêntica à anterior. Folga da Ordem.
KANNAN 1998	Celular Virtual Celular Tradicional	Tarefas que não alterem o <i>setup</i> da máquina. FCFS (“ <i>First Come, First Served</i> ”)
SURESH e SLOMP 2005	Celular Virtual Celular Tradicional <i>Job Shop</i>	FCFS (“ <i>First Come, First Served</i> ”) SPT (“ <i>Shortest Processing Time</i> ”)

## 2.4 Emprego de DOE em combinação com simulação

Neste item é descrito sucintamente o método que combina o emprego de projeto de experimentos (DOE) com a simulação. Também descreve-se neste item como são avaliados os resultados obtidos através deste método.

O subitem 2.4.1 descreve o emprego do projeto de experimentos. O escopo dos fatores e níveis a serem considerados são tratados no subitem 2.4.2, enquanto o escopo das variáveis resposta a serem utilizadas no projeto de experimentos é descrito no subitem 2.4.3. No subitem 2.4.4 é abordada a integração da planilha de dados Microsoft Excel® com o software de simulação, particularmente em termos de entrada e saída de dados e a sua influência no projeto de experimentos. O subitem 2.4.5 encerra o presente capítulo descrevendo a metodologia empregada no tratamento dos dados de saída com o auxílio do software MINITAB.

### 2.4.1. Projeto de experimentos (DOE – *Design of Experiments*)

O projeto de experimentos, segundo Sanchez (2008), é um método testador que emprega fatores, que são variáveis independentes, para testar o impacto nas saídas dos experimentos, que são as variáveis resposta. A técnica consiste de testar uma seqüência de experimentos em que são feitas alterações simultâneas em diversos fatores – segundo determinados níveis, tais como, alto, médio e baixo - de modo que se possa avaliar os possíveis efeitos dessas alterações nas variáveis resposta envolvidas no processo. Dentre os benefícios desta metodologia tem-se a compreensão com maior profundidade das relações entre os parâmetros do projeto (os fatores analisados) e a variação de desempenho nos processos (as variáveis resposta). Este método também permite avaliar os efeitos dos ruídos, que são interferências difíceis de avaliar num processo, como por exemplo uma vibração em uma máquina, ou a falta de precisão de uma medida, e dessa maneira reduzir variações indesejáveis no processo. Como os ruídos podem mascarar os resultados, a análise de variância (ANOVA) consegue identificar estatisticamente se o efeito dos fatores é significativo nas variações de desempenho do processo (variáveis resposta), ou se o nível dos erros de medição e dos ruídos em geral tem influência significativa nos resultados.

Além de escolher os fatores a serem testados, o responsável pela elaboração do projeto de experimentos deve estabelecer os níveis destes fatores, bem como eleger as variáveis resposta. É necessário por exemplo definir o número de replicações que cada experimento terá, isto é, quantas vezes será executado cada ensaio do experimento, com os mesmos níveis de cada um dos fatores. É necessário também estabelecer um nível de significância (por exemplo, 5%) que possa ser utilizado na análise de variância para verificar

se as variações nos resultados são estatisticamente significativas. Para isso é necessário planejar os experimentos de forma a obter os efeitos desejados. Deve-se decidir ainda se deve-se usar experimentos fatoriais completos (ou seja, a combinação completa de todos os fatores em todos os níveis) ou se o fatorial será fracionado, ou ainda se será aplicada uma série ortogonal, que permite identificar os principais efeitos com um mínimo de experimentações.

Caso seja decidido empregar um plano de experimento com fatorial completo, considerando  $k$  fatores, com  $n$  níveis para cada fator, para testar todas as combinações de fatores serão necessários  $n^k$  experimentos, que é geralmente um número bastante elevado. A título de exemplo, considerando um planejamento de experimento com apenas dois fatores e três níveis cada um, ter-se-á nove experimentos. Mas se forem quatro fatores a três níveis cada um, será necessário realizar oitenta e um experimentos.

Quando se trabalha com muitos fatores, uma solução para reduzir o número de experimentos necessários é usar um fatorial com série ortogonal. Neste caso também serão considerados os  $k$  fatores com  $n$  níveis cada um. A série ortogonal não realiza todas as combinações, mas testa todos os níveis de cada fator de uma maneira equilibrada, correspondendo a um número de experimentos igual a  $1+k(n-1)$ . Assim, para um experimento com quatro fatores e três níveis cada, ter-se-ão apenas nove experimentos, isto é,  $(1+4(3-1))$ .

Para ilustrar esses conceitos, um exemplo extraído de Montgomery (1997) é apresentado, no qual deseja-se pesquisar qual é o efeito da temperatura e do percentual de carbono na dureza de um determinado material. São ensaiadas as amostras do material para dois valores de temperatura, 900° C e 1100° C (níveis 1 e 2 respectivamente), e para dois valores de concentração de carbono, 5% e 15% (níveis 1 e 2 respectivamente). São feitos três ensaios para cada combinação de temperatura e percentual de carbono. A tabela 2.14 apresenta os resultados obtidos neste ensaio.

Tabela 2.14 – Resultados do ensaio de dureza para o exemplo considerado - fonte: Montgomery (1997)

<i>EXPERIMENTO</i>	<i>TEMPERATURA (nível)</i>	<i>%CARBONO (nível)</i>	<i>DUREZA – REPLICAÇÃO R1</i>	<i>DUREZA – REPLICAÇÃO R2</i>	<i>DUREZA – REPLICAÇÃO R3</i>
1	1	1	18	20	22
2	1	2	42	43	35
3	2	1	31	31	28
4	2	2	51	53	52

Pode-se verificar que este experimento emprega uma só variável resposta que é a dureza do material. Dois fatores são utilizados - temperatura e percentagem de Carbono - ambos com dois níveis, o primeiro com valores 900° C e 1100° C e o segundo com 5% e 15%. Foi adotado o padrão de três replicações por cada combinação de ensaio, R1, R2 e R3. O fatorial empregado é o completo, e como são apenas dois fatores com dois níveis cada, tem-se  $n^k$  igual a  $2^2$ , ou seja, são necessárias quatro combinações de experimentos.

A tabela 2.15 apresenta o cálculo da análise de variância dos resultados do ensaio de dureza. O valor da distribuição  $F$  é calculado pela razão entre o média quadrada do fator e do erro, obtidas, das somas dos respectivos quadrados dos desvios ( $SQ$ ) divididas pelos seus correspondentes graus de liberdade ( $gl$ ). O valor de  $F$  é obtido então pela equação 2.1:

$$F = \frac{MQ_{fator}}{MQ_{erro}} \quad (2.1)$$

Onde:  $MQ_{fator}$  e  $MQ_{erro}$  são respectivamente as médias quadradas do fator e do erro

Tabela 2.15 – Análise de variância para o experimento considerado - ensaio de dureza - fonte: Montgomery (1997)

	$SQ$	$gl$	$MQ$	$F$	FATOR AFETA ESTATISTICAMENTE?
% Carbono	1323	1	1323	196,00	Sim
Temperatura	363	1	363	53,78	Sim
Interação	3	1	3	0,44	Não
Erro	54	8	6,75		
Total	1743	11			
					F crítico: 5,317645

O valor  $F$  foi calculado para cada um dos fatores isoladamente e também para o valor de interação entre eles. O valor de  $F$  crítico pode ser obtido da tabela, no caso para 5% de nível de significância, grau de liberdade ( $gl$ ) do fator ou interação, no caso  $gl=1$ , e grau de liberdade do erro, no caso  $gl=8$ . Se  $F$  for maior que  $F$  crítico, isto indica que o fator (ou interação entre os fatores) em questão influi significativamente na variável resposta. No presente exemplo, tanto a temperatura do forno como o percentual de carbono no material influem na dureza do material, porém o mesmo não se pode afirmar com relação à interação entre esses dois fatores.

A tabela 2.16 apresenta um exemplo de série ortogonal, que é aplicado para treze fatores e três níveis. Caso se optasse por realizar o fatorial completo, o número de

experimentos necessário seria igual a  $3^{13}$ , ou seja, 1.594.323 experimentos. Com a opção do emprego da série ortogonal este número é reduzido para apenas 27 experimentos.

Tabela 2.16 – Série ortogonal para 13 fatores e 3 níveis

SÉRIES ORTOGONAIS													
NÍVEIS (A, B, C, ..., M) e FATORES (1, 2 e 3)													
Experimento	A	B	C	D	E	F	G	H	I	J	K	L	M
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	2	2	2	2	2	2	2	2	2
3	1	1	1	1	3	3	3	3	3	3	3	3	3
4	1	2	2	2	1	1	1	2	2	2	3	3	3
5	1	2	2	2	2	2	2	3	3	3	1	1	1
6	1	2	2	2	3	3	3	1	1	1	2	2	2
7	1	3	3	3	1	1	1	3	3	3	2	2	2
8	1	3	3	3	2	2	2	1	1	1	3	3	3
9	1	3	3	3	3	3	3	2	2	2	1	1	1
10	2	1	2	3	1	2	3	1	2	3	1	2	3
11	2	1	2	3	2	3	1	2	3	1	2	3	1
12	2	1	2	3	3	1	2	3	1	2	3	1	2
13	2	2	3	1	1	2	3	2	3	1	3	1	2
14	2	2	3	1	2	3	1	3	1	2	1	2	3
15	2	2	3	1	3	1	2	1	2	3	2	3	1
16	2	3	1	2	1	2	3	3	1	2	2	3	1
17	2	3	1	2	2	3	1	1	2	3	3	1	2
18	2	3	1	2	3	1	2	2	3	1	1	2	3
19	3	1	3	2	1	3	2	1	3	2	1	3	2
20	3	1	3	2	2	1	3	2	1	3	2	1	3
21	3	1	3	2	3	2	1	3	2	1	3	2	1
22	3	2	1	3	1	3	2	2	1	3	3	2	1
23	3	2	1	3	2	1	3	3	2	1	1	3	2
24	3	2	1	3	3	2	1	1	3	2	2	1	3
25	3	3	2	1	1	3	2	3	2	1	2	1	3
26	3	3	2	1	2	1	3	1	3	2	3	2	1
27	3	3	2	1	3	2	1	2	1	3	1	3	2

#### 2.4.2. Emprego de software estatístico

Existe uma enorme gama de software estatísticos, desde os específicos, ou seja, aqueles voltados a uma só aplicação, como por exemplo o controle estatístico de processo (CEP), como os genéricos, ou seja, aqueles que executam a maioria dos cálculos estatísticos habituais. A grande maioria corresponde a softwares comerciais. Segundo Curral (1994), os softwares genéricos quase sempre tiveram a sua origem com ênfase em uma área. Alguns exemplos de tais softwares são: o BMDP, que foi desenvolvido originalmente para a área biomédica; o GLIM para estatísticos; o Genstat, para a área de

agricultura; o Minitab para professores de estatística; o Systat para psicólogos, e o SPSS para as Ciências Sociais. Neste subitem será descrita a característica básica de um dos principais softwares estatísticos, a saber, o Minitab, o qual é utilizado neste trabalho.

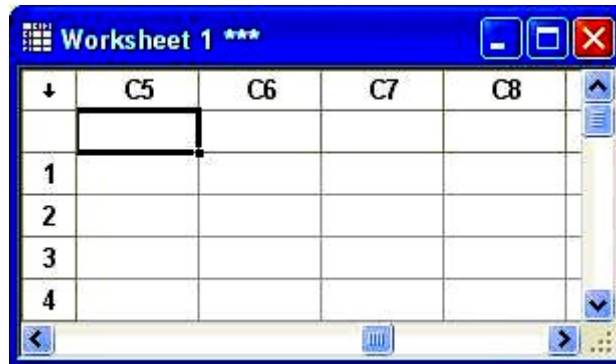


Figura 2.13 – Módulo de Entrada de Dados do software Minitab

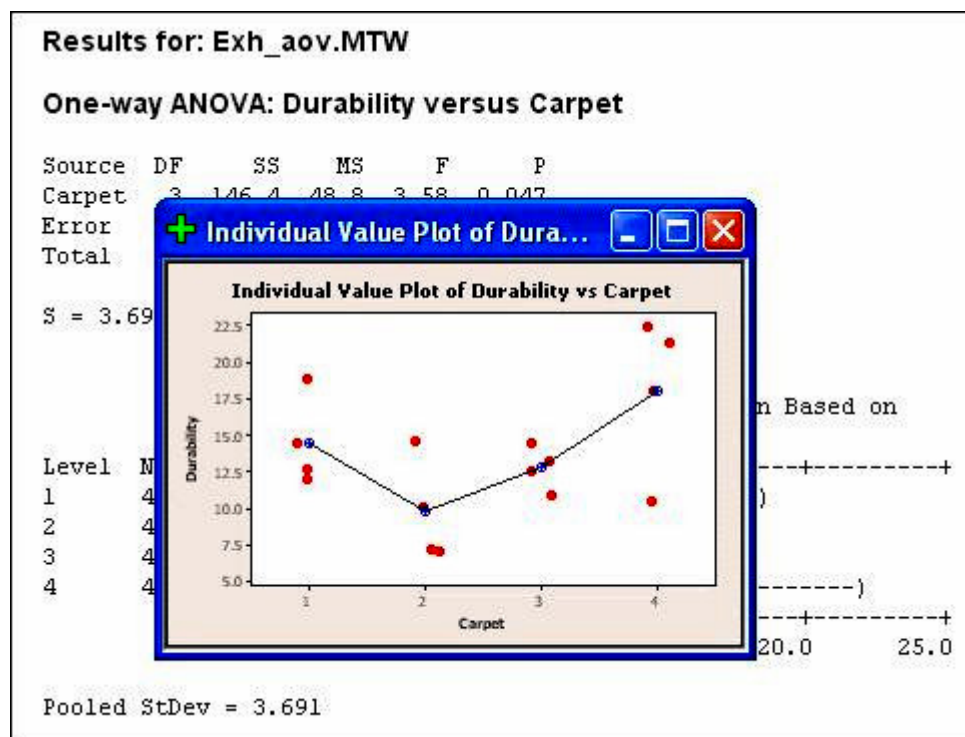


Figura 2.14 – Resultados de uma análise de variância do software Minitab

O software Minitab é um sistema computacional comercial que foi projetado por professores com o objetivo de ser útil ao ensino. Ele é adequado para a execução da análise de variância (ANOVA). O software suporta conectividade ODBC (*Open DataBase Connectivity*), que permite a obtenção de dados de bases como Access e Paradox. Os resultados gráficos podem ser salvos nos formatos TIFF, PNG e JPEG, dispensando o uso



de programas intermediários para o tratamento de imagens, caso se queira utilizar as saídas em algum processador de texto particular, tornando-se também bastante apropriado para envio de análises pela Internet. A figura 2.13 exemplifica o módulo de entrada de dados, e a figura 2.14 apresenta os resultados de uma análise de variância.

## 2.5 Conclusões e Comentários

No presente capítulo foi feita uma revisão do estado da arte dos principais temas que se relacionam com o trabalho de tese. Assim, para a revisão do primeiro âmbito relacionado, a saber, os leiautes associados ao tipo de processo por lote de fabricação, foram descritos os tipos de processos de manufatura, ou seja, projeto, *Jobbing*, lotes, em massa e contínuo. A seguir foram abordados os leiautes empregados nesses processos, sendo aprofundadas as características dos leiautes posicional, por processo, celular tradicional e por produto. Na seqüência, fez-se uma revisão dos modelos utilizados pela academia. Além disso, foram identificados, classificados e especificados os modelos utilizados nos trabalhos realizados por Drolet (1989), Morris e Tersine (1990), Kannan e Ghosh (1996), Kannan (1997, 1997b e 1998), Vakharia et al. (1999), Jensen (2000), Mak e Wang (2002) e Suresh e Slomp (2005).

O segundo âmbito relacionado à tese é a simulação das operações de manufatura com os modelos que representam as configurações escolhidas. Para tanto, foram abordadas as características da simulação e as correspondentes técnicas de verificação e validação dos modelos. Na seqüência, mencionou-se os softwares de simulação disponíveis.

O âmbito seguinte corresponde ao modo de gerar seqüências de produção, mediante o uso de regras de prioridade, com ênfase as regras Data Prometida, LIFO, FIFO e Menor Tempo de Processamento (SPT).

O último âmbito relacionado à tese abordado neste capítulo corresponde à técnica empregada no tratamento dos resultados obtidos com auxílio da simulação. Para tanto, descreve-se inicialmente o emprego do projeto de experimentos em combinação com simulação, seus elementos e características, tais como fatores, variáveis resposta, replicações, tipos de fatoriais etc. O capítulo de encerra com um levantamento dos softwares estatísticos empregados nos trabalhos científicos, destacando e descrevendo em particular algumas características do software Minitab, o qual é utilizado no presente trabalho.

## CAPÍTULO 3 PROCEDIMENTOS METODOLÓGICOS

Neste capítulo descreve-se a metodologia empregada neste trabalho de tese visando atingir o objetivo proposto, que consiste em comparar, mediante simulação e projeto de experimentos, o desempenho das operações de manufatura nas configurações *job shop*, celular tradicional e celular virtual. Dessa forma, busca-se avaliar as vantagens da aplicação deste último tipo de configuração com relação aos demais. Esta metodologia também está em conformidade com as conclusões do capítulo anterior, no qual foi apresentado o estado da arte das pesquisas relativas às células virtuais.

### 3.1 Padrão e Modelos Adotados

Neste item descreve-se o padrão e os modelos selecionados como recursos metodológicos para modelar e simular as operações de manufatura nas configurações empregadas nos processos tipo Lote, as quais são estudadas neste trabalho de tese, isto é, as configurações *job shop*, células de manufatura tradicionais e células de manufatura virtuais. Além disso, é apresentada a motivação para a escolha adotada.

No capítulo anterior, subitem 2.1.3, foram identificados seis desses padrões de referência, os quais foram empregados na literatura para modelar e comparar as configurações estudadas, e esses trabalhos foram: (a) Drolet (1989) e Mak e Wang (2002); (b) Morris e Tersine (1990), Kannan e Ghosh (1996) e iKannan (1997a) e Kannan (1998); (c) Kannan (1997b); (d) Vakharia et al. (1999); (e) Jensen (2000), e (f) Suresh e Slomp (2005).

Entretanto, dois desses padrões não se adequam bem ao foco desta tese, uma vez que, ainda que abarquem o conceito de células virtuais, se enquadram mais na avaliação do leiaute por produto (*flow shop*), conforme apontado anteriormente, ou seja, Vakharia et al. (1999) e Jensen (2000). Por esse motivo, esses trabalhos foram descartados na escolha do padrão ideal, bem como para efeito de comparação entre as configurações.

Por outro lado, somente Suresh e Slomp (2005) comparam os três tipos de configuração contemplados no estudo desta tese. O padrão desses autores é realmente o mais complexo com relação aos demais, uma vez que envolve o maior número de máquinas, ou seja, 31 máquinas, além de empregar simultaneamente duas restrições de recursos. De fato, Drolet (1989) e Mak e Wang (2002) empregam 21 máquinas, enquanto Morris e Tersine (1990), Kannan e Ghosh (1996), Kannan (1997a), Kannan (1998) e Kannan (1997b) empregam 30 máquinas. E com relação às restrições de recursos, o trabalho de

Suresh e Slomp (2005) é o único que emprega simultaneamente a restrição com relação à disponibilidade de máquinas e de operadores, sendo por isso classificado como DRC - *dual resource constrained*. Além disso, o padrão desses autores é o mais recente.

Devido aos motivos apresentados acima, o padrão e modelos de Suresh e Slomp (2005) foram escolhidos como base para a elaboração dos modelos utilizados nesta tese. Esses autores também empregam em seus trabalhos a metodologia da simulação associada ao projeto de experimentos, o que favorece posteriores comparações. Deve ser creditado ainda aos referidos autores a concepção balanceada dos três modelos, que contam com a mesma capacidade de recursos, assim como idênticas cargas. Essas cargas são geradas: (a) por uma demanda equilibrada de lotes de peças que podem ser processados de duas formas; (b) por etapas nos distintos departamentos de um sistema *job shop*, ou nas células dedicadas a famílias específicas de peças no sistema celular tradicional ou no celular virtual.

Apesar desses autores terem apresentado modelos muito bem elaborados e detalhados, conferindo uma boa qualidade aos seus trabalhos de pesquisa em comparação aos demais trabalhos anteriores, pode-se identificar lacunas, imprecisões e pequenos equívocos que são apontados mais adiante. Procedeu-se desta forma não com o intuito de desmerecer a significativa contribuição de Suresh e Slomp (2005), mas para justificar a impossibilidade de replicar seus experimentos de forma que os resultados do trabalho desenvolvido nesta tese pudessem ser comparados inequivocamente com os resultados de Suresh e Slomp (2005).

Suresh e Slomp (2005) especificaram bastante bem a maior parte desse experimento, apesar de terem omitido algumas partes importantes que impedem a sua replicação para efeito de aprofundamento científico, além de terem apresentado alguns pequenos equívocos - provavelmente por motivo de falha de impressão gráfica. Ambos os casos são abordado com profundidade no próximo capítulo, entretanto, como já foi frisado, não chegam a tirar a qualidade das descrições feitas de praticamente toda a sua metodologia e dos modelos empregados. Entretanto, essas poucas falhas acabam afetando os modelos significativamente, dentre as quais tem-se: (a) ausência de algumas informações sobre o cálculo adotado para o intervalo entre chegadas de peças; (b) alguns valores referentes à flexibilidade dos operadores; (c) ausência de informações sobre os roteiros de fabricação empregados para a produção das peças; (d) algumas vezes a descrição das regras de prioridade na utilização das máquinas e dos operadores é ambígua, não se entendendo, por exemplo, se o serviço fica esperando numa fila por operador e máquina livres ou só por um operador livre. Esse último problema é o mais grave, pois afeta diretamente o modelo conceitual.

Por outro lado, gostaria-se de reforçar e reconhecer que, graças à elevada acuidade do trabalho apresentado por Suresh e Slomp (2005), bem como nos resultados e nas conclusões, foi possível desenvolver novos modelos. Isto facilitou o descarte de fatores que se mostraram menos eficientes nos seus resultados, e pode-se introduzir outros que não foram contemplados por esses autores. Permitiu também estudar e avaliar novos fatores geradores de desempenhos melhores.

Um dos fatores do padrão de Suresh e Slomp (2005) que não foi incorporado nos modelos desta tese é aquele denominado “regra quando”. Esta regra estabelece dois procedimentos diferentes para a liberação dos operadores. No primeiro, denominado de “COMP”, os operadores são liberados para atender solicitações de outros departamentos sempre que terminarem uma tarefa. No segundo, denominado de “IDLE”, os operadores só são liberados quando, além de concluir a tarefa, não haja filas no departamento em que estão alocados. Os dois casos ou níveis dessa regra (COMP e IDLE) foram empregados como os dois níveis de um dos fatores do experimento. Como o resultado desse fator, apesar de significativo, foi reduzido em relação à contribuição dos outros fatores, optou-se por não empregar os dois níveis, mas apenas o que obteve o melhor resultado, ou seja, o COMP.

Associada a esta regra, Suresh e Slomp (2005) introduziram outra regra, denominada “regra onde”, que dá uma maior prioridade de uso de um operador ao centro de trabalho que tenha a maior fila, ou seja, a regra de prioridade ao maior tamanho de fila (LNQ – *Longest Number of Jobs in Queue*). Esta regra também foi utilizada nos modelos *job shop* e celular tradicional desta tese. Por outro lado, esta regra não foi utilizada na configuração celular virtual, pois essa configuração é significativamente diferente da apresentado por Suresh e Slomp (2005), como se explica com mais detalhes no próximo capítulo.

Entretanto, o que mais diferencia o padrão de Suresh e Slomp (2005) com o apresentado nesta tese é a nova concepção de modelo de células virtuais utilizada nessa tese. Segundo a lógica de Suresh e Slomp (2005), cada novo serviço busca prioritariamente uma máquina já operando um serviço da mesma família desse serviço, e assim, antes de se iniciar o processamento, com frequência já há um tempo perdido numa fila esperando pela liberação de uma máquina ocupada. Caso não haja máquina operando um serviço dessa família, busca-se uma máquina livre e um operador livre do mesmo departamento da máquina livre. Caso isso não seja possível, procura-se uma máquina livre e um operador livre de outro departamento habilitado para operar esta máquina. Caso isso ainda não seja possível, espera-se numa fila. O texto de Suresh e Slomp (2005) não especifica claramente se essa é uma fila geral, se o serviço irá ocupar a máquina e esperar um operador livre, ou se vai esperar ter operador e máquina livres etc.

O modelo virtual desta tese prioriza a utilização dos recursos disponíveis, ainda que favorecendo também o uso dos recursos já preparados que tendem a contribuir para a redução do tempo de fluxo etc. Por esse motivo, a lógica do modelo faz com que cada serviço novo verifique se há alguma máquina livre já ajustada para a família desse serviço. Caso não haja uma máquina nessas condições, busca-se uma máquina livre. Caso não haja também máquina nessas condições, busca-se uma máquina ocupada já ajustada para este serviço. Caso não haja ainda essa máquina, será necessário esperar pela primeira máquina livre. No quarto capítulo serão descritos mais detalhadamente as características conceituais desse modelo.

### **3.2 Sequenciamento de Produção por Regras de Prioridade**

O método utilizado nos modelos para a geração das seqüências de produção é fundamental para o desempenho do modelo, devido à enorme influência que o método acarreta nos resultados. Conforme apontado no capítulo 2, o principal método empregado para a geração das seqüências de produção, nos trabalhos de pesquisa que desenvolveram comparações entre os tipos de processos de manufatura, foi o uso das regras de prioridade. Os trabalhos de Kannan e Ghosh (1996), Kannan (1997a, 1997b e 1998) e Suresh e Slomp (2005) utilizam regras de prioridade.

As regras de prioridade apresentam uma enorme flexibilidade e simplicidade de aplicação. Podem ser empregadas facilmente para os mais diversos modelos, além de serem versáteis e fáceis de modelar na simulação combinada com projeto de experimentos. Os autores as utilizaram com finalidade instrumental e o foco das pesquisas foram os sistemas produtivos. Provavelmente por todos esses motivos elas são empregadas pela maioria dos autores, e por essa mesma razão optou-se por empregar este método também neste trabalho de pesquisa.

As regras de prioridade empregadas nos modelos dessa tese são LNQ– *Longest Number of Jobs in Queue*, SPT - *Shortest Processing Time*, FCFS - *First Come, First Served* e tarefas que não alterem o *setup* da máquina. A escolha dessas regras em particular baseou-se nas melhores práticas dos diversos autores, conforme descrito no capítulo anterior. Os modelos que simulam o *job shop* e o celular tradicional utilizam apenas as duas primeiras regras, enquanto o celular virtual emprega as três últimas. No capítulo 4 faz-se uma descrição detalhada da aplicação dessas regras nos modelos.

### 3.3. O software de simulação

Uma vez definidos os padrões e modelos para representar os cenários que são considerados, bem como o mecanismo empregado para o sequenciamento de produção, este item busca descrever a metodologia empregada no processo de simulação desses modelos e o software de simulação utilizado.

No capítulo anterior foi destacado o enorme benefício do uso da simulação na compreensão da dinâmica de um sistema, através de modelos, e os efeitos das mudanças introduzidas nos mesmos. Também foi visto que muitas etapas são necessárias para a confecção dos modelos a serem empregados na simulação. Como os modelos dos cenários a serem avaliados já estão concebidos, considera-se que a etapa de planejamento já está realizada (ver figura 2.11). Além disso, considera-se também que a primeira tarefa da etapa de modelagem, que corresponde à coleta de dados, conforme mostrado na figura 2.11, também está concluída, para a qual serão utilizados os padrões encontrados na literatura, conforme mostrado anteriormente. Portanto, para completar a etapa de modelagem, restam duas atividades: a tradução do modelo e a verificação e validação do modelo.

A tradução dos três modelos, que correspondem às configurações *job shop*, celular tradicional e celular virtual, é uma das tarefas que demandou uma significativa parte dos esforços desse trabalho de tese. Com relação à etapa de verificação e validação do modelo, foram utilizadas algumas das técnicas indicadas por Freitas Filho (2001), mencionadas no subitem 2.2.1, com emprego particular de: (a) modelos determinísticos, e (b) variações sobre os dados de entrada. Por se tratarem de modelos teóricos e ou adaptados de modelos extraídos de trabalhos anteriores, pode-se considerar que o conceito de validação dos modelos não se aplica no presente trabalho, uma vez que não está sendo modelado um cenário real, mas sim teórico. Entretanto, admite-se que é feita uma validação num sentido mais amplo, através da comparação com os resultados obtidos por outros autores, o que efetivamente foi realizado nesta tese.

No referencial teórico se destacaram os grandes benefícios do emprego do planejamento de experimentos associado à simulação, em particular a possibilidade de se realizar inúmeros experimentos sem precisar dispor de instalações industriais dedicadas para este fim. Além disso, o espectro do experimento pode ser bastante amplo, bem como os resultados podem ser obtidos e analisados de maneira rápida a um custo bastante reduzido, bastando usar adequadamente os fatores e seus respectivos níveis.

Com relação ao software de simulação, foi utilizado o Arena, por ser um dos melhores software desse tipo existentes na atualidade e por se dispor de uma licença acadêmica com capacidade de uso pleno de todos os recursos do programa, que viabiliza o seu emprego nesta pesquisa.

### 3.4 Metodologia para a avaliação dos resultados da simulação

Neste item são descritos os métodos empregados para planejar os experimentos a serem executados via simulação das operações de manufatura nas configurações estudadas neste trabalho de tese, isto é, *job shop*, células de manufatura tradicionais e células de manufatura virtuais, assim como descrever como serão avaliados os resultados obtidos.

#### 3.4.1. O emprego do projeto de experimentos

Conforme foi abordado no capítulo anterior, o método do projeto de experimentos consiste em escolher os fatores a serem testados, os níveis em que estes fatores irão variar, eleger as variáveis resposta, definir o número de replicações, especificar um nível de significância na análise de variância e, finalmente, decidir se devem ser usados experimentos fatoriais completos ou fracionados ou ainda uma série ortogonal.

Conforme será descrito nos próximos subitens, este projeto de experimentos utilizou seis fatores, sendo dois com dois níveis, três com três níveis e o último com quatro níveis. Empregando-se o fatorial completo com dez replicações cada, o número de ensaios necessários foi de  $r \times n^k$ , ou seja,  $10 \times 2^2 \times 3^3 \times 4^1$ , o que significa 4.320 experimentos. Em um certo ponto neste projeto de pesquisa, pensou-se em recorrer aos fatoriais com séries ortogonais, uma vez que seriam necessários apenas  $r \times (1+k(n-1))$ , ou seja  $10 \times (1+2(2-1)+3(3-1)+1(4-1))$ , isto é, apenas 120 experimentos. Entretanto, como esse procedimento gera uma quantidade de resultados menor do que o fatorial completo, e também devido ao fato que o tempo para a execução de cada simulação não é elevado, optou-se por empregar neste trabalho o método fatorial completo. O nível de significância adotado foi de 5%.

Nos próximos subitens são definidos os fatores a serem empregados e seus respectivos níveis, assim como as variáveis resposta adotadas.

#### 3.4.2. Fatores e níveis considerados no projeto de experimentos.

Optou-se pela utilização de uma quantidade elevada de fatores, de forma a gerar-se uma abrangência significativa, permitindo chegar a conclusões mais aprofundadas sobre os cenários pesquisados e estabelecendo um grau de complexidade e confiabilidade maior nos resultados do projeto de experimentos.

Foram escolhidos seis fatores: tipo de sistema, nível de redução de *setup*, tamanho do lote das peças, flexibilidade dos operadores, variabilidade da demanda e intervalo entre chegadas de peças. Os quatro primeiros também foram empregados por Suresh e Slomp

(2005), e sua utilização na presente tese favoreceu a comparação com os resultados obtidos por aqueles autores. Os demais, ou seja, variabilidade da demanda e intervalo entre chegadas de peças, procuraram agregar ao padrão escolhido novos dados sobre o desempenho das três configurações estudadas ante as variações habituais da demanda e da frequência de chegada de serviços nos sistemas produtivos. Suresh e Slomp (2005) usaram um quinto fator denominado de “regra quando”, que foi descartado no presente trabalho pelos motivos já citados anteriormente.

A tabela 3.1 ilustra os fatores e níveis selecionados para os experimentos e que são abordados com mais profundidade no subitem 4.2.1.

Tabela 3.1 – Fatores e quantidade de níveis selecionados para os experimentos

<i>Tipo de Sistema</i>	<i>Nível de Redução de Setup</i>	<i>Tamanho do Lote de peças</i>	<i>Flexibilidade dos Operadores</i>	<i>Variabilidade da Demanda</i>	<i>Variação nos Intervalos entre Chegadas de Peças</i>
3	2	4	3	3	2

### 3.4.3. Variáveis resposta utilizadas no projeto de experimentos.

Dentro do conceito de projeto de experimentos, o emprego de diversos tratamentos replicados, variando e combinando diversos níveis de uma gama de fatores, busca avaliar e identificar quais desses fatores são responsáveis por determinados efeitos, chamados de variáveis resposta, sendo que as variáveis resposta utilizadas neste trabalho foram: tempo médio de fluxo; tempo médio de fila; ocupação dos operadores e quantidade de peças produzidas no tempo. A tabela 3.2 ilustra as variáveis resposta selecionadas nesta metodologia e que são abordado com mais profundidade no subitem 4.2.2.

Tabela 3.2 – Variáveis resposta selecionadas para os experimentos

<i>Experimento</i>	<i>Fator 1</i>	<i>Fator 2</i>	<i>.....</i>	<i>Fator r</i>	<i>Variáveis Resposta</i>			
					<i>Tempo Médio de Fluxo</i>	<i>Tempo Médio de Fila</i>	<i>Ocupação dos operadores</i>	<i>Quantidade de peças produzidas</i>
1	1	1	....	1	45	45	89	20
2	2	1	....	1	47	47	79	17
...	...	...	.	...	...	...	...	...
N	3	3	....	3	50	50	82	10



#### 3.4.4. O software MINITAB para o projeto de experimentos

Conforme mencionado anteriormente, para se verificar se a variação do nível dos fatores afeta significativamente as variáveis resposta, deve-se submeter os resultados à análise de variância (ANOVA). Por outro lado, o cálculo da análise de variância para experimentos que envolvem muitos fatores e com mais de dois níveis, como é o caso desta proposta, é muito trabalhoso. Por último, os gráficos podem se tornar imprescindíveis para determinar se há interação entre os fatores. Por todos esses motivos o emprego de uma ferramenta estatística foi essencial, sendo que o software escolhido foi o MINITAB.

Entre outros benefícios do uso dessa ferramenta estatística, destacar-se os seguintes:

- Cálculo automático dos graus de liberdade, dos erros, das variâncias, dos quadrados das somas, das interações etc.
- Geração automática dos gráficos dos efeitos principais (variáveis resposta) em função de cada um dos fatores testados ou da combinação deles.
- Permite trabalhar com uma gama elevada de fatores.
- Permite trabalhar com um número elevado de níveis para cada fator.
- Permite realizar o experimento com fatorial completo.
- Permite planejar o experimento com um número elevado de replicações, isto é, número de repetições do mesmo tratamento.
- Capacidade de importação e exportação de dados com os formatos de planilha de cálculo, editor de texto, etc.

Conclui-se o presente capítulo sintetizando que a metodologia apresentada permitiu estudar as operações de manufatura das configurações *job shop*, celular tradicional e celular virtual, mediante modelos baseados no padrão estabelecido por Suresh e Slomp (2005), representativos das características dos três modelos de configuração, com capacidades e demandas idênticas. As seqüências de produção foram geradas por regras de prioridade, com ênfase nas seguintes: LNQ – *Longest Number of Jobs in Queue*, SPT - *Shortest Processing Time*, FCFS - *First Come, First Served*, e trabalhos que não alterem o *setup* da máquina. Os três modelos serão avaliados e comparados de forma estatisticamente significativa mediante uma análise de variância com nível de significância de 5%, empregando o software estatístico MINITAB, fazendo variar seis fatores, a saber, tipo de sistema, nível de redução de *setup*, tamanho do lote das peças, flexibilidade dos operadores, variabilidade da demanda e intervalo entre chegadas de peças, cada um desses fatores variando entre dois a quatro níveis, e realizando-se dez replicações para cada experimento. Serão consideradas quatro variáveis resposta, a saber, tempo médio de

fluxo, tempo médio de fila, ocupação dos operadores e quantidade de peças produzidas, cujo projeto de experimentos empregou um arranjo fatorial completo executado através do software de simulação Arena versão 7.0.

No próximo capítulo será apresentado em detalhes o modelo conceitual dos sistemas a serem simulados.

## CAPÍTULO 4

### DESCRIÇÃO DO MODELO CONCEITUAL

Este capítulo descreve o modelo conceitual desse trabalho de tese. Trata-se, conforme abordado nos capítulos anteriores, de uma pesquisa que utiliza técnicas de modelagem e simulação para realizar um projeto de experimentos (DOE – *Design of Experiments*) que permita comparar configurações distintas para avaliar o escopo real das vantagens competitivas de uma nova concepção e modelagem de célula virtual com relação às demais configurações.

#### 4.1 Descrição do Padrão e Modelos Adotados

O padrão adotado neste trabalho, conforme abordado anteriormente foi aquele apresentado por Suresh e Slomp (2005), com algumas modificações. Este padrão adotado contém três modelos, representando as três configurações estudadas, isto é, *job shop*, celular tradicional e celular virtual. Nos próximos subitens é feita a descrição geral comum aos três modelos, e a seguir a descrição específica de cada modelo.

##### 4.1.1 Descrição geral do padrão comum aos três modelos

Os três modelos têm a mesma capacidade (a que é gerada por trinta e uma máquinas e vinte e um operadores) e estão sujeitos à mesma carga (os mesmos pedidos, com os mesmos roteiros de fabricação e com as mesmas demandas). Trata-se então de um padrão com modelos sujeitos a uma dupla restrição de recursos (DRC – *Dual-Resource-Constrained*), de máquinas e operadores. Nesses modelos são processados 50 distintos tipos de peças. Como se pode observar na tabela 4.1, cada um dos cinquenta tipos de peças tem um roteiro de fabricação próprio, necessitando de determinadas máquinas e quantidades de operações, o que será especificado com maiores detalhes mais adiante.

Tabela 4.1 – Tipos de peças

<b>Famílias de Peças</b>					
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Nº de tipos de peças	12	10	8	8	12
Nº de operações	4 – 6	3 – 6	4 – 6	4 - 7	4 – 6
Tipos de máquinas	1, 2, 4, 5, 6, 8	2, 3, 7, 8, 9, 10	1, 3, 5, 6, 8, 10	1, 2, 4, 5, 7, 9, 10	1, 3, 4, 6, 7, 9
Demanda anual	248 - 2.232	310 – 2.790	395 – 3.555	395 – 3.555	248 – 2.232

Fonte: adaptado de Suresh e Slomp (2005)

Por outro lado, cada tipo de peça, nos modelos celular tradicional e celular virtual, está associado a uma das cinco famílias de peças processadas nos modelos, de forma que cada família agrupa de oito a doze tipos de peças.

Conforme mencionado anteriormente, para cada um dos 50 tipos de peças é atribuído um roteiro de fabricação diferente. Suresh e Slomp (2005) não descreveram a metodologia empregada para a definição dos roteiros de fabricação das peças, e nem especificaram os roteiros das peças, segundo eles, para serem concisos, mas essa informação é essencial. Por este motivo, neste trabalho foram desenvolvidos e introduzidos os roteiros de cada peça, que estão sintetizados na tabela 4.2, onde o recurso de sombreamento foi utilizado para facilitar a identificação das peças e suas famílias. O apêndice 2 descreve como se chegou a esta configuração de roteiros.

Tabela 4.2 – Os roteiros de fabricação das 50 peças

Família 1												Família 2								Família				
P 10	P 11	P 12	P 13	P 14	P 15	P 16	P 17	P 18	P 19	P 1A	P 1B	P 20	P 21	P 22	P 23	P 24	P 25	P 26	P 27	P 28	P 29	P 30	P 31	P 32
1	2	1	1	1	4	6	1	6	1	1	8	2	2	10	2	8	7	8	2	8	3	1	1	5
2	4	2	4	4	5	8	2	8	2	2	1	3	7	2	3	9	8	9	9	9	7	3	3	6
4	5	4	6	5	6	1	5	1	4	4	2	7	8	3	7	10	9	2	10	10	8	5	5	8
5	6	5	8	6	8	2	6	2	5	6	4	8	9	7		2	10	3		2	9	6	8	10
	8	6		8	1		8	4		8	5	9	10			3	2			3		8	10	
		8			2			5			6	10				7						10		

3			Família 4							Família 5														
P 33	P 34	P 35	P 36	P 37	P 40	P 41	P 42	P 43	P 44	P 45	P 46	P 47	P 50	P 51	P 52	P 53	P 54	P 55	P 56	P 57	P 58	P 59	P 5A	P 5B
5	6	1	8	1	1	1	1	4	10	1	10	1	1	1	9	1	9	9	7	1	1	1	7	3
6	8	3	10	3	2	2	4	5	9	2	1	2	3	3	4	3	7	7	9	3	3	3	6	4
8	10	6	1	5	4	4	7	7	7	4	2	4	4	6	3	4	6	6	1	4	4	4	4	7
10	1	8	3	6	5	7	9	9	5	5	5	5	6	7	1	6	3	4	3	6	6	6	3	9
1	3		5		7	9	10		4	7	7		7	9		7	1		4	7		7	1	
3			6		9	10			2	9			9			9			6			9		
					10				1															

Conforme abordado anteriormente, a variabilidade de demanda é teoricamente favorável ao sistema *job shop* e prejudicial ao celular tradicional, uma vez que o primeiro tem alta flexibilidade e o último baixa. Por outro lado, a configuração celular virtual deve ter um desempenho intermediário aos dois. Para permitir esta avaliação, o modelo adotado admite 3 níveis de variação na demanda dos cinquenta tipos de peças, os quais são: demanda com variação nula, demanda com variação uniforme com um quarto do intervalo pleno, e demanda com variação uniforme com intervalo pleno. A tabela 4.3 apresenta esses três tipos de demanda. O primeiro é o estabelecido no padrão de Suresh e Slomp (2005) e se chamou aqui de variação plena. O segundo tipo de demanda, denominado de “um quarto de variação”, corresponde a 25% do intervalo de variação da demanda plena. Assim, nas

famílias 1 e 5, enquanto na variação plena a demanda pode variar entre os limites 248 e 2.232 unidades, isto é, um total de 1.984 unidades), o tipo de demanda correspondendo a um quarto de variação tem apenas 496 unidades de intervalo entre os limites superior e inferior (isto é, corresponde a um quarto da variação plena). O último tipo de variação da demanda anual é a ausência de variação, ou variação nula. Os valores adotados para esta demanda com ausência de variação são os valores médios dos intervalos anteriores.

Tabela 4.3 – Os três tipos adotados de variação de demanda anual de peças

Famílias	Nula	1/4	Plena
1 e 5	1240	Unif( 992 – 1488)	Unif (248 – 2232)
2	1550	Unif (1240 – 1860)	Unif (310 – 2790)
3 e 4	1975	Unif (1580 – 2370)	Unif (395 – 3555)

O modelo adotado admite dois regimes com relação aos tempos de intervalos entre as chegadas dos lotes de peças no sistema: o regime constante e o regime conforme uma distribuição exponencial. O regime constante foi introduzido neste trabalho de tese. A tabela 4.4 mostra esses valores, assim como o tempo de chegada do primeiro lote. A definição do tempo de chegada do primeiro lote de cada uma das cinquenta peças é importante, pois senão os primeiros lotes de cada peça chegariam todos juntos no início da simulação, gerando uma demanda inicial extremamente elevada, ou então iria ocorrer um grande tempo ocioso até que começassem a chegar os primeiros lotes. O tempo de chegada dos primeiros lotes é uma característica introduzida pela metodologia proposta nesta tese, pois não constava no padrão original.

No caso do regime constante, para cada tipo de peça, o intervalo entre chegadas de lotes é dado, pelo valor de tempo fixo gerado pela expressão  $q/d_i * 3600$  horas, onde  $q$  é o tamanho do lote,  $d_i$  é a demanda gerada para a peça  $i$  e 3600 são as horas de simulação (aproximadamente um ano). No regime exponencial, para cada tipo de peça, o intervalo entre chegadas de lotes é dado, pelo valor variável de tempo gerado pela expressão  $expo(q/d_i) * 3600$  horas, onde, mais uma vez,  $q$  é o tamanho do lote,  $d_i$  é a demanda gerada para a peça e 3600 são as horas de simulação. Na simulação as unidades processadas foram os lotes e não as peças individuais. Não houve limitação na quantidade de lotes gerados exceto em função do tempo de execução de cada replicação, que foi de 3600 horas.

O primeiro lote gerado para cada tipo de peça ocorreu em  $Expo((q/d_{anual}) * 3600) / 2$  horas, onde  $q$  é o tamanho do lote,  $d_{anual}$  é a demanda anual média de cada tipo de peça. Suresh e Slomp (2005) apresentam o cálculo dos intervalos entre chegadas, que são dados pela

relação *Uniform*  $[(q/d)*3600]$  horas. Esta informação não é muito clara naquele trabalho, pois a função uniforme exige dois parâmetros, que são o valor máximo e o mínimo. Por isso, no modelo adotado no presente trabalho optou-se por considerar que os lotes gerados para atender a demanda foram introduzidos na simulação com intervalos de chegada segundo uma distribuição exponencial, isto porque, segundo Bose (2002), se costuma usar esta função para representar intervalos de chegada.

Tabela 4.4 – Tempos (em horas) dos intervalos entre chegadas de lotes de peças

1° lote	Exponencial	Constante
$\text{Expo}(q / d_{\text{anual}}) \times 3600 / 2$	$\text{Expo}(q / d_i) \times 3600$	$q / d_i \times 3600$

Onde:  $q$  = tamanho do lote;  $d_i$  = demanda da peça  $i$ ;  $d_{\text{anual}}$  = média da demanda anual da peça  $i$

O tempo de *setup* é considerado integral (isto é, sem o fator de redução  $\delta$ ) quando o lote seguinte a ser processado é de uma família distinta do lote anterior, e nesse caso o valor de tempo de *setup* é dado pela função exponencial com média de três horas, isso é  $\text{Expo}(3)$ . Nos modelos celular tradicional e virtual, quando o lote seguinte a ser processado é da mesma família que o lote anterior, aplica-se o fator de redução  $\delta$ , ou seja,  $\text{Expo}(3 \times \delta)$ . Mas sempre é necessário um tempo de *setup* para a substituição do lote de peças. A tabela 4.5 ilustra esses valores de *setup*.

Tabela 4.5 – Tempos de *setup* (em horas)

Integral	com Redução
$\text{Expo}(3)$	$\text{Expo}(3 \times \delta)$

Onde:  $\delta$  = fator de redução do *setup*

Por outro lado, o tempo de processamento de um lote em qualquer máquina é gerado pela função exponencial  $\text{Expo}(0.1*q)$  horas, onde  $q$  representa o tamanho do lote. Os tempos de movimentação dos lotes são gerados pela função uniforme  $\text{UNIF}(30, 60)$  minutos para as configurações *job shop* e celular virtual. E para a configuração celular tradicional esses tempos são dados pela função  $\text{UNIF}(3, 6)$  minutos, uma vez que os tempos de movimentação entre um departamento e outro, no caso das duas primeiras configurações, são muito maiores do que os tempos de movimentação na configuração celular tradicional. Não se consideram tempos de espera para iniciar a movimentação dos lotes. Além disso, não se contempla neste padrão como esses lotes são transportados, se por via mecânica ou por operadores específicos.

Os tempos de movimentação dos operadores, da máquina que acaba de operar para a próxima a ser operada, são gerados pela função uniforme  $UNIF(30, 60)$  minutos nas configurações *job shop* e celular virtual. Na configuração celular tradicional esses tempos são dados pela função  $UNIF(3, 6)$  minutos, novamente por se tratarem de movimentos que ocorrem respectivamente entre departamentos para as primeiras configurações e dentro de uma única célula para a configuração celular tradicional. A tabela 4.6 sintetiza esses dados.

Tabela 4.6 – Tempos (em minutos) de movimentação de lotes e de operadores

	<b>Job shop / Celular virtual</b>	<b>Celular tradicional</b>
Movimentação dos lotes	Unif ( 30 , 60 )	Unif ( 3 , 6 )
Movimentação dos operadores	Unif ( 30 , 60 )	Unif ( 3 , 6 )

A tabela 4.7 indica as máquinas que cada operador está habilitado a operar para cada nível de *cross training* ( $\eta = 1, 2$  ou  $3$  tipos diferentes de máquinas que cada operador está habilitado a operar). Ou seja, o operador “a” pode operar máquinas do tipo 1 para  $\eta = 1$ , as do tipo 1 e 2 para  $\eta = 2$  e as do tipo 1, 2 e 4 para  $\eta = 3$ . No caso da configuração celular tradicional, as operações são impossíveis para  $\eta = 1$ , pois neste caso cada operador só operaria um tipo de máquina, e há mais tipos de máquinas do que operadores em cada célula. Os três valores da tabela 4.7 que se encontram indicados por elipses correspondem a valores deduzidos a partir da lógica da tabela, mas que estavam ausentes no trabalho de Suresh e Slomp (2005).

Tabela 4.7 – Habilidades de cada operador

$\eta$	a	d	h	l	f	n	p	r	j	m	q	t	b	e	k	u	s	c	g	i	o
1	1	2	4	6	3	7	8	9	5	6	8	10	1	2	5	10	9	1	3	4	7
2	1	2	4	6	3	7	8	9	5	6	8	10	1	2	5	10	9	1	3	4	7
	2	4	5	8	2	8	9	10	3	1	10	1	2	4	1	7	10	6	9	1	3
3	1	2	4	6	3	7	8	9	5	6	8	10	1	2	5	10	9	1	3	4	7
	2	4	5	8	2	8	9	10	3	1	10	1	2	4	1	7	10	6	9	1	3
	4	5	6	5	7	3	10	2	6	3	5	8	9	7	4	5	7	4	7	6	9

Fonte: adaptado de Suresh e Slomp (2005)

#### 4.1.2 Descrição específica do modelo da configuração *job shop*

O modelo *job shop* adotado neste trabalho tem algumas peculiaridades que em parte o diferenciam do modelo apresentado no trabalho de Suresh e Slomp (2005). A figura 4.1

ilustra os dez departamentos, representados pelos retângulos, as trinta e uma máquinas, representadas por números, e os vinte um operadores, representados pelas letras. Cada lote que entra na fábrica é transportado até o departamento em que ele deve ser primeiramente processado de acordo com o seu roteiro de produção. O tempo dessa movimentação, por não estar explicitado no modelo de Suresh e Slomp (2005), foi definido como sendo igual a dez minutos, valor bem reduzido de maneira a não ter peso no conjunto da simulação.

A seguir descreve-se a lógica das operações do lote dentro do departamento. O fluxograma da figura 4.2 foi desenvolvido de forma sintética para facilitar a compreensão, seguindo o padrão dos fluxogramas de atividades (Chiavenato, 2001). Sempre que um lote chega a um departamento, primeiramente ele busca obter uma máquina disponível para realizar o processamento. Caso não houver uma máquina disponível, o lote espera numa fila. A ordem dos lotes na fila é determinada pela regra de prioridade SPT – *Shortest Processing Time*. Após obter-se a máquina para a fabricação do lote, o mesmo busca obter um operador interno (do próprio departamento). Caso não houver a disponibilidade de um operador interno, o lote busca obter um operador externo (de outro departamento) que possua a habilidade necessária, e seria mais fácil encontrar tal operador se o nível de *cross training* fosse de três máquinas por operador. Caso não houver disponibilidade de um operador com esta habilidade, o lote espera numa fila por qualquer operador (externo ou interno) que tivesse esta habilidade. Caso um operador fique livre e estiver sendo solicitado por vários departamentos ao mesmo tempo, a prioridade é dada para o departamento que tenha a maior fila (LNQ – *Longest Number of Jobs in Queue*). Caso mais de um lote do departamento esteja aguardando este operador, a ordem dos lotes na fila é determinada pela regra de prioridade SPT – *Shortest Processing Time*.

1 1 1 1 <i>A b c</i>	2 2 2 <i>d e</i>	3 3 3 <i>f g</i>	4 4 4 <i>H i</i>
5 5 5 <i>j k</i>	6 6 6 <i>l m</i>	7 7 7 <i>n o</i>	8 8 8 <i>p q</i>
9 9 9 <i>r s</i>		10 10 10 <i>t u</i>	

Figura 4.1 Representação esquemática do modelo *Job Shop*

Fonte: Suresh e Slomp (2005)

Assim que o lote consiga o operador, ele começa a ser processado. Se o operador obtido for externo, o tempo de movimentação do operador é acrescido ao tempo de *setup* e ao de processamento. Após cada operação de processamento o lote se move, com tempo de movimentação gerado pela função uniforme UNIF(30, 60) minutos, para a próxima



operação prevista na sua seqüência de produção. Quando a última operação de um lote é concluída, ele é movido (com o mesmo tempo de movimentação anterior) para a expedição, encerrando o ciclo desse lote.

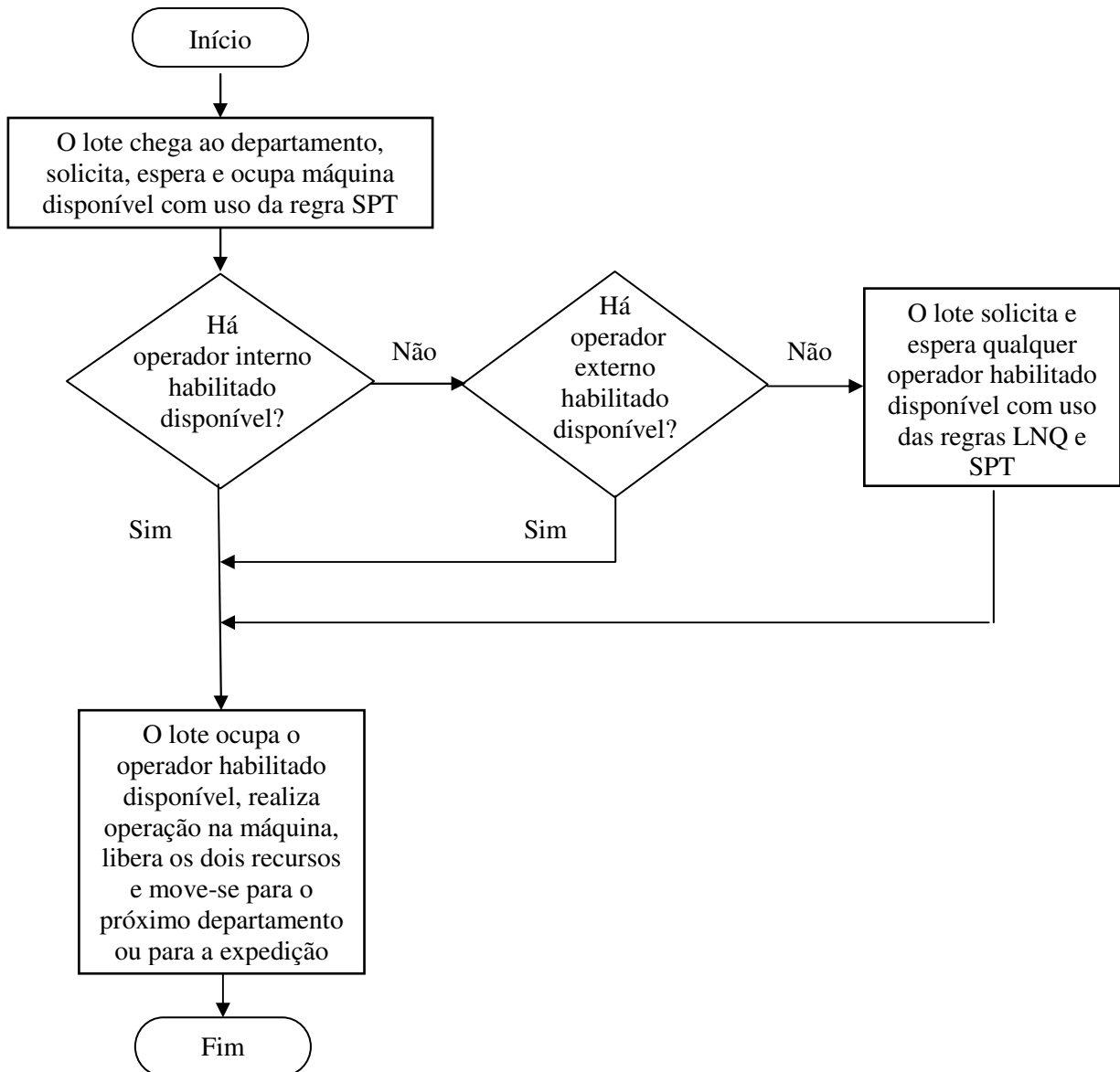


Figura 4.2 Fluxograma da lógica das operações dentro do departamento no *job shop*

#### 4.1.3 Descrição específica do modelo da configuração celular virtual

O modelo celular virtual adotado neste trabalho tem algumas peculiaridades, que o torna bem distinto daquele empregado por Suresh e Slomp (2005). A figura 4.1 também representa este modelo, uma vez que fisicamente ele é idêntico ao *job shop*. Cada lote que entra na fábrica é transportado até o departamento em que deve ser primeiramente processado segundo o seu roteiro de produção.

A seguir descreve-se a lógica das operações do lote dentro do departamento. O fluxograma da figura 4.3 foi desenvolvido de forma sintética para facilitar a compreensão. O tempo dessa movimentação, por não ter sido explicitado no modelo de Suresh e Slomp (2005), definiu-se este valor como sendo igual a dez minutos, bem reduzido para não ter peso no conjunto da simulação. Sempre que um lote chega a um departamento, sua primeira opção é tentar obter uma máquina disponível (livre) e já ajustada para processar peças da família desse lote e, caso houver máquina nessas circunstâncias, reter este recurso. Caso contrário sua segunda opção é procurar então uma máquina desse departamento que esteja disponível (livre) mesmo que não esteja ajustada para processar peças dessa família, novamente, caso houver, reter este recurso. Caso contrário, a terceira opção do lote é procurar então uma máquina desse departamento que esteja processando um lote da mesma família que as peças desse lote. Em caso afirmativo o lote solicita esta máquina com alta prioridade (caso haja outro lote também esperando por esta mesma máquina, a fila é ordenada pela regra de prioridade SPT – *Shortest Processing Time*). Caso contrário, a quarta e última opção do lote é solicitar, com prioridade média, qualquer máquina deste departamento que fique disponível (caso haja outro lote nessas mesmas circunstâncias, ou seja, nesta última opção, a fila será ordenada novamente pela regra SPT).

No caso da máquina ter sido obtida através da primeira ou terceira opção, ou seja, a máquina obtida já estar ajustada para a família de peças adequada, o lote solicita então, com prioridade alta, um operador (preferencialmente interno, senão um externo habilitado). Caso contrário, ou seja, a máquina ter sido obtida através da segunda ou quarta opção, solicita o operador apenas com média prioridade. Assim, a prioridade na concessão de um operador (seja para o próprio departamento ou para outro) será para os lotes que tenham obtido máquinas já ajustadas em relação àqueles que tenham obtido máquinas não ajustadas. E, entre os lotes de cada um desses dois grupos (com máquinas obtidas já ajustadas ou não), a prioridade é para aqueles que tenham solicitado antes, ou seja, segundo a regra FIFO, *Fist In First Out*.

Após a obtenção do operador, o lote começava a ser processado. O tempo de operação das máquinas considerado é a soma do tempo de processamento com o tempo de *setup*, que pode ser completo, se a máquina não estiver ajustada para esta família de peças, ou reduzido, caso contrário. Se o operador for externo ao departamento, é acrescido ainda o tempo de movimentação gerado pela função uniforme UNIF(30, 60) minutos. A seguir o lote é movimentado para a operação seguinte com o mesmo tempo de movimentação anterior. Quando a última operação de um lote é concluída, ele é movido (sempre com o mesmo tempo de movimentação) para a expedição, encerrando o ciclo desse lote.

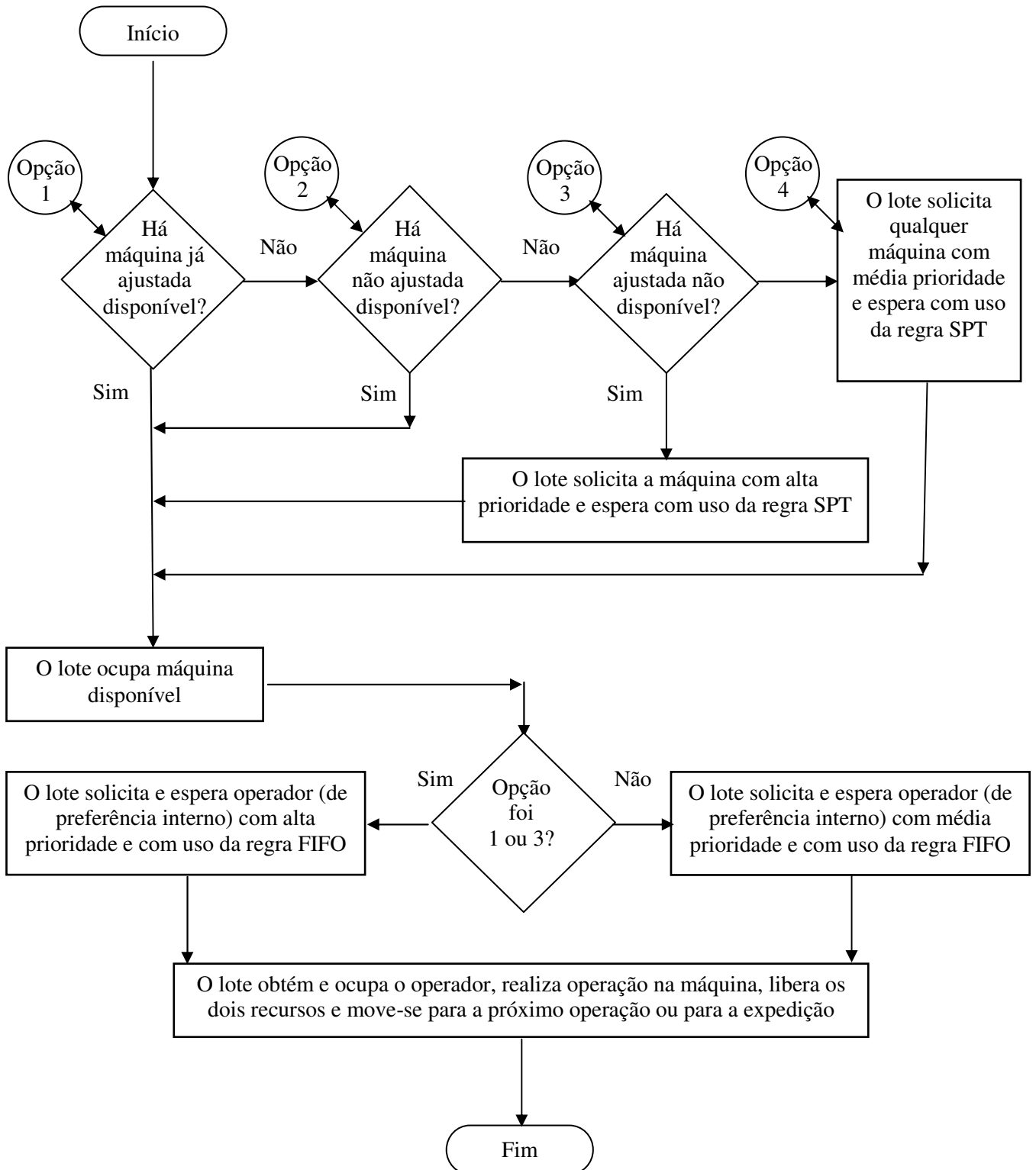


Figura 4.3 Fluxograma da lógica das operações dentro do departamento no celular virtual

O modelo da configuração celular virtual apresentado por Suresh e Slomp (2005) seguia outra lógica. Sempre que um lote chegava a um departamento, primeiramente ele verificava se alguma das máquinas estava processando peças dessa família. Em caso afirmativo ele se dirigia para a fila dessa máquina. Caso contrário, o lote procurava uma outra máquina e um operador interno disponíveis, e no caso de serem encontrados, o processamento era iniciado. Caso lotes de departamentos distintos solicitassem e aguardassem (ficar disponível) um mesmo operador, a prioridade para conseguir esse operador era dada ao lote cuja máquina já obtida pertencesse ao departamento que tivesse naquele momento a fila mais longa de lotes aguardando por um operador. Assim que o operador que ficasse disponível, era então transferido para a máquina do lote solicitante considerado prioritário (máquina essa que poderia estar no mesmo ou em distinto departamento do operador), e o processamento desse lote era então iniciado. Caso contrário este lote era colocado numa fila.

Também tinha um mecanismo, cuja lógica não foi descrita, desenvolvido para dar preferência, dentro da fila para obtenção de máquinas, aos lotes que pertenciam a famílias de peças que não estavam sendo processadas em nenhuma das máquinas do departamento. Dentre esses lotes, maior preferência era dada ainda para os que tinham mais lotes da mesma família na fila.

Os autores citam um outro mecanismo para impedir que lotes de uma mesma família assumissem muitas máquinas do mesmo departamento, mas também não descrevem o seu procedimento. A principal diferença desta lógica com a adotada neste trabalho é que a lógica proposta por Suresh e Slomp (2005) não busca prioritariamente máquinas livres já ajustadas, situação bastante factível no planejamento de produção.

#### 4.1.4 Descrição específica do modelo da configuração celular tradicional

O modelo celular tradicional tem algumas peculiaridades, que mais uma vez, em parte, são distintas de Suresh e Slomp (2005). A figura 4.4 ilustra as cinco células, representadas por retângulos, sendo que as trinta e uma máquinas são representadas por números, enquanto os vinte e um operadores são representados por letras. Como normalmente ocorre nos sistemas celulares tradicionais reais, no modelo adotado cada família de peças só passa pela célula que lhe corresponde, e como no padrão adotado há no máximo uma máquina de cada tipo em cada célula, as seqüências de produção são bem mais simples, porém menos flexíveis.

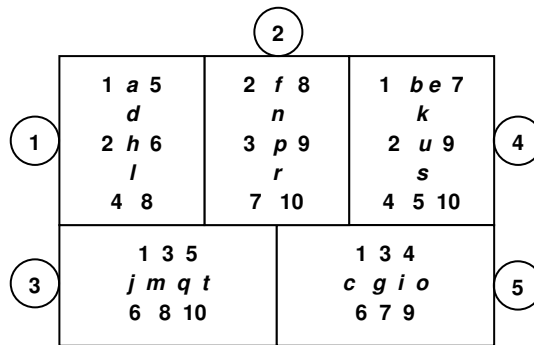


Figura 4.4 Representação esquemática do modelo da configuração celular tradicional

Fonte: Suresh e Slomp (2005)

A única flexibilidade é dos operadores que podem se movimentar dentro da célula e operar mais de uma máquina, desde que seja da sua própria célula e que eles tenham habilidade para operá-las segundo as suas características e os seus níveis de *cross training* ( $\eta = 2$  ou  $3$ ). Por outro lado, este mínimo de flexibilidade é essencial porque o número de operadores é menor que o de máquinas, e por isso o modelo não é factível para  $\eta = 1$ .

A seguir descreve-se a lógica das operações do lote dentro da célula. O fluxograma da figura 4.5 foi desenvolvido de forma sintética para facilitar a compreensão. Cada lote que entra na fábrica é transportado à célula em que deve ser processado segundo a sua família. Da mesma maneira que nos outros modelos, o tempo dessa movimentação, por não ter sido explicitado no modelo de Suresh e Slomp (2005), definiu-se como sendo igual a dez minutos, bem pequeno para não ter peso no conjunto da simulação. Sempre que um lote chega à célula que lhe corresponde, ele se dirige diretamente para a única máquina que pode realizar a operação prevista na sua seqüência de produção. Se a máquina estiver ocupada, o lote espera na fila correspondente, que é ordenada pela regra SPT. Uma vez obtida a máquina, o lote solicita um operador. Caso houver mais de um operador livre, dá-se preferência ao operador que tenha capacidade de operar esta máquina como sua primeira habilidade. Caso contrário é solicitado qualquer outro operador que tenha esta habilidade, caso esteja livre. Caso nenhum operador esteja disponível, o lote espera na fila do conjunto de operadores da célula que operam esse tipo de máquina. Nesta fila é dada prioridade maior ao lote que esteja sendo processado na máquina com maior fila de trabalhos (LNQ - *longest number of jobs*). Após obter-se o operador, o lote começa a ser processado.

O tempo de operação das máquinas considerado neste caso corresponde à soma do tempo de processamento com o tempo de *setup* reduzido, sendo que se o operador não tiver capacidade de operar esta máquina como sua primeira habilidade (ou seja, o operador

tiver que se movimentar internamente até esta máquina), é acrescido ainda o tempo de movimentação, o qual é gerado pela função uniforme UNIF(3, 6) minutos. Após cada operação de processamento, o lote se move para a próxima operação prevista na sua seqüência de produção. Se o recurso estiver ocupado, o lote espera numa fila, ordenada pelo menor tempo de processamento. Caso todas as operações tenham sido concluídas, o lote é encaminhado para a expedição, sempre com tempo de movimentação gerado pela função uniforme UNIF(3, 6) minutos.

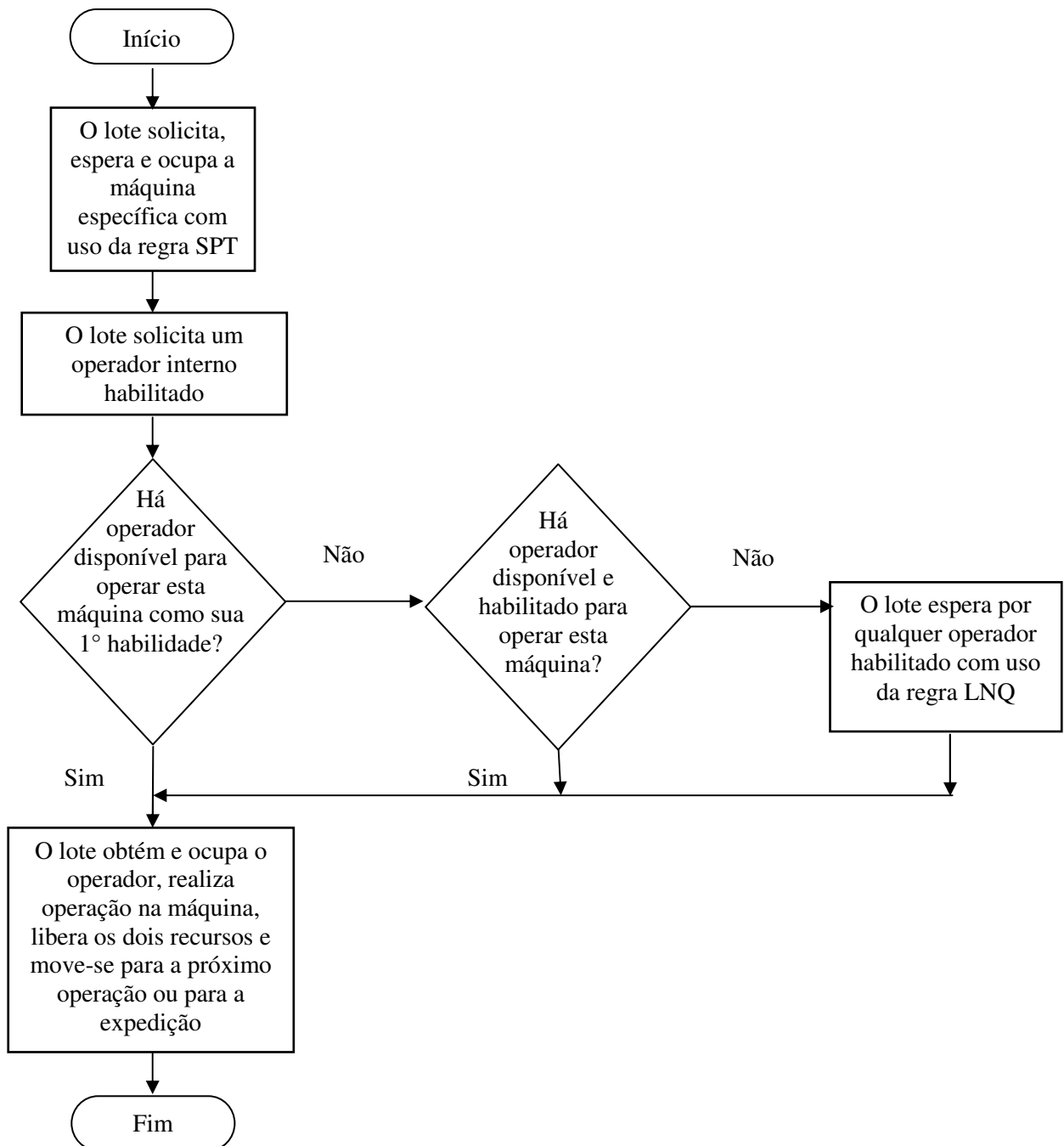


Figura 4.5 Fluxograma da lógica das operações dentro da célula na configuração celular tradicional

## 4.2 Descrição do Projeto de Experimentos Utilizado

Nos próximos subitens são descritos os fatores e as variáveis resposta empregados no projeto de experimentos utilizado neste trabalho, visando comparar as diferentes configurações para avaliar o escopo real das vantagens competitivas de uma nova concepção e modelagem de célula virtual com relação aos demais sistemas produtivos. A seguir é descrita a metodologia empregada para a coleta dos dados gerados no software de simulação e a transferência dos mesmos para as planilhas de dados. Finalmente é descrito o tratamento dos dados realizado com auxílio do software estatístico.

### 4.2.1 Descrição dos fatores empregados no projeto de experimentos

Para o estudo e avaliação de desempenho dos sistemas comparados nesta tese, os modelos descritos nos itens anteriores deste capítulo foram experimentados via simulação. Esses experimentos foram realizados de forma sistemática, com as diversas combinações de níveis dos fatores selecionados de acordo com a metodologia do projeto de experimentos adotada, mediante um fatorial completo, fornecendo valores de desempenho segundo variáveis resposta pré-selecionadas.

Neste projeto de experimentos, em conformidade com as características dos modelos empregados, seis fatores foram selecionados e utilizados: (a) Tipo de sistema de configuração; (b) Nível de redução de *setup*; (c) Tamanho médio do lote; (d) Flexibilidade dos operadores; (e) Variação nos intervalo entre chegadas de lotes de peças; e (f) Variabilidade da demanda anual ( $d_i$ ). O primeiro fator, tipo de sistema de configuração, tem três níveis: (1) *job shop*; (2) Celular Tradicional e (3) Celular Virtual. Conforme mencionado anteriormente, para cada um desses níveis foi construído um modelo conceitual e de simulação.

O segundo fator, nível de redução de *setup*, é representado neste trabalho pela letra grega  $\delta$ . O *setup* integral de cada máquina no modelo adotado é dado pela função exponencial de média três horas, ou seja, pela expressão:  $\text{Expo}(3)$ . São dois os níveis de redução desse fator, aplicados sobre a expressão  $\text{Expo}(3 \times \delta)$ : (1) redução média, que consiste de uma redução de 40% do valor integral de *setup*, ou seja,  $\text{Expo}(3 \times 0,60)$ ; e (2) redução alta, ou seja, de 75% do valor integral de *setup*, ou seja,  $\text{Expo}(3 \times 0,25)$ . Essas reduções se aplicam ao sistema celular tradicional e ao sistema celular virtual que trabalham com o sistema de famílias de peças. No leiaute *job shop* esta redução não teria sentido, pois tempos longos de *setup* é uma de suas características mais marcantes, conforme foi abordado no capítulo 1.

Para o fator tamanho médio do lote foram adotados quatro níveis: 30, 40, 50 e 60 peças. Neste fator, Suresh e Slomp (2005) trabalharam com doze níveis: 5, 10, 15, 20, 25, 30, 35, 40, 50, 60, 80 e 100 peças, portanto um número bem maior. Entretanto, uma quantidade maior de níveis, conforme foi abordado no capítulo 2, gera um número muitíssimo superior de experimentos no fatorial completo. Por outro lado, de acordo com os resultados obtidos por aqueles autores, verifica-se que os valores intermediários de tamanhos de lote correspondem àqueles que levam às maiores diferenças entre os sistemas. Assim, no presente trabalho optou-se por utilizar valores de tamanhos de lote na faixa intermediária daqueles utilizados por Suresh e Slomp (2005), mas ao mesmo tempo buscou-se um maior aprofundamento na pesquisa, ampliando-se o número de fatores e de variáveis resposta.

O fator flexibilidade dos operadores busca avaliar a influência da maior ou menor habilidade dos operadores em operar diferentes máquinas, também cunhada na literatura com o termo *cross training*. Esta flexibilidade é apresentada em três níveis, conforme abordado no subitem 4.1.1, e representado pela letra  $\eta$ : (1) Nula ou  $\eta = 1$  (o operador só opera um tipo de máquina); (2) Dupla ou  $\eta = 2$  (o operador opera dois tipos de máquinas); e Tripla ou  $\eta = 3$  (o operário opera três tipos de máquinas); Suresh e Slomp (2005) utilizaram cinco níveis:  $\eta = 1, 2, 3, 4$  e  $6,2$ . Este último nível correspondia a uma média de  $6,2$  habilidades por operador, onde a maioria tinha seis habilidades e os demais sete. Novamente, aproveitando-se a vantagens de dispor dos resultados desses autores, optou-se por empregar apenas os níveis que foram mais significativos, conforme ilustrado na tabela 4.7.

O quinto fator é a variação nos intervalos entre chegadas de lote de peças, o qual apresenta dois níveis: nula (chegadas a intervalos constantes) e alta (segundo uma função exponencial). Suresh e Slomp (2005) não empregaram este fator nem outros autores na avaliação das células virtuais. A finalidade do emprego deste fator consiste em avaliar como os três sistemas estudados se comportariam com relação às descontinuidades e irregularidades das solicitações de serviços de processamentos. Conforme abordado anteriormente, a descontinuidade na chegada das peças teoricamente seria mais prejudicial para o sistema celular tradicional que é indicado para situações de previsibilidade e demandas balanceadas. No subitem 4.1.1 foi apresentada uma definição detalhada dos níveis empregados neste fator.

A variabilidade da demanda anual ( $d_i$ ) de cada peça é o último fator utilizado no projeto de experimentos. Foram empregados três níveis: nula (demanda constante), um quarto da plena e plena. Este fator também não foi empregado por Suresh e Slomp (2005) e nem por



outros autores na avaliação das células virtuais. O subitem 4.1.1 também abordou a finalidade do emprego desse fator, assim como uma descrição aprofundada dos seus níveis.

#### 4.2.2 Descrição das variáveis resposta empregados no projeto de experimentos

Conforme citado no capítulo anterior, as variáveis resposta utilizadas no presente projeto de experimentos foram: tempo médio de fluxo; tempo médio de fila; ocupação dos operadores e quantidade de peças produzidas. A primeira variável mede a capacidade de cada configuração mover com rapidez os pedidos em execução até a sua conclusão. O tempo médio de fila indica falta de capacidade ou mau sequenciamento de produção ou baixa flexibilidade de produção. A ocupação dos operadores, no padrão adotado, indica a capacidade de gerir os recursos gargalo do sistema. E a última variável resposta, quantidade de peças produzidas, aponta para a produtividade dos sistemas estudados. Essas variáveis avaliam direta ou indiretamente duas características fundamentais para a viabilidade da atividade industrial, e que ao mesmo tempo são difíceis de conciliar, que são a redução dos tempos de fluxo e a flexibilidade de roteiro.

Para a obtenção dos resultados de cada variável resposta foi desenvolvido um conjunto de três modelos no software Arena, e cada modelo correspondendo a um tipo de configuração: *job shop*, celular tradicional e celular virtual. Para isso, foram empregados os doze arquivos abaixo:

- job shop\_tempo de fluxo\_24\_10\_07.doe
- celular tradicional\_tempo de fluxo\_24\_10\_07.doe
- celular virtual\_tempo de fluxo\_24\_10\_07.doe
- job shop\_tempo de fila\_19\_11\_07.doe
- celular tradicional\_tempo de fila\_19\_11\_07.doe
- celular virtual\_tempo de fila\_19\_11\_07.doe
- job shop\_ocupacao do sistema\_05\_12\_07.doe
- celular tradicional\_ocupacao do sistema\_05\_12\_07.doe
- celular virtual\_ocupacao do sistema\_05\_12\_07.doe
- job shop\_producao\_15\_1\_08.doe
- celular tradicional\_produção\_15\_1\_08.doe
- celular virtual\_produção\_15\_1\_08.doe

#### 4.2.3 Geração, coleta e transferência dos dados da simulação

Descreve-se neste subitem a metodologia adotada para a coleta de dados da simulação dos modelos e a respectiva transferência para as planilhas de dados, sendo que estas planilhas foram utilizadas posteriormente para a execução do tratamento estatístico.

Primeiramente, conforme ilustrado na figura 4.6, abria-se simultaneamente na tela do computador os programas Arena, Microsoft Word, Microsoft Excel (duas seções), e o Output Analyzer (que é um programa da Rockwell Software e que pertence ao conjunto de programas que dão suporte ao software Arena). A seguir escolhia-se um dos doze modelos descritos no fim do subitem anterior para executar a simulação. A título de ilustração, a figura 4.7 apresenta o modelo “celular virtual\_tempo de fluxo\_24\_10\_07.doe”, com o qual se obteve os valores da variável resposta tempo de fluxo, para o nível celular virtual do fator sistema. No Microsoft Word abria-se um arquivo, conforme a figura 4.8, contendo alguns dos valores obtidos em uma das sessões do Microsoft Excel se abria o arquivo “transposer.xls”, conforme mostrado na figura 4.9, e na outra seção abria-se o arquivo “Resultados\_24\_10\_07\_eta\_2\_3\_tempo de fluxo.xls”, conforme a figura 4.10. O programa Output Analyzer tratava os dados após cada rodada de simulação, conforme ilustrado na figura 4.11.

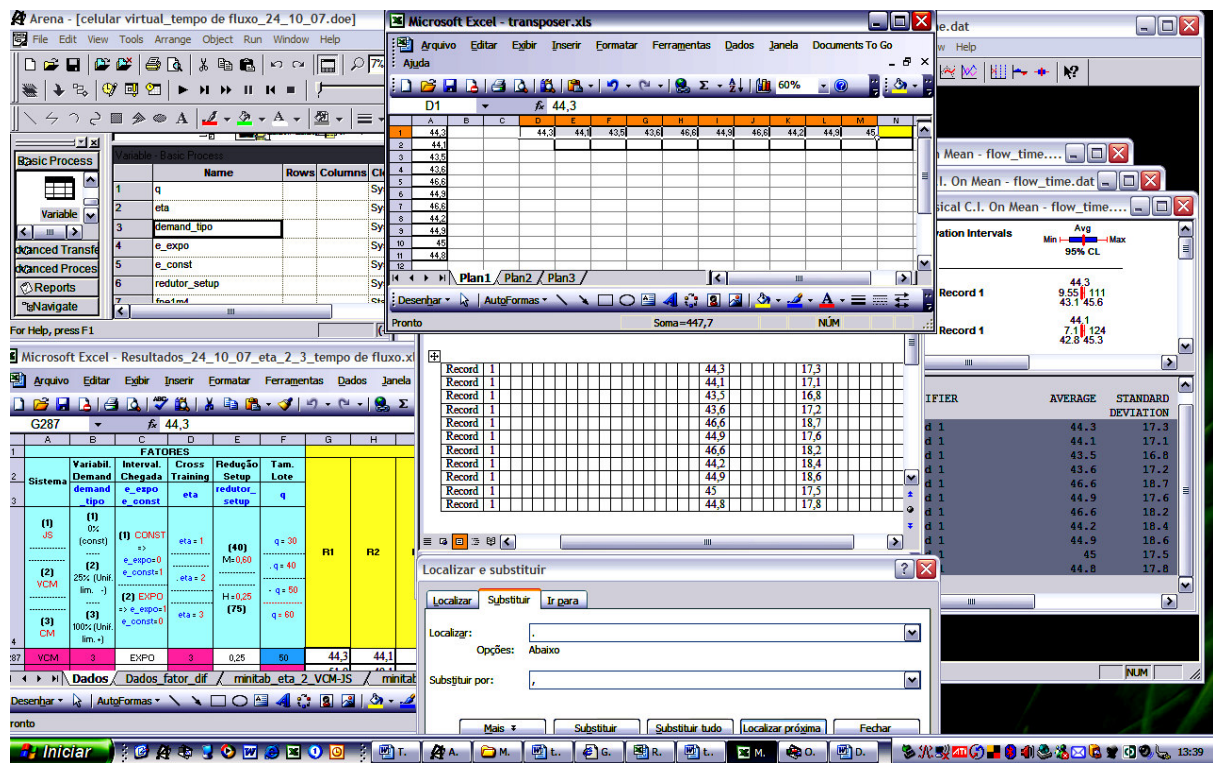


Figura 4.6 Tela do computador durante a simulação e coleta de dados

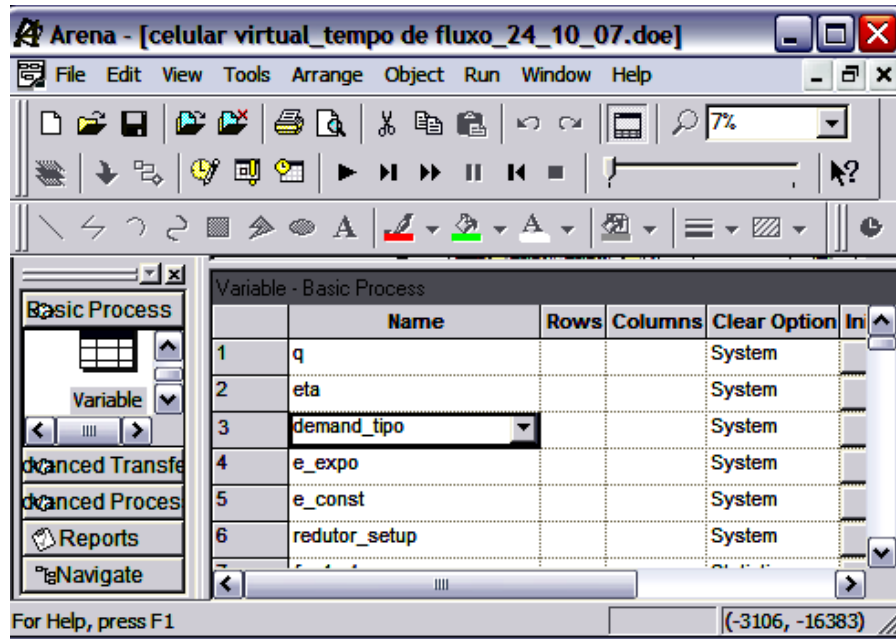


Figura 4.7 Software Arena com o arquivo “celular virtual\_tempo de fluxo\_24\_10\_07.doe”

Antes de cada rodada de simulação, alteravam-se os valores dos níveis dos fatores. Assim, por exemplo, mudava-se no software Arena os valores de  $q$  para 50, indicando o tamanho do lote com este número de peças, o valor de  $\eta$  para 3, indicando assim tripla flexibilidade do operador, o valor de demand\_tipo para 3, indicando que a variabilidade da demanda era plena, o valor de e\_expo para 1 e o de e\_const para 0, indicando que a variação do intervalo entre chegadas de peças era alta, e o valor de reductor\_setup para 0,25, indicando uma redução alta de *setup*. A seguir executava-se a simulação.

The screenshot shows a Microsoft Word document titled "Documento2 - Microsoft Word". The document contains a table with 11 rows and 11 columns. The first two columns are labeled "Record" and "1". The last two columns contain numerical values. The data in the table is as follows:

Record	1								44,3		17,3
Record	1								44,1		17,1
Record	1								43,5		16,8
Record	1								43,6		17,2
Record	1								46,6		18,7
Record	1								44,9		17,6
Record	1								46,6		18,2
Record	1								44,2		18,4
Record	1								44,9		18,6
Record	1								45		17,5
Record	1								44,8		17,8

The status bar at the bottom indicates "Pág 1", "Seção 1", "1/1", "Em 2,5 cm", "Lin 1", "Col 1", and "GRA ALT EST SE Ingl".

Figura 4.8 Arquivo do software Microsoft Word

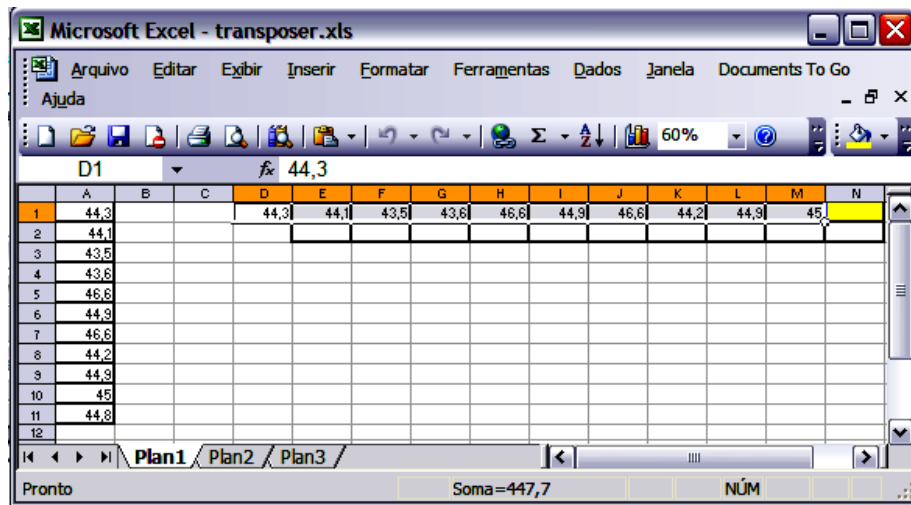


Figura 4.9 Software Microsoft Excel com o arquivo “transposer.xls”

Ao terminar cada rodada da simulação eram coletados os resultados no software Output Analyser. A Figura 4.12 mostra, em uma tela expandida, os resultados calculados por esse software. Os primeiros dez valores, da primeira à décima linha da tela da figura, na coluna *AVERAGE*, correspondem aos tempos médios de fluxo dos cerca de 700 lotes que foram completamente processados em cada uma das dez replicações de cada experimento, correspondendo a uma rodada na simulação. Os valores exatos dos tamanhos de lotes estão listados na coluna “*number of obs.*” na tela da figura. Para completar a parte do experimento com a variável resposta tempo de fluxo, somente para  $\eta = 2$  e 3, correspondente apenas às células virtuais foram necessárias 288 rodadas de simulação, de dez replicações cada, com o software Arena. A décima primeira linha da tela da figura corresponde à média global das dez replicações de uma rodada, que é aproximadamente igual à média das médias de cada replicação, uma vez que o desvio padrão e o número de observações são bastante uniformes, como se pode observar na mesma figura.

Os dados eram então transferidos para planilhas eletrônicas para viabilizar futuros tratamentos. Para armazenar separadamente os resultados referentes às quatro variáveis resposta foram empregadas oito planilhas, duas por variável. Recorreu-se à utilização de duas planilhas por variável resposta devido ao fato que a configuração celular tradicional no padrão adotado, conforme mencionado anteriormente, não admite o nível mais baixo do fator flexibilidade dos operadores, ou seja, a flexibilidade Nula (ou  $\eta = 1$ ). Assim, para  $\eta = 1$ , a simulação considerou para cada variável resposta apenas as configurações *job shop* e celular virtual, enquanto que os outros dois níveis de flexibilidade  $\eta = 2$  e (Dupla) e  $\eta = 3$  (Tripla) foram considerados as três configurações. Abaixo são listados os nomes dos oito arquivos Excel utilizados neste caso:

- Resultados\_24\_10\_07\_eta\_1\_tempo de fluxo.xls
- Resultados\_24\_10\_07\_eta\_2\_3\_tempo de fluxo.xls
- Resultados\_19\_11\_07\_eta\_1\_tempo de fila.xls
- Resultados\_19\_11\_07\_eta\_2\_3\_tempo de fila.xls
- Resultados\_05\_12\_07\_4\_eta\_1\_ocupacao.xls
- Resultados\_05\_12\_07\_4\_eta\_2\_3\_ocupacao.xls
- Resultados\_15\_1\_08\_eta\_1\_producao.xls
- Resultados\_15\_1\_08\_eta\_2\_3\_producao.xls

FATORES						REPLICAÇÕES											
Sistema	Variabil. Demand	Interval. Chegada	Cross Training	Redução Setup	Tam. Lote	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	verificador	
(1) JS	0% (const)	(1) CONST => e_expo=0 e_const=1	eta = 1	(40) M=0,60 H=0,25 (75)	q = 30 - q = 40 - q = 50 q = 60												
(2) VCM	25% (Unif. lim. -)	(2) EXPD => e_expo=1 e_const=0	eta = 2														
(3) CM	100% (Unif. lim. +)	(3) EXPD => e_expo=1 e_const=0	eta = 3														
2	1	2	3	75	50	44,3	44,1	43,5	43,6	46,6	44,9	46,6	44,2	44,9	45	44,8	

Figura 4.10 O arquivo “Resultados\_24\_10\_07\_eta\_2\_3\_tempo de fluxo.xls do Excel”

O autor desta tese constatou que o software Output Analyser não exporta os dados processados, e a solução encontrada para este problema foi usar o recurso de copiar os dados da tela do próprio software e colá-los em um arquivo do software Word, conforme mostrado na figura 4.13. Dentre os dados copiados, aqueles que interessavam neste trabalho eram os da coluna *average*. Por outro lado, era necessário também dispô-los na horizontal em forma de linha, num padrão tabela, e finalmente os pontos, empregados na notação americana para indicar a separação das casas decimais, deveriam ser substituídos por vírgulas. Assim, os valores ficariam dispostos de forma adequada para a sua manipulação futura nas planilhas de dados, conforme indicado nas figuras 4.10 (no Excel) e 4.14 (no Word). Depois o texto era convertido em tabela, utilizando-se as seguintes opções

do Word: Tabela / Converter / Texto em tabela; selecionava-se a coluna com os dados que interessavam, conforme mostra a figura 4.15, e a seguir copiavam-se esses dados.

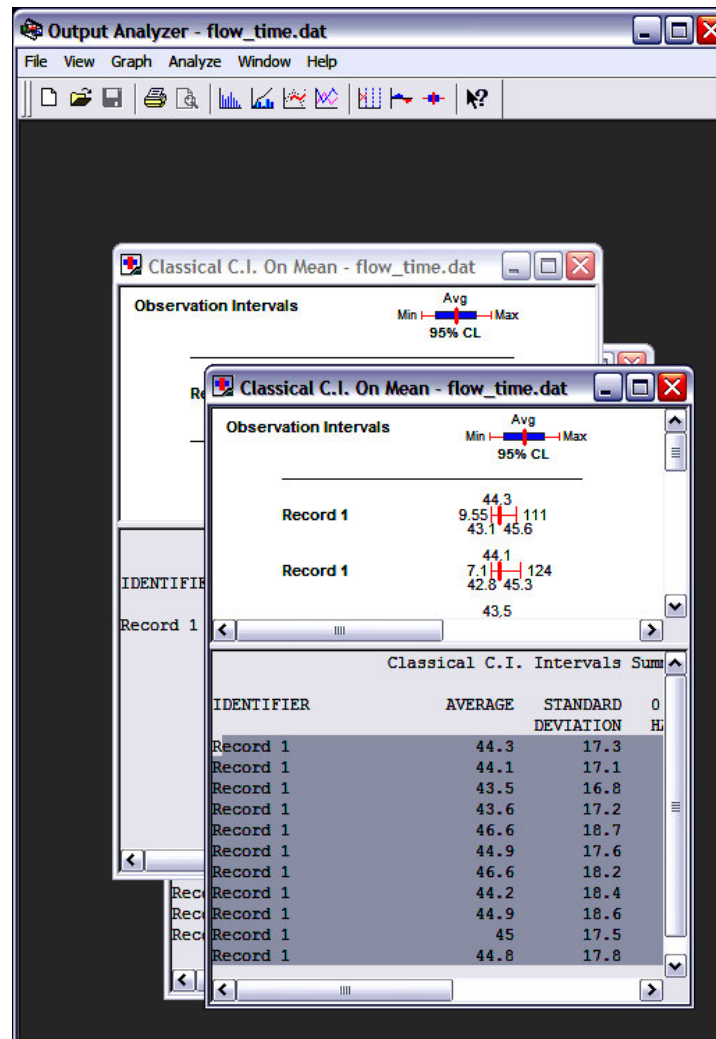


Figura 4.11 A tela do software Output Analyser

A coluna copiada do arquivo Word era então colada na coluna A de um arquivo Excel denominado “transposer.xls”, conforme indica a figura 4.9. A seguir a mesma coluna era copiada mas apenas da primeira à décima linha. Depois o conteúdo era transferido para a célula D1, através da opção do menu do programa Excel. Finalmente, os dados recém colados eram novamente copiados e colados na outra seção do Excel em que estava aberta a planilha de resultados, indicada na figura 4.10, assegurando-se que se iria colar na linha em que estavam os fatores correspondentes dessa rodada de simulação.

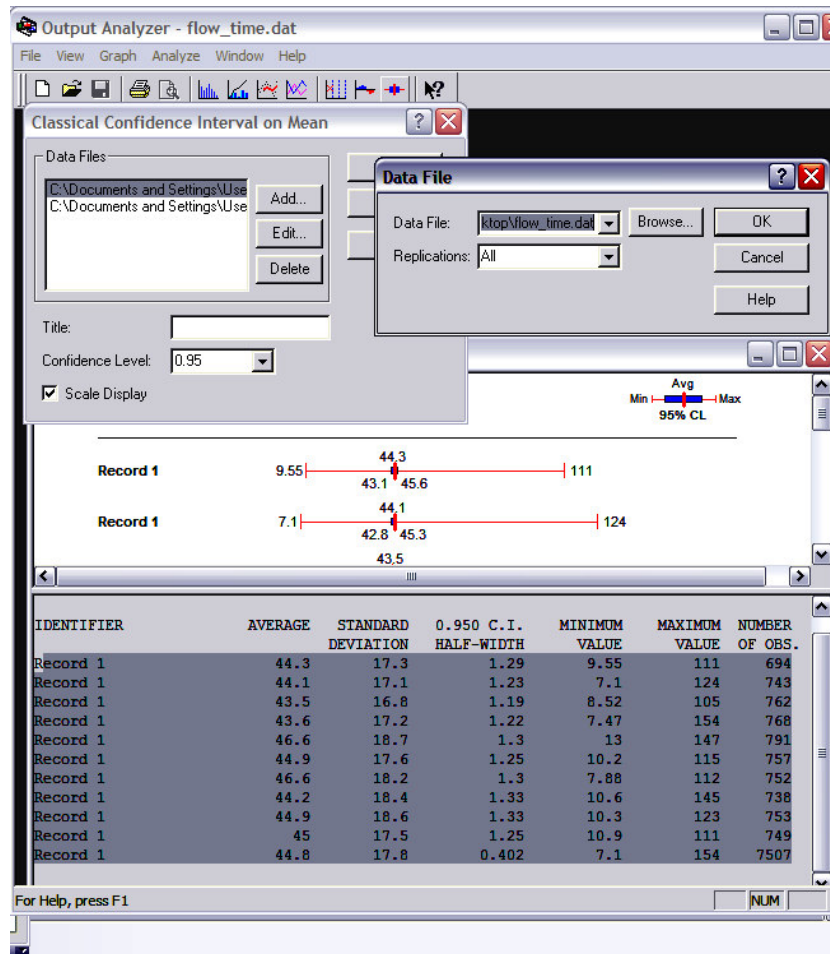


Figura 4.12 A tela expandida com os resultados calculados pelo Output Analyser

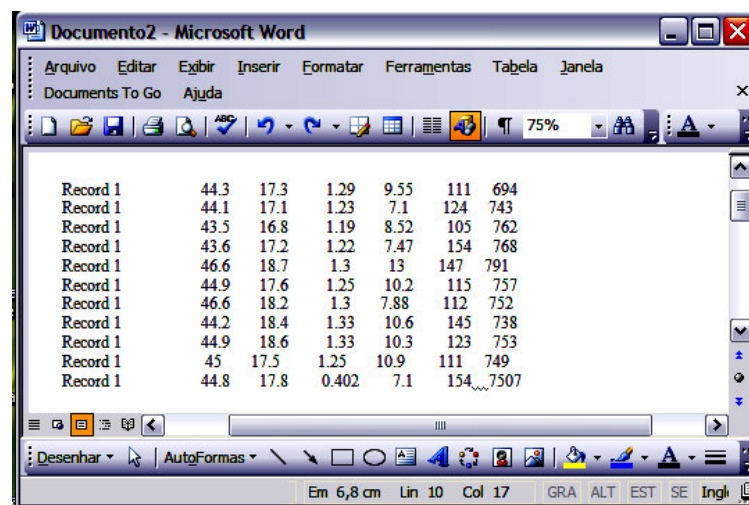


Figura 4.13 Os resultados do Output Analyser colados para um arquivo Word





#### 4.2.4 Descrição do tratamento de dados realizado nas planilhas de resultados

Conforme foi abordado no subitem anterior, as oito planilhas de resultados recebem os valores das médias de todas as replicações de cada uma das quatro variáveis resposta empregadas no projeto de experimentos. Duas planilhas para cada variável, uma para o nível nulo de flexibilidade dos operadores, isto é,  $\eta = 1$ , e outra para flexibilidade dupla e tripla, ou seja,  $\eta = 2$  e  $3$  respectivamente. Cada linha de cada uma dessas planilhas corresponde às médias das dez replicações de cada experimento. Assim, observando a planilha “Resultados\_24\_10\_07\_eta\_2\_3\_tempo de fluxo.xls” mostrada na figura 4.10, pode-se verificar que as primeiras seis colunas correspondem aos fatores, ou seja, tipo de configuração do sistema com três níveis, variabilidade da demanda com três níveis, variação do intervalo entre chegadas de peças com dois níveis, flexibilidade do operador (*cross training*) com três níveis, redução de *setup* com dois níveis e, finalmente, tamanho médio do lote com quatro níveis, nessa ordem.

As dez colunas seguintes correspondem aos valores das médias de cada uma das dez replicações. E a última coluna, a coluna denominada *Q*, corresponde às médias das dez médias anteriores. Assim, o valor de 44,8 que aparece na célula Q131 da planilha Excel na figura 4.10, corresponde ao conteúdo  $MÉDIA(G131:P131)$ , e que, como foi abordado no último parágrafo do subitem 4.2.3, serve de número verificador. Assim, esta planilha que está nessa guia do Excel, intitulada de “Dados”, foi utilizada apenas para armazenar os dados gerados no software Output Analyser.

Outras pastas desses mesmos arquivos Excel, foram utilizadas para preparar e armazenar os dados a serem transferidos para o software Minitab, que foi empregado na análise de variância dos fatores do experimento. Assim, no arquivo Excel “Resultados\_24\_10\_07\_eta\_2\_3\_tempo de fluxo.xls” mostrado na figura 4.10, existem outras quatro guias que preparam os dados para o Minitab: “minitab\_eta\_2\_VCM-JS”, “minitab\_eta\_2\_VCM-CM”, “minitab\_eta\_3\_VCM-JS”, “minitab\_eta\_3\_VCM-CM”. A figura 4.16 mostra a tela desse arquivo de resultados aberto na guia “minitab\_eta\_2\_VCM-JS”.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1							FATORES			Replicações										
2					Sistema	Demand	Interval	Redução	Tam.			Fórmula	Valor							
3					demand	e_expo	e_const	redução	lotep	q										
4					(1) JS 0% (const)	(1) CONST => e_expo=0 e_const=1	(40) M=0,60		q = 30	VCM	JS									
					(2) VCM 25% (Unif. lim.)	(2) EXPO => e_expo=1 e_const=0	H=0,25 (75)		q = 40 q = 50				VCM . JS							
					(3) CM 100% (Unif. lim.)				q = 60											
485	1	1	1	1	2	1	1	40	30	52,3	111	-58,7	-58,7							
486	1	1	1	1	2	1	1	40	30	53,6	120	-66,4	-66,4							
487	1	1	1	1	2	1	1	40	30	53,1	106	-52,9	-52,9							
488	1	1	1	1	2	1	1	40	30	55,1	104	-48,9	-48,9							
489	1	1	1	1	2	1	1	40	30	51,4	106	-54,6	-54,6							
490	1	1	1	1	2	1	1	40	30	57,5	118	-60,5	-60,5							
491	1	1	1	1	2	1	1	40	30	57,1	98,6	-41,5	-41,5							
492	1	1	1	1	2	1	1	40	30	56,1	103	-46,9	-46,9							
493	1	1	1	1	2	1	1	40	30	52,4	111	-58,6	-58,6							
494	1	1	1	1	2	1	1	40	30	57,8	102	-44,2	-44,2							
495	1	1	1	1	2	1	1	40	40	50,3	60,1	-9,8	-9,8							
496	1	1	1	1	2	1	1	40	40	47,3	55,2	-7,9	-7,9							
497	1	1	1	1	2	1	1	40	40	50,7	57,3	-6,6	-6,6							
498	1	1	1	1	2	1	1	40	40	47,8	57,6	-9,8	-9,8							

Figura 4.16 Dados da guia “minitab\_eta\_2\_VCM-JS”

Por essa figura pode-se perceber que os dados da coluna *M*, que foram armazenados para posterior transferência ao Minitab, correspondem à diferença entre os tempos de fluxo da configuração da célula virtual e os da configuração *job shop*. Diferente do que ocorria na guia “Dados” dessa mesma planilha conforme aparece na figura 4.10, onde os valores das 10 replicações apareciam na horizontal, um do lado do outro, sob os títulos R1, R2, ... R10, agora, nesta pasta “minitab\_eta\_2\_VCM-JS”, os valores das replicações estão também na vertical para facilitar a transferência dos dados na forma adequada para o software Minitab. Por esse motivo, os níveis dos cinco fatores, colunas E, F, G, H e I aparecem repetidos por dez linhas, da linha 485 até a 494, porque correspondem as dez replicações de um mesmo nível de experimento. Bem mais abaixo, nesta mesma guia - não aparece na figura 4.16 - encontram-se repetidos os dados das primeiras nove colunas, de A a I, imediatamente seguidos dos da coluna *M*, totalizando dez colunas. O bloco com os dados das dez colunas assim constituídas, com todos os níveis do experimento, é posteriormente transferido de forma integral para o software estatístico.

Suresh e Slomp (2005) avaliaram os efeitos dos fatores por eles escolhidos em termos um pouco imprecisos. Eles avaliaram por gráficos os efeitos dos fatores em cada uma das configurações estudadas, e assim, de forma visual, afirmaram que um fator influenciava mais uma configuração do que outra baseando-se nas curvas obtidas. Entretanto, em termos de análise de variância, não se podia afirmar que a variação de nível de um fator

poderia gerar uma vantagem de desempenho de um sistema com relação a outro, ou indicar cenários em que um sistema se mostrasse mais adequado do que outro. Por isso se adotou neste trabalho a análise de variância da diferença de desempenho, e por esse motivo é que se prepararam as quatro guias do arquivo mencionado. Para facilitar a análise de variância da diferença de desempenho com relação ao tempo de fluxo entre os diferentes tipos de configuração, utilizou-se a seguinte nomenclatura para as guias:

- entre as configurações celular virtual e *job shop* para  $\eta = 2$ , ou seja, flexibilidade do operador dupla: guia “minitab\_eta\_2\_VCM-JS”;
- entre as configurações celular virtual e celular tradicional para  $\eta = 2$ : guia “minitab\_eta\_2\_VCM-CM”;
- entre as configurações celular virtual e *job shop* para  $\eta = 3$ : guia “minitab\_eta\_3\_VCM-JS”;
- entre as configurações celular virtual e celular tradicional para  $\eta = 3$ : guia “minitab\_eta\_3\_VCM-CM”.

Para completar a análise de variância da diferença de desempenho com relação ao tempo de fluxo, uma quinta comparação é realizada, a da guia “minitab\_eta\_1\_VCM-CM” da planilha “Resultados\_24\_10\_07\_eta\_1\_tempo de fluxo.xls”, preparando os dados para a comparação entre as configurações celular virtual e celular tradicional para  $\eta = 1$ . Como a configuração *job shop* do padrão adotado não admite o nível de flexibilidade nulo, não é possível uma comparação entre as configurações celular virtual e o *job shop* para  $\eta = 1$ . E assim como foram feitas essas cinco análises com relação a variável resposta tempo de fluxo, foram feitas também as mesmas análises com relação as outras três variáveis resposta. Sendo então quatro as variáveis resposta, no total foram realizadas vinte análises, cinco por variável.

A variável resposta quantidade de peças produzidas teve a peculiaridade de ter sido obtida no software Output Analyser em forma de quantidade de lotes, e somente nas planilhas correspondentes é que ela foi convertida efetivamente em quantidades de peças, multiplicando-se pelo número de peças em cada lote.

#### 4.2.5 Descrição do tratamento de dados realizado com o software Minitab

No subitem anterior citou-se que no total do experimento foram realizadas vinte análises de variância com o software Minitab. Cada análise realizada ficou registrada, sendo preservada em um arquivo de projeto de experimento do Minitab. Abaixo são listados os nomes dos vinte arquivos do Minitab que contêm essas análises:

- \_24\_10\_07\_ETA\_1\_VCM\_MENOS\_JS\_TEMPO FLUXO.MPJ

- \_24\_10\_07\_ETA\_2\_VCM\_MENOS\_JS\_TEMPO FLUXO.MPJ
- \_24\_10\_07\_ETA\_3\_VCM\_MENOS\_JS\_TEMPO FLUXO.MPJ
- \_24\_10\_07\_ETA\_2\_VCM\_MENOS\_CM\_TEMPO FLUXO.MPJ
- \_24\_10\_07\_ETA\_3\_VCM\_MENOS\_CM\_TEMPO FLUXO.MPJ
- \_19\_11\_07\_ETA\_1\_VCM\_MENOS\_JS\_TEMPO FILAS.MPJ
- \_19\_11\_07\_ETA\_2\_VCM\_MENOS\_JS\_TEMPO FILAS.MPJ
- \_19\_11\_07\_ETA\_3\_VCM\_MENOS\_JS\_TEMPO FILAS.MPJ
- \_19\_11\_07\_ETA\_2\_VCM\_MENOS\_CM\_TEMPO FILAS.MPJ
- \_19\_11\_07\_ETA\_3\_VCM\_MENOS\_CM\_TEMPO FILAS.MPJ
- \_05\_12\_07\_ETA\_1\_VCM\_MENOS\_JS\_OCUPACAO.MPJ
- \_05\_12\_07\_ETA\_2\_VCM\_MENOS\_JS\_OCUPACAO.MPJ
- \_05\_12\_07\_ETA\_3\_VCM\_MENOS\_JS\_OCUPACAO.MPJ
- \_05\_12\_07\_ETA\_2\_VCM\_MENOS\_CM\_OCUPACAO.MPJ
- \_05\_12\_07\_ETA\_3\_VCM\_MENOS\_CM\_OCUPACAO.MPJ
- \_15\_1\_08\_ETA\_1\_VCM\_MENOS\_JS\_PRODUCAO.MPJ
- \_15\_1\_08\_ETA\_2\_VCM\_MENOS\_JS\_PRODUCAO.MPJ
- \_15\_1\_08\_ETA\_3\_VCM\_MENOS\_JS\_PRODUCAO.MPJ
- \_15\_1\_08\_ETA\_2\_VCM\_MENOS\_CM\_PRODUCAO.MPJ
- \_15\_1\_08\_ETA\_3\_VCM\_MENOS\_CM\_PRODUCAO.MPJ

Cada um desses arquivos de projeto de experimento utiliza outro arquivo do Minitab, do tipo planilha de dados, cujos nomes são os mesmos desses listados acima, alterando-se apenas a extensão do arquivo de “\*.MPJ” para “\*.MTW.” A figura 4.17 apresenta a tela do Minitab com o arquivo “\_24\_10\_07\_ETA\_2\_VCM\_MENOS\_CM\_TEMPO FLUXO.MPJ”. Também aparece na tela nesta figura o arquivo “\_24\_10\_07\_ETA\_2\_VCM\_MENOS\_CM\_TEMPO FLUXO.MTW”, que é a planilha de dados correspondente a este arquivo de projeto de experimentos. Os dados que aparecem neste arquivo de dados correspondem aos da figura 4.16. O método adotado para a transferência dos dados das planilhas de resultado do Excel para as planilhas correspondentes do Minitab foi: (a) seleção da área de dados do Excel a ser colada; (b) cópia da área selecionada para

a memória do computador; e (c) cópia desses dados para a área de transferência na planilha correspondente do Minitab.

Uma vez feita a cópia para a planilha do Minitab dos dados da diferença de desempenho com relação a uma variável resposta (no exemplo sendo apresentado esta variável é o tempo de fluxo), entre as configurações celular virtual e *job shop* para  $\eta = 2$ , podia-se fazer a análise de variância desse experimento. Para tanto, no Minitab, bastavam o emprego dos comandos: “Stat” (no menu principal), seguido da opção “ANOVA”, e a seguir, da opção “Balanced ANOVA”, aparecendo então a tela mostrada na figura 4.18. Então devia-se introduzir no campo “Responses” a variável resposta correspondente, no exemplo, “T. Fluxo”, que se encontra na coluna C9 da planilha do aplicativo. No campo “Model” devia-se fornecer os fatores, “Variabil. Demanda”, “Inter Chegada”, “Redução Setup” e “Tam. Lote”, que se encontravam disponíveis respectivamente nas colunas C5, C6, C7 e C8 da planilha correspondente. Então, podia-se obter os resultados clicando no botão “ok”.

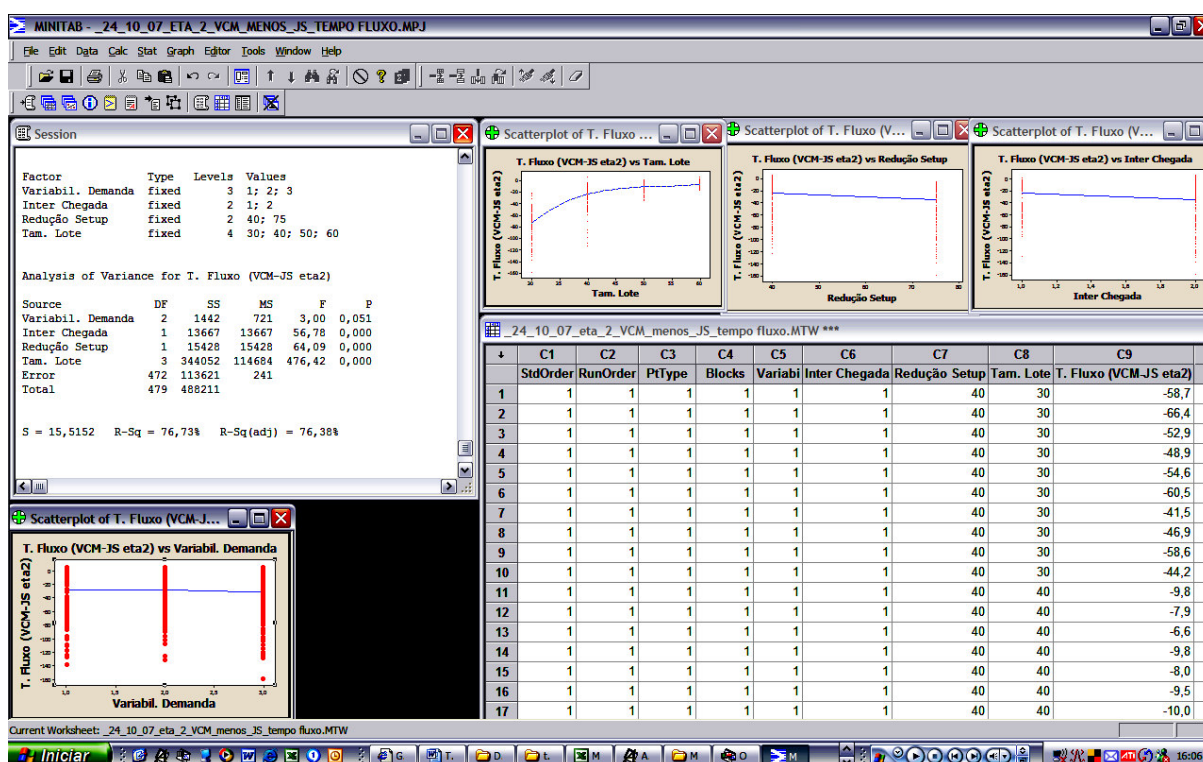


Figura 4.17 Tela do Minitab

Os resultados da Análise de Variância apareciam então na tela “Session”, apresentada na figura 4.19, onde eram exibida a estrutura do experimento, e a análise de variância para cada um dos fatores selecionados. O nível de significância adotado foi de 99%. Esses resultados são descritos e avaliados no capítulo 7 - Análise de Resultados.

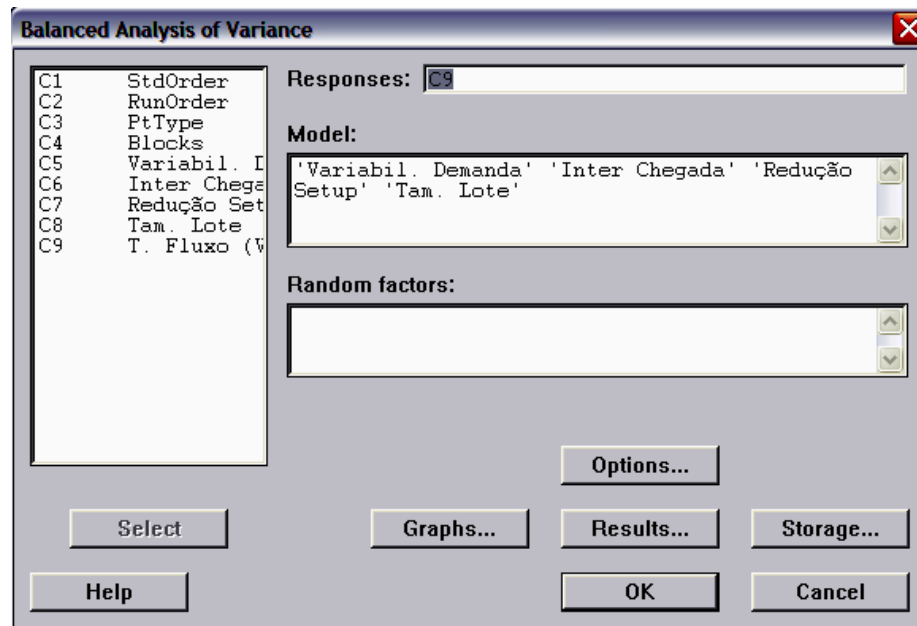


Figura 4.18 Solicitação de dados para a Análise de Variância Balanceada

Factor	Type	Levels	Values
Variabil. Demanda	fixed	3	1; 2; 3
Inter Chegada	fixed	2	1; 2
Redução Setup	fixed	2	40; 75
Tam. Lote	fixed	4	30; 40; 50; 60

Analysis of Variance for T. Fluxo (VCM-JS eta2)

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1442	721	3,00	0,051
Inter Chegada	1	13667	13667	56,78	0,000
Redução Setup	1	15428	15428	64,09	0,000
Tam. Lote	3	344052	114684	476,42	0,000
Error	472	113621	241		
Total	479	488211			

S = 15,5152    R-Sq = 76,73%    R-Sq(adj) = 76,38%

Figura 4.19 Os resultados da Análise de Variância na tela “session”

Para encerrar a descrição metodológica empregada no tratamento de dados realizado com o software Minitab, resta a descrição dos gráficos empregados na análise dos

experimentos. Dentre os gráficos disponíveis no Minitab para análise do relacionamento entre um par de variáveis tem-se o “*Scatterplots*” (dispersão) com regressão, o qual era indicado para a representação gráfica dos dados obtidos, mostrando a dispersão dos valores obtidos por simulação e seus valores médios. Nas figuras 4.20, 4.21 e 4.22, os pontos plotados formam linhas descontínuas verticais. Na mesma representação gráfica, isto é, no “*Scatterplots*”, utilizou-se também a plotagem de tendência por regressão.

Foi utilizada regressão linear nos gráficos que envolviam fatores com dois níveis, e a figura 4.20 mostra dois gráficos do tipo “*Scatterplots*”, utilizando regressão linear, um relativo ao fator redução de *setup* e o outro relativo à variação nos intervalos entre chegadas de lote de peças.

Foi empregada a regressão quadrática para representar graficamente o fator variabilidade da demanda, isto porque seus experimentos têm três níveis. A figura 4.21 ilustra o emprego do “*Scatterplots*” com regressão quadrática. Por outro lado, a regressão cúbica foi empregada na representação do fator tamanho do lote, porque este fator tem quatro níveis no projeto de experimentos, e a figura 4.22 representa esse último caso.

Como a quantidade de níveis de um experimento costuma ser pequeno, optou-se pela regressão com “*fit intercept*”, na qual as curvas passam pelos pontos médios dos pontos plotados, para evitar extrapolações nas conclusões. Desta forma, se dispôs de um recurso muito útil na análise da influência dos fatores no desempenho das variáveis resposta.

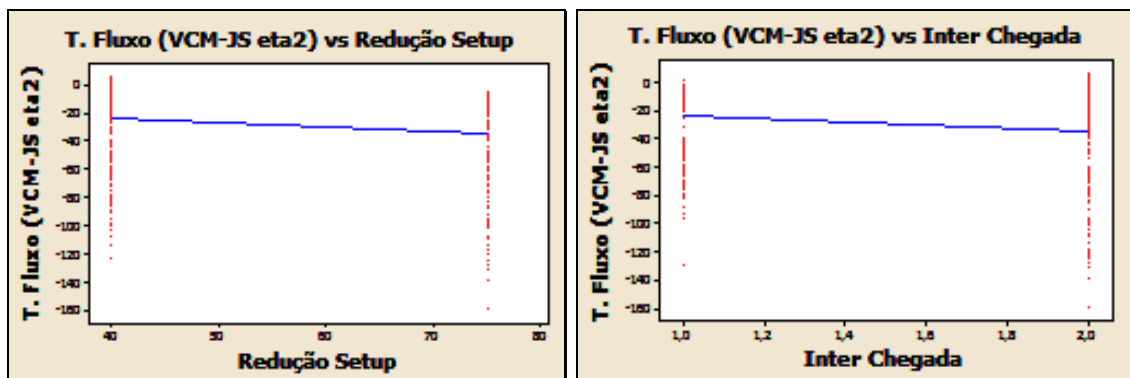


Figura 4.20 Representação com o gráfico “*Scatterplots*” com regressão linear

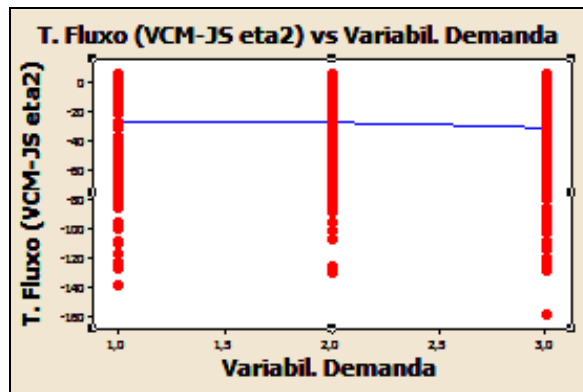


Figura 4.21 Representação com o gráfico “Scatterplots” com regressão quadrática

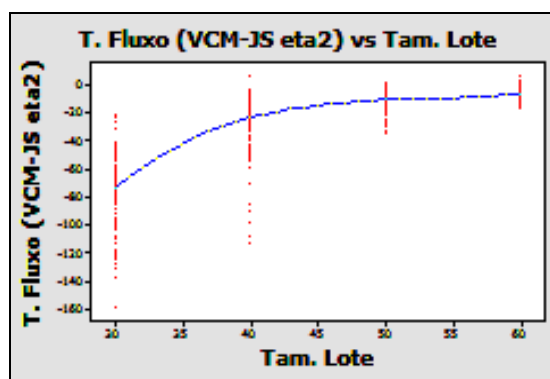


Figura 4.22 Representação com o gráfico “Scatterplots” com regressão cúbica

### 4.3 Recursos Empregados no Desenvolvimento da Tese

Os recursos empregados nesta tese resumiram-se basicamente a:

- 1 Microcomputador Pentium IV 512 Megabytes
- 1 licença do software Arena 9.0
- 1 licença do software MINITAB 14
- 1 licença do Microsoft Office (Excel e Word) 2003

Todos esses recursos se encontravam disponíveis e acessíveis no laboratório GRIMA/GRUCON do Departamento de Engenharia Mecânica da Universidade Federal de Santa Catarina.

### 4.4 Conclusões

Este capítulo descreveu o modelo conceitual desse trabalho de tese. Foram descritos o padrão e modelos adotados, o projeto de experimentos utilizado, a metodologia empregada na geração, coleta e transferência dos dados da simulação, o tratamento empregado com os



dados através dos softwares Excel e Minitab, e finalmente os recursos utilizados no desenvolvimento da tese. Neste capítulo não foi apresentada a metodologia adotada para a validação e verificação dos modelos físicos desenvolvidos, uma vez que ela é demasiadamente ampla e complexa, requerendo por esse motivo e por sua importância um capítulo específico para esta finalidade, o qual será o capítulo seis. Por outro lado, no próximo capítulo são descritos os modelos físicos desenvolvidos neste trabalho.

## **CAPÍTULO 5**

### **DESCRIÇÃO DOS MODELOS FÍSICOS**

Este capítulo descreve os modelos físicos desenvolvidos na tese, os quais foram apresentados inicialmente no item 4.1, que correspondeu ao modelo conceitual. Busca-se basicamente descrever a modelagem realizada através do software de simulação ARENA versão 9.0 dos três sistemas de manufatura contemplados, ou seja, o que emprega a configuração *job shop*, o que emprega a configuração celular tradicional e a que utiliza configuração celular virtual.

O desenvolvimento e implementação desses modelos de simulação permitiram avaliar as vantagens da aplicação da nova concepção de manufatura virtual apresentada nessa tese em comparação com os outros dois sistemas. A estrutura desse capítulo é semelhante à do item 4.1, no qual foi descrito o modelo conceitual.

A descrição mais específica do modelo desenvolvido neste trabalho se encontra no Apêndice 2, principalmente devido às especificidades do mesmo, além dessa descrição ser significativamente técnica e repetitiva em termos de elementos associados ao software Arena. Sempre que possível, a descrição dos módulos é feita da esquerda para a direita, de cima para baixo. Ao descrever os módulos, ou conjunto de módulos que aparecem repetidos nos modelos descritos, visando facilitar e tornar mais concisa a explicação, optou-se muitas vezes por descrevê-los como aparecem em uma célula ou departamento, deixando para o leitor a generalização para todo o modelo. As funções, os parâmetros e os módulos do software Arena são representados no texto em negrito, enquanto os nomes dos módulos, das variáveis, dos atributos das entidades, dos recursos, dos conjuntos de recursos etc. foram escritos entre aspas.

#### **5.1 Descrição Geral do Padrão Comum aos Três Modelos**

Nos próximos subitens descrevem-se algumas características que são comuns aos três modelos, como é o caso de algumas variáveis do sistema, da geração dos roteiros de fabricação dos tipos de peças e de suas demandas, os recursos e os conjuntos de recursos.

##### **5.1.1 Algumas variáveis do sistema**

Algumas variáveis foram criadas para permitir ao usuário alterá-las antes de cada simulação, permitindo que sejam facilmente alterados os níveis dos fatores, reduzindo-se o tempo de obtenção dos resultados correspondentes ao fatorial completo do projeto de

experimentos, conforme mostra a figura 5.1. Listam-se a seguir essas variáveis, seus parâmetros e o seu significado no modelo:

- “q” – seleciona o nível do fator tamanho do lote, que pode admitir quatro níveis, ou seja, 30, 40, 50 e 60 peças.
- “eta” ( $\eta$ ) – seleciona o nível do fator flexibilidade dos operadores. O valor 1 indica flexibilidade nula (o operador só opera um tipo de máquina), o valor 2 indica dupla flexibilidade (o operador opera dois tipos de máquinas) e o valor 3 indica flexibilidade tripla (o operador opera três tipos de máquinas)
- “demand\_tipo” – seleciona o nível do fator variabilidade da demanda, conforme a tabela 4.3, que admite três níveis: o valor 1 indica variabilidade nula (demanda constante), o valor 2 indica  $\frac{1}{4}$  da variabilidade plena e o valor 3 indica variabilidade plena.
- “reductor\_setup” – seleciona o nível do fator redução de *setup*. O valor 0,6 indica redução do *setup* em 40%, enquanto o valor 0,25 indica redução de 75%.
- “e\_expo” e “e\_const” – selecionam o nível do fator variação nos intervalos entre chegadas de lotes de peças. A variável “e\_expo” com valor 0 e “e\_const” com valor 1 indicam variação nula (c, chegadas a intervalos constantes) e “e\_expo” com valor 1 e “e\_const” com valor 0 indicam variação alta (segundo uma função exponencial).

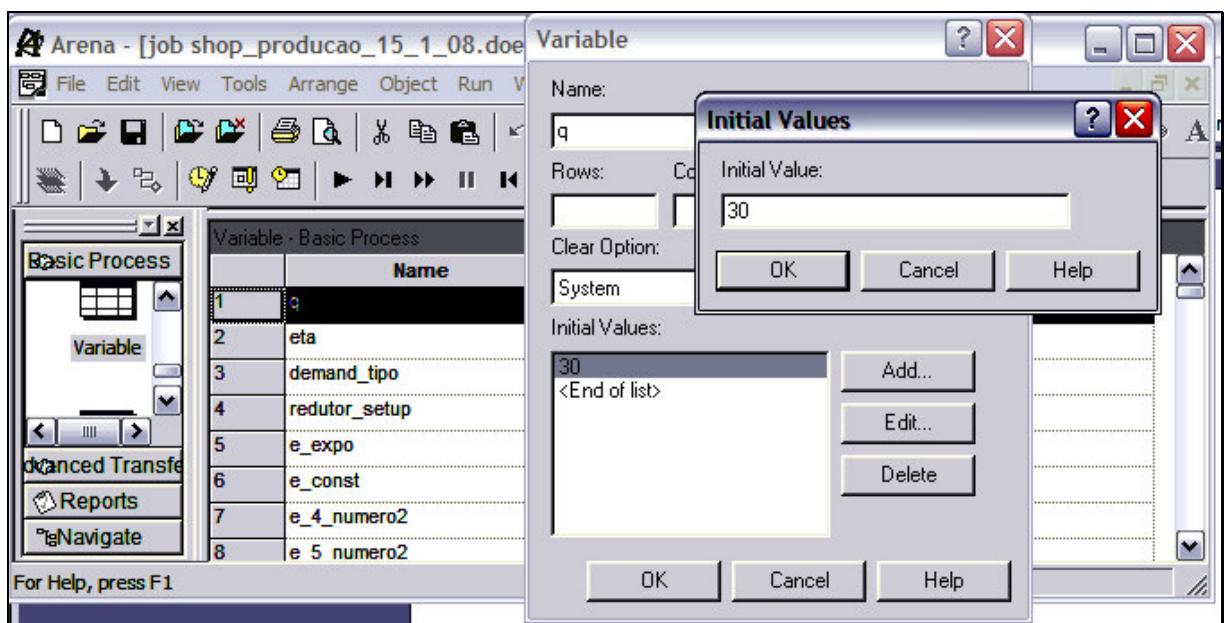


Figura 5.1 Tela do software Arena com algumas variáveis que definem os níveis dos fatores

Outras variáveis são importantes para armazenar informações que o sistema irá necessitar durante a execução da simulação. Entre elas estão as variáveis que irão guardar as demandas anuais de cada um dos cinquenta tipos diferentes de peças que são processados durante a simulação. Assim, foram definidas cinquenta variáveis cujos nomes são: “d\_p10” (demanda da peça 0 da família 1), “d\_p11”, “d\_p12”, “d\_p12”, “d\_p13”, “d\_p14”, “d\_p15”, “d\_p16”, “d\_p17”, “d\_p18”, “d\_p19”, “d\_p1A” e “d\_p1B” totalizando as doze peças da família de peças “família 1”. As demandas correspondentes às dez peças da “família 2” são representadas então por “d\_p20” até “d\_p29”. E com a mesma nomenclatura foram definidas as variáveis para as demandas das peças das demais famílias.

Muitas outras variáveis foram utilizadas nesses modelos e, ainda que estas variáveis sejam citadas nesse capítulo em vários módulos, elas serão explicadas com mais profundidade apenas no próximo capítulo, que correspondem à Validação e Verificação, uma vez que elas dizem respeito diretamente à tarefa de verificação do modelo.

#### 5.1.2 Os recursos e os conjuntos de recursos

Conforme citado anteriormente, os recursos empregados foram trinta e uma máquinas e vinte e um operadores. Todos os recursos eram de capacidade 1, ou seja, eram um só recurso, e não uma estação com vários recursos. As figuras 4.1 e 4.2 mostram as distribuições desses recursos respectivamente nas configurações *job shop* (também na celular virtual) e na celular tradicional.

As designações desses recursos foram feitas em conformidade com essas figuras. Por exemplo, os operadores foram designados de “operador\_a”, “operador\_b”,... “operador\_u”. As máquinas na configuração *job shop* (também na celular virtual) foram designadas como “Depto 01\_1”, “Depto 01\_2” (significando máquina 2 do departamento 01), ... “Depto 01\_4”, “Depto 02\_1” até “Depto 10\_3”. Como todas as máquinas em cada departamento eram idênticas, o índice final indicava apenas a numeração dentro do departamento. A figura 5.2 ilustra o módulo **Resource** dentro do **template Basic Process**. As máquinas na configuração celular tradicional foram designadas de forma idêntica, mas tiveram uma interpretação diferente, que foi a seguinte: a máquina “Depto 01\_4” significava a máquina do tipo 4 (que na configuração *job shop* estavam alocadas no departamento 4) da célula 1. Portanto, as máquinas da célula 1 foram: “Depto 01\_1”, “Depto 01\_2”, “Depto 01\_4”, “Depto 01\_5”, “Depto 01\_6” e “Depto 01\_8” conforme o padrão mostrado na figura 4.4.

Foram criados conjuntos de recursos, os quais englobavam os recursos descritos anteriormente, isto é, operadores e máquinas. Esses conjuntos eram os mesmos nas configurações *job shop* e celular virtual, porém eram distintos entre essas duas configurações e a celular tradicional. A nomenclatura empregada para as máquinas na

configuração *job shop* era “Depto 01”, que representa o **set** de máquinas do Departamento 1. A figura 5.3 mostra o módulo **set** de recursos da primeira configuração dentro do **template Basic Process**, com destaque para as máquinas que fazem parte desse primeiro conjunto de recursos. Havia então conjuntos de máquinas de “Depto 01” a “Depto 10”.

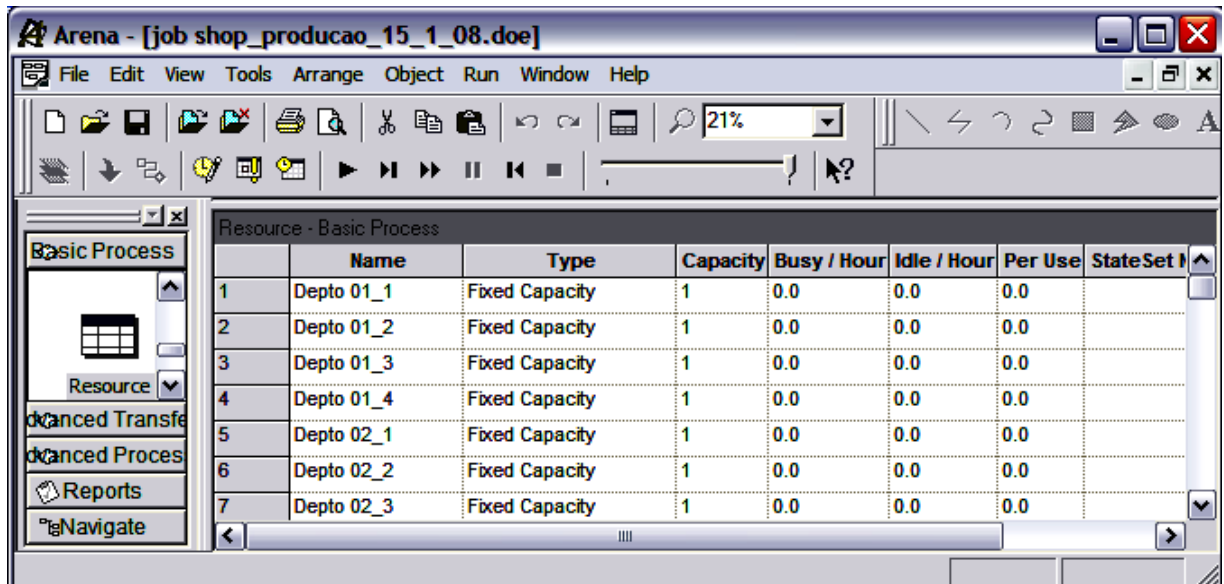


Figura 5.2 Detalhes do módulo **Resource** em uma tela parcial do software Arena

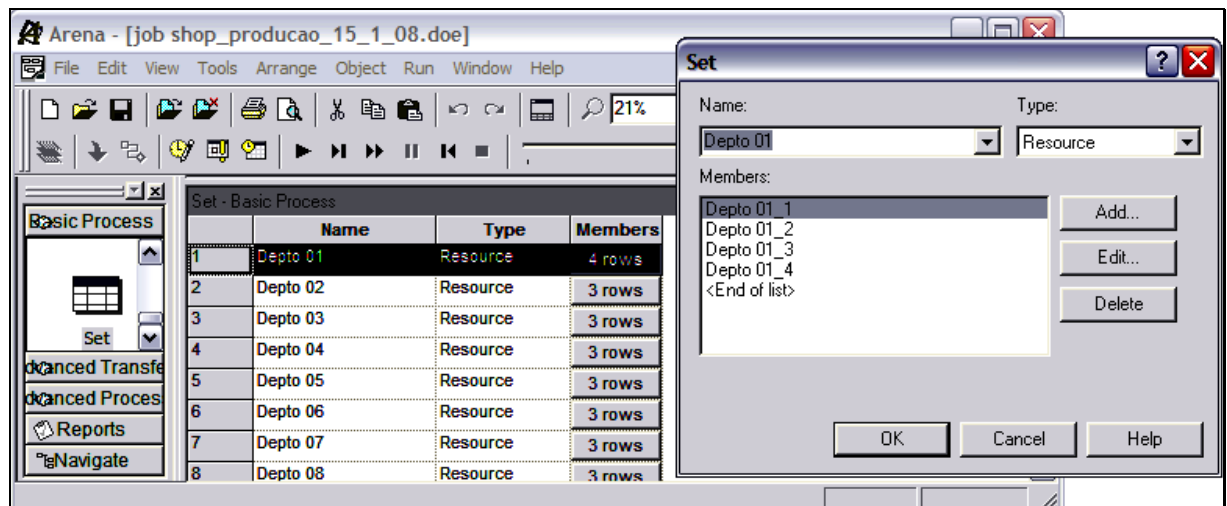


Figura 5.3 Os recursos do conjunto “Depto 01” do módulo **Set** da configuração *job shop*

Com relação aos conjuntos de operadores na configuração *job shop*, havia três nomenclaturas para definir estes **sets** de operadores em função do nível selecionado do fator flexibilidade dos operadores, abordado no subitem 4.2.1, correspondendo à maior ou

menor habilidade dos operadores em operar diferentes máquinas, ou *cross training*. O nível 1, selecionado pelo valor 1 da variável “eta”, fazia com que cada operador só operasse um tipo de máquina, sendo então empregados os **sets** “operador\_1”, “operador\_2”, ... “operador\_10”. Os recursos pertencentes a esses conjuntos estavam em conformidade com o padrão apresentado na tabela 4.7. Assim o **set** “operador\_1” contém os recursos “operador\_a”, “operador\_b” e “operador\_c”, conforme ilustra a figura 5.4. Para os demais níveis, “eta”=2 e “eta”=3, a nomenclatura desses **sets** foram respectivamente “eta\_2\_operador\_1”, “eta\_2\_operador\_2” etc. e “eta\_3\_operador\_1”, “eta\_3\_operador\_2” etc., conforme a tabela 4.7.

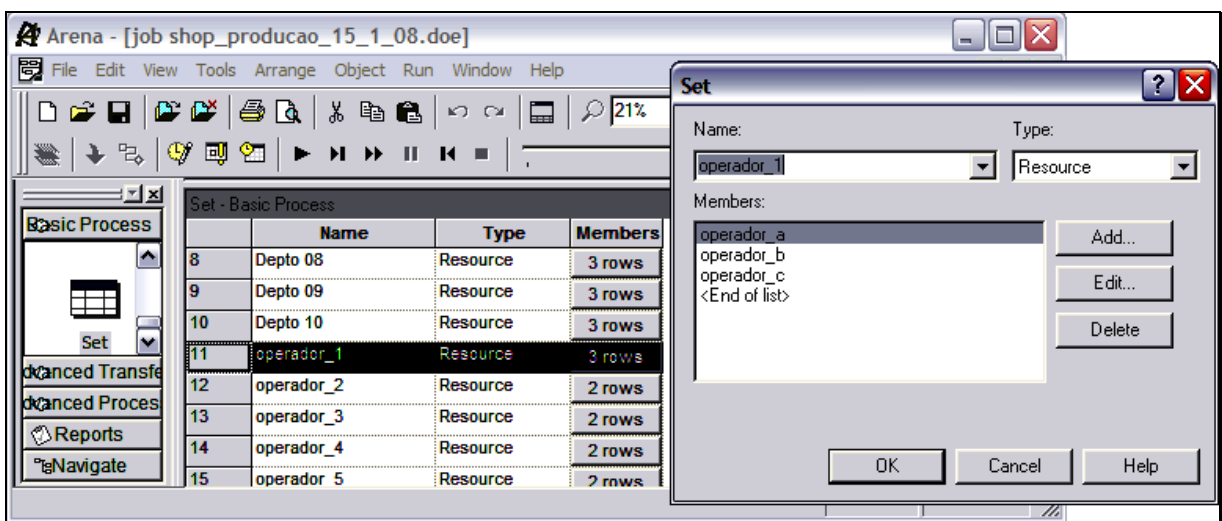


Figura 5.4 Os recursos do conjunto “operador\_1” do módulo **Set** da configuração *job shop*

Na configuração celular tradicional não havia conjuntos de máquinas, pois as entidades na simulação eram processadas inteiramente usando apenas a célula da sua família e nesses recursos não havia máquinas repetidas. Por outro lado, na configuração celular tradicional os **sets** de operadores só admitiam dois níveis do fator flexibilidade dos operadores, uma vez que no padrão adotado, como já foi citado anteriormente, esta configuração não admitia o nível da seleção “eta”=1. Assim, para “eta”=2 a nomenclatura utilizada foi “opr\_d1\_m1” (conjunto de operadores da célula 1 habilitados para operar a máquina do tipo 1 dessa célula). A título de exemplo, e levando em conta a tabela 4.7 e a figura 4.4, no caso de “opr\_d1\_m1” só o operador “operador\_a” estava habilitado, enquanto no caso de “opr\_d1\_m2” apenas os operadores “operador\_d” e “operador\_a” estavam habilitados. Os demais **sets** de operadores da célula 1 eram “opr\_d1\_m4”, “opr\_d1\_m5” e “opr\_d1\_m6” e os **sets** de operadores da configuração celular tradicional para “eta”=3 da célula 1 foram “ex\_3\_opr\_d1\_m1”, “ex\_3\_opr\_d1\_m2”, “ex\_3\_opr\_d1\_m4”,

“ex\_3\_opr\_d1\_m5” e “ex\_3\_opr\_d1\_m6”. Novamente a título de exemplo, no caso de “ex\_3\_opr\_d1\_m4”, apenas os operadores “operador\_h”, “operador\_d” e “operador\_a” estavam habilitados (ver figura 4.4). Esse mesmo padrão e lógica foi empregado para a nomenclatura das demais células nos dois níveis de flexibilidade de operadores.

### 5.1.3 Geração das demandas de cada tipo de peça

Este trecho do modelo físico é responsável por atribuir os valores de demanda para as 50 peças. Esses valores podem ser com variação nula,  $\frac{1}{4}$  ou plena, conforme a tabela 4.3. A finalidade é realizar a simulação com níveis diferentes do fator variabilidade da demanda. A figura 5.5 ilustra este trecho dos modelos, e o item A2.1 do Apêndice 2 os detalha com profundidade.

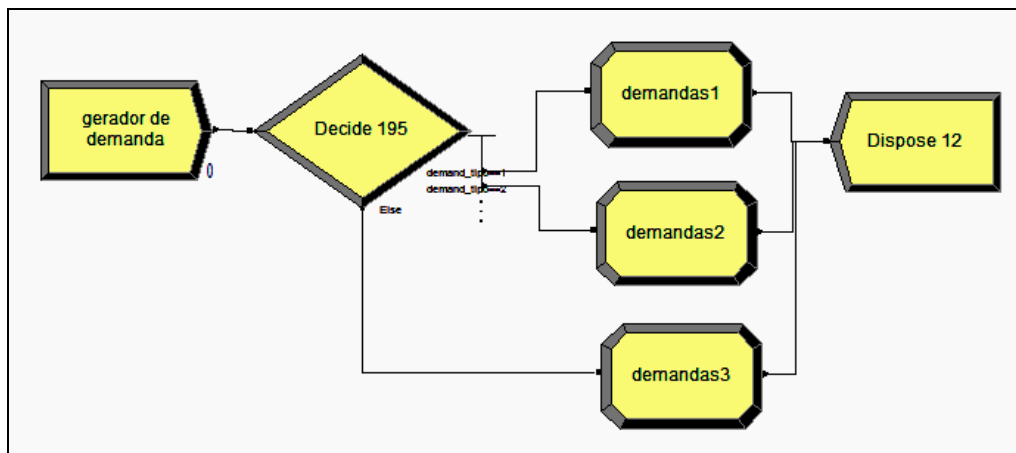


Figura 5.5 Conjunto de módulos responsáveis pela geração das demandas das peças

### 5.1.4 Geração dos lotes de peças e atribuição das seqüências de produção para cada peça

Este trecho do modelo físico é responsável pela geração de cada um dos lotes das cinquenta peças que compõem as cinco famílias de peças. Assim, por exemplo, no cálculo do intervalo entre chegadas de lotes da “peca\_10” pertencente à família 1, emprega a expressão  $EXPO((q/d_{p10}) * 3600) * e_{expo} + ((q/d_{p10}) * 3600) * e_{const}$  em horas. Nesta expressão “q” é a variável que contém o nível de tamanho do lote selecionado, “d\_p10” é a variável que armazena a demanda gerada para um tipo de peça (no exemplo a peça 10). Por outro lado, “e\_expo” e “e\_const” são as variáveis associadas aos dois níveis do fator variabilidade dos intervalos entre chegadas de peças, assumindo alternadamente os valores 0 e 1, os quais definem regimes constantes ou exponenciais desses intervalos, e o valor 3600 corresponde às horas de simulação (um ano aproximadamente).

Nesse trecho dos modelos atribui-se ainda aos lotes das peças o roteiro de fabricação específico do seu tipo de peça, segundo a tabela 4.2. A seguir encaminha o lote recém criado para o primeiro destino estabelecido no seu roteiro, sendo que no exemplo considerado, o primeiro destino do lote da peça 10 é a estação “Depto 01\_st”, com tempo de deslocamento igual a 10 minutos. As figuras 5.6 e 5.7 ilustram este trecho dos modelos, e o item A2.2 do Apêndice 2 o detalha com profundidade.

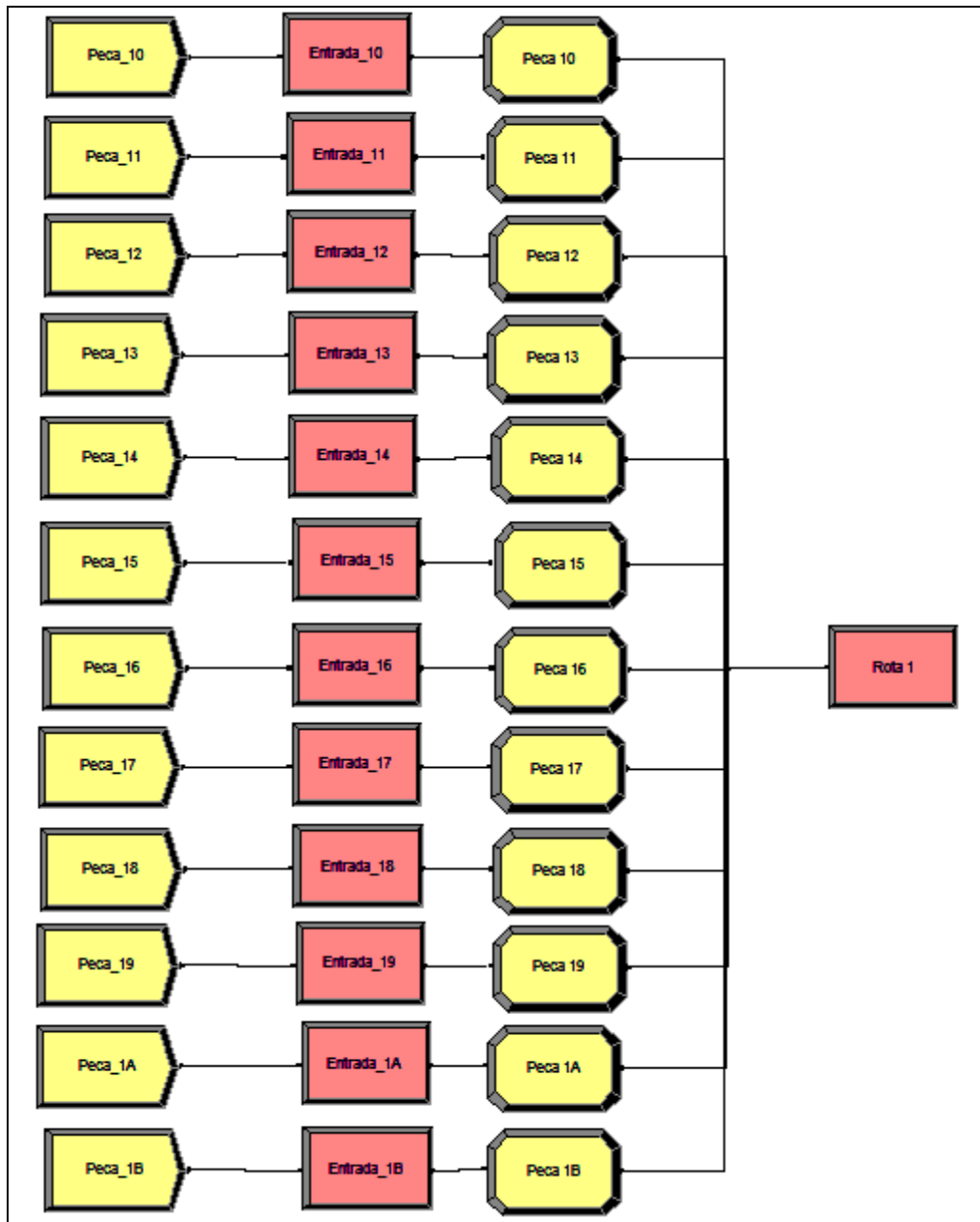


Figura 5.6 Conjunto de módulos responsáveis pela geração das peças da família 1



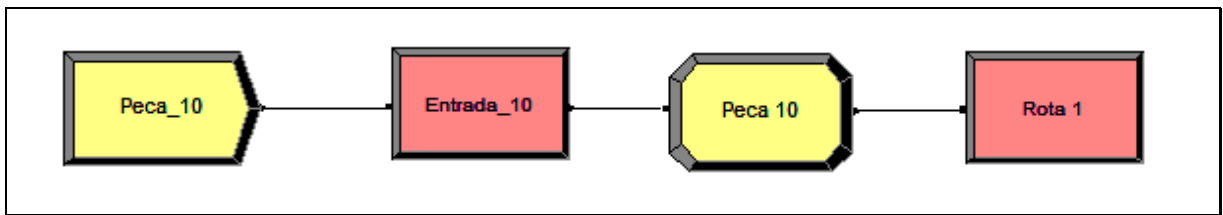


Figura 5.7 Conjunto de módulos responsáveis pela geração das peças “Peca\_10”

## 5.2 Descrição Específica do Modelo da Configuração *Job Shop*

### 5.2.1 Visão geral do modelo *job shop*

A figura 5.8 dá uma visão geral do modelo *job shop*. Procurou-se subdividir o modelo em trechos lógicos que facilitassem a sua compreensão. Há dois trechos já explicados no item anterior, que são “Geração de Demandas” e “Geração dos Lotes das Peças de uma família”. Os demais trechos são “Animação do Leiaute”, “Lógica de um Departamento” e “Saídas”. Que são abordados nos próximos subitens.

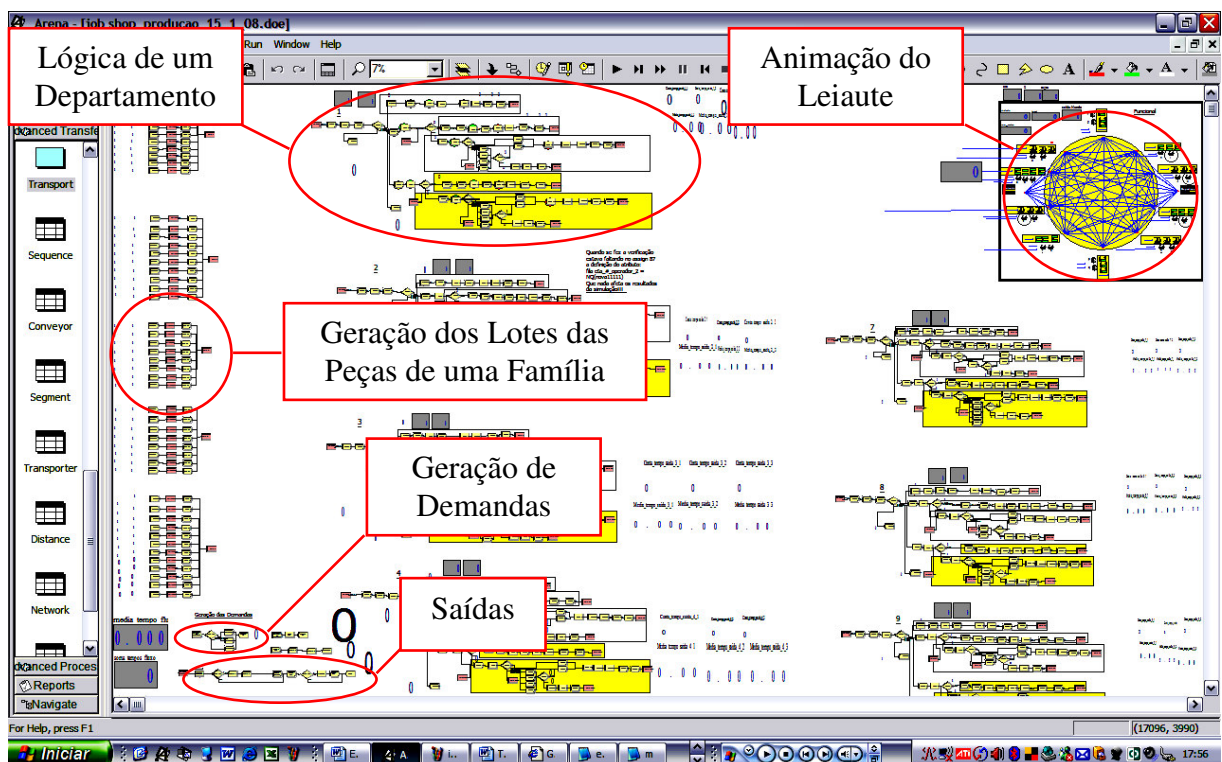


Figura 5.8 Visão geral do modelo da configuração *job shop*

### 5.2.2 Animação do leiaute *job shop*

A figura 5.9 mostra a animação do leiaute *job shop*. Os departamentos na figura estão representados pelos retângulos com máquinas em seu interior e com dois ou três

operadores próximos às máquinas. Cada departamento realiza um processo de fabricação diferente e por isso, em cada departamento todas as suas máquinas são iguais. Pode-se verificar que foram representados os dez departamentos, as trinta e uma máquinas e os vinte e um operadores, exatamente da forma estabelecida no padrão adotado e esquematizado na figura 4.1. As linhas que se entrecruzam no meio da figura correspondem às rotas que as entidades (isto é, os lotes de peças) percorriam durante a simulação, desde a entrada no sistema para os diversos departamentos, passando entre os departamentos e desses para a saída do sistema.

É fácil constatar que esta representação e modelo correspondem exatamente ao conceito de leiaute *job shop* uma vez que, como foi abordado no subitem 2.1.2, caracteriza-se por agrupar as máquinas por processo e, por este motivo, apresentar um fluxo de materiais bastante complexo, pois o roteiro de fabricação envolve departamentos distintos.

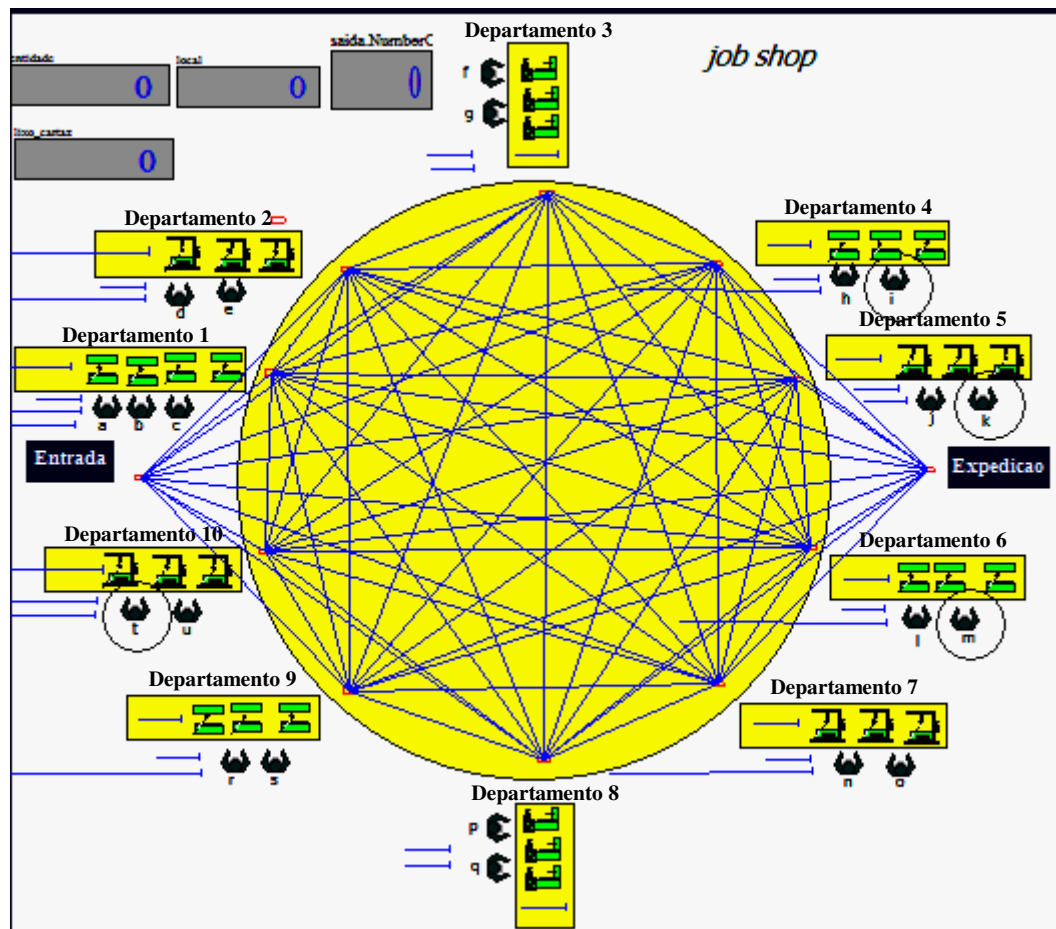


Figura 5.9 Animação do leiaute *job shop*

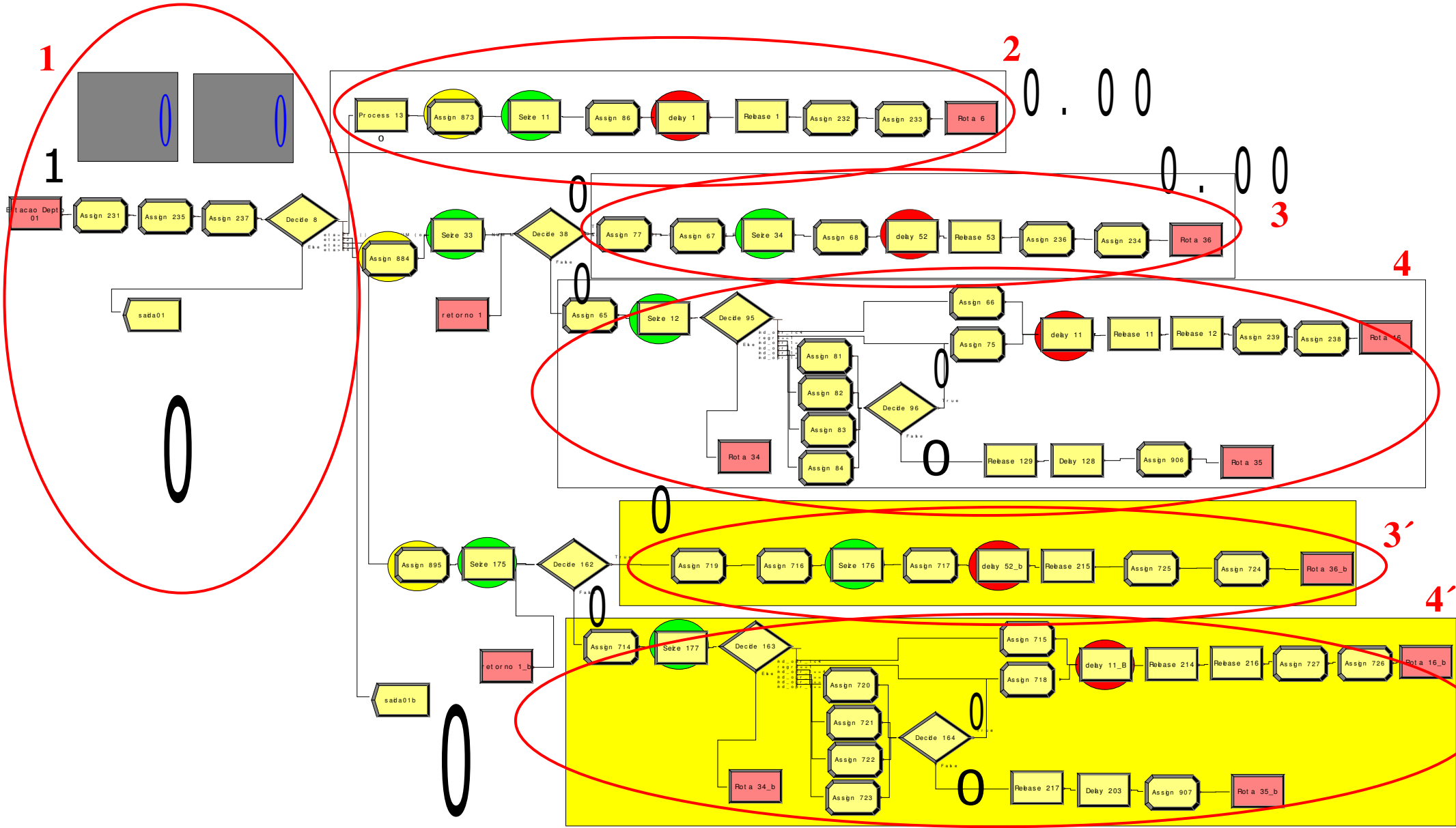


Figura 5.10 Conjunto de módulos que formavam um departamento

### 5.2.3 Lógica de um departamento.

A figura 5.10 mostra o trecho do modelo correspondente à lógica de um departamento na configuração *job shop*. Este trecho do modelo se repete dez vezes, uma para cada departamento. Para melhor compreensão da lógica do departamento, este trecho foi subdividido em quatro blocos: “Entrada”, “Sistema Livre”, “Obtida a máquina obtém operador interno” e “Obtida a máquina busca qualquer operador”, representados na figura 5.10 pelos números 1, 2, 3 (e 3’) e 4 (e 4’) respectivamente. Um detalhamento de cada um desses blocos é feito nos itens e figuras subseqüentes.

#### 5.2.3.1 O bloco “Entrada” na lógica de um departamento

Este bloco verifica o número de operadores que existem neste set (que é diferente se  $\eta$  é igual a 1, 2 ou 3). Verifica também o número de operadores que estão ocupados neste set. Por outro lado obtém, o número total de máquinas do departamento, ainda obtém o número de máquinas ocupadas. A seguir testa as condições múltiplas descritas abaixo:

1. Caso “ $\eta$ ” seja igual a 1, isto é, flexibilidade nula, **OU** havendo operador **E** máquina disponíveis, encaminha-se a entidade (o lote de peças) para o bloco “Sistema Livre”. Esta segunda parte do “OU” corresponde à especificação do padrão adotado que estabelece que quando uma ordem chega a um departamento, se houver máquina e operador disponíveis, imediatamente se inicia o processamento.
2. Caso contrário, referente ao item 1, testa-se se “ $\eta=2$ ”, ou seja, se o *Cross training* ( $\eta$ ) for igual a 2 (isto é, flexibilidade dupla). Em caso afirmativo encaminha-se a entidade (o lote de peças) para o trecho em que se encontram os blocos “Espera de máquina com operador Livre” e “Espera de máquina e de operador”, referentes ao *Cross training* ( $\eta$ ) igual a 2, ou seja, o trecho que antecede imediatamente os blocos 3 e 4 da figura 5.10.
3. Caso contrário, no que se referem os itens 1 e 2 acima, faz-se o teste “ $\eta=3$ ”, ou seja, se o *Cross training* ( $\eta$ ) for igual a 3 (flexibilidade tripla). Em caso afirmativo encaminha-se a entidade (o lote de peças) para o trecho em que se encontram os blocos “Espera de máquina com operador Livre” e “Espera de máquina e de operador” referentes ao *Cross training* ( $\eta$ ) igual a 3, ou seja, o trecho que antecede imediatamente os blocos 3’ e 4’ da figura 5.10.

O item A2.3 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

### 5.2.3.2 O bloco “Sistema Livre” na lógica de um departamento.

Este bloco monitora o número de entidades que estão entrando neste trecho da simulação e identifica a família do lote de peças. Ele solicita um recurso com prioridade alta e obtém um operador e uma máquina. A máquina obtida é a primeira encontrada livre por ordem de cadastramento na planilha deste **set**. Assim, no exemplo considerado, é selecionada a primeira máquina livre do **set** “Depto 01”, na ordem de cadastro, ou seja, “Depto 01\_1”, “Depto 01\_2”, “Depto 01\_3” ou “Depto 01\_4”, conforme mostra a figura 5.3. O mesmo critério é seguido na seleção do operador. A seguir realiza-se o *setup* (no caso pleno) em “expo(3)” horas e todo o lote é processado em “expo(0.1\*q)” horas. Também calcula-se o tempo médio de uso dos recursos no bloco. Finalmente encaminha-se a entidade (isto é, o lote de peças) para a próxima operação (ou para a saída, no caso de encerramento de todas as operações) segundo o seu roteiro de fabricação, com tempo de rota dado pela função “UNIF(30, 60)” minutos.

O item A2.4 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

### 5.2.3.3 O bloco “Blocos intermediários” na lógica de um departamento

A figura 5.11 apresenta um dos dois blocos intermediários que aparecem na figura 5.10, localizados entre o bloco “Entrada” e os blocos “Obtida a máquina obtém operador interno” e “Obtida a máquina busca qualquer operador”, respectivamente representados pelos números 1, 3 (e 3´) e 4 (e 4´). Esses dois blocos intermediários são idênticos e são usados, um quando “eta” igual a 2 e outro quando “eta” é igual a 3, para quando ocorre de uma entidade chegar num departamento e não haver disponibilidade nem de operadores nem de máquinas. O item A2.5 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

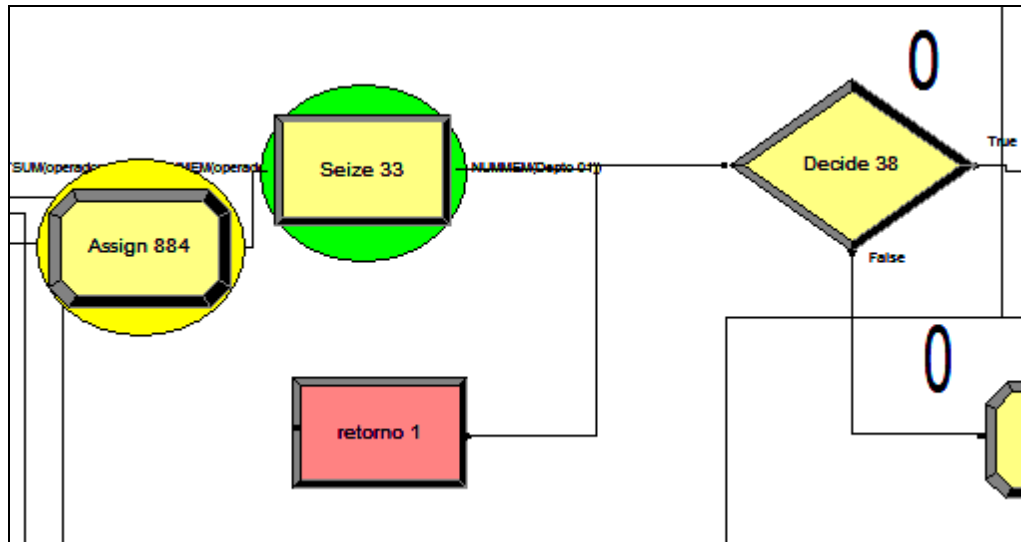


Figura 5.11 Os blocos “intermediários” na lógica de um departamento

5.2.3.4 O bloco “Obtida a máquina obtém operador interno” na lógica de um departamento.

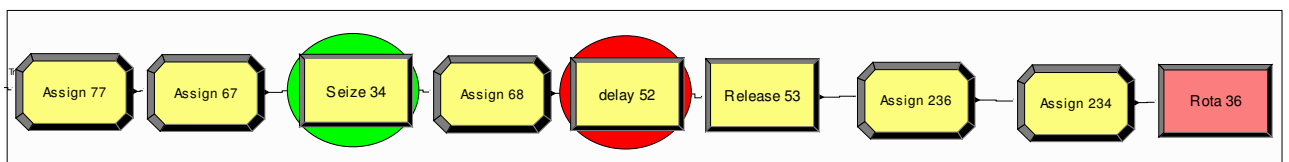


Figura 5.12 Módulos que compõem o bloco “Obtida a máquina obtém operador interno”

O Bloco “Obtida a máquina obtém operador interno”, ilustrado na figura 5.12, busca fazer com que a entidade que acaba de conseguir uma máquina (para realizar a próxima etapa da seqüência de operações) obtenha via seize um operador interno que esteja livre. Caso de mais de um lote solicitar o mesmo operador, é dada prioridade ao que está numa fila maior. O operador interno obtido é o primeiro encontrado livre por ordem de cadastramento na planilha do set de operadores internos desse departamento. A obtenção de um operador interno evita tempos de ida e volta de operador(es) de outro departamento. A seguir é realizado o setup (no caso pleno) em “expo(3)” horas e todo o lote é processado em “expo(0.1\*q)” horas. A máquina é então liberada, e o operador encaminha a entidade (o lote de peças) para a próxima operação (ou à saída, se todas as operações já foram encerradas) segundo o seu roteiro de fabricação. O tempo de rota é dado pela função “UNIF( 30, 60)” minutos. O item A2.6 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

5.2.3.5 O bloco “Obtida a máquina busca qualquer operador” na lógica de um departamento.

O bloco “Obtida a máquina busca qualquer operador”, ilustrado na figura 5.13, é o maior do trecho da “lógica de um departamento”. A entidade que acaba de conseguir uma

máquina, para realizar a próxima etapa da seqüência de operações, não dispondo de nenhum operador interno (do próprio departamento) disponível, tenta obter (através de um **seize**) qualquer operador (interno ou externo habilitado). Em caso de mais de um lote solicitar o mesmo operador, é dada prioridade àquele que está numa fila maior (LNQ - *Longest number of jobs*), e em caso de empate neste critério a prioridade é dada àquele que tenha o menor tempo de processamento. Caso não haja nenhum operador disponível, a entidade permanece na fila para obter um desses operadores. Este tipo de bloco é usado tanto para “eta” = 2 como “eta” =3, correspondendo respectivamente aos números 4 e 4’ na figura 5.10.

Quando o operador é obtido, considera-se o tempo de *setup* pleno dado pela expressão “expo(3)” horas. Caso o operador obtido seja externo, é considerado um tempo de movimentação do mesmo até o respectivo departamento com valor dado pela expressão “UNIF(0.5,1)” horas. Depois é simulado o tempo de processamento do lote dado pela expressão “expo(0.1\*q)” horas. Na seqüência tanto a máquina como o operador são liberados. Finalmente, a entidade (isto é, o lote de peças) é encaminhada para a próxima operação (ou à saída, se já foram realizadas todas as operações) segundo o seu roteiro de fabricação, com o tempo de rota dado pela expressão “UNIF(30, 60)” minutos. O item A2.7 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

#### 5.2.4 O trecho “Saídas” na configuração *job shop*.

A figura 5.14 mostra o trecho “Saídas” na configuração *job shop*, e para facilitar a descrição desse trecho, o mesmo foi dividido em três blocos, os quais são descritos nos próximos itens.

##### 5.2.4.1 O bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos”

A figura 5.15 mostra o bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos”. Este bloco é responsável pela coleta de dados estatísticos referentes à ocupação dos operadores e do número de lotes de peças que saíram durante a simulação. O item A2.8 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

##### 5.2.4.2 O bloco “Verificação de necessidade de *Warm up*”

A figura 5.16 mostra o bloco “Verificação de necessidade de *Warm up*”. A finalidade desse bloco é avaliar o tempo de *Warm up* adotado através do número médio de peças produzidas por hora. O item A2.9 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

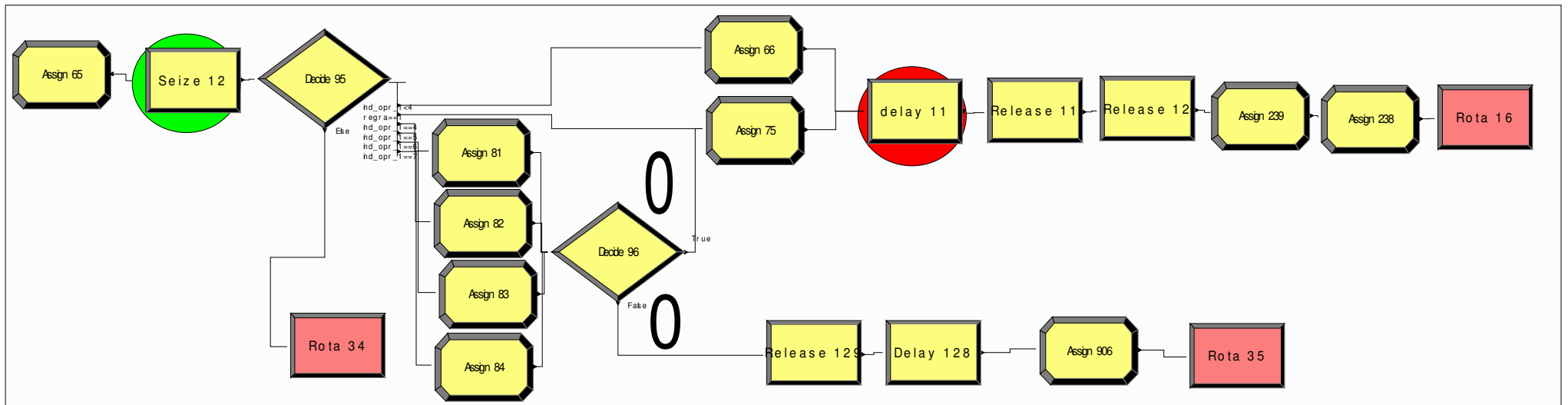


Figura 5.13 Os blocos “Obtida a máquina busca qualquer operador”



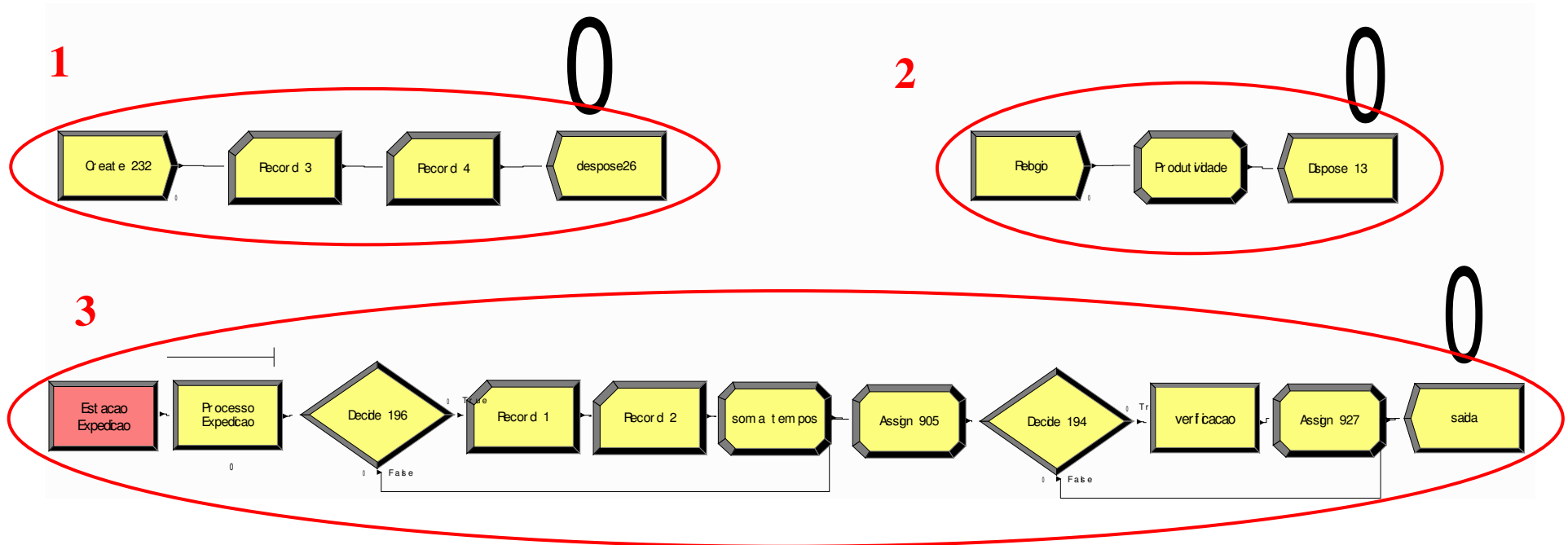


Figura 5.14 Os blocos no trecho “Saídas” na configuração *job shop*

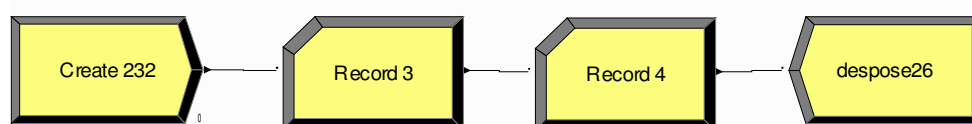


Figura 5.15 O bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos”

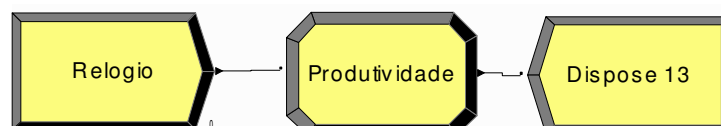


Figura 5.16 O bloco “Verificação de necessidade de *Warm up*”

#### 5.2.4.3 O bloco “Saída das Entidades”

A figura 5.17 mostra o bloco “Saída das Entidades”. Este bloco tem por finalidade recolher os tempos que cada entidade (lote) permanece na simulação, ou seja, o tempo de fluxo (*flow Time*), assim como os tempos de fila de todas as entidades (lotes) que saem do sistema durante a simulação. O item A2.10 do Apêndice 2 detalha com profundidade este trecho do modelo *job shop*.

### 5.3 Descrição Específica do Modelo Celular Tradicional

#### 5.3.1 Visão geral do modelo celular tradicional

A figura 5.18 apresenta uma visão geral do modelo celular tradicional no software Arena. Procurou-se subdividir o modelo em trechos lógicos que facilitassem a sua compreensão. Há dois trechos já explicados anteriormente, que são “Geração de Demandas” e “Geração dos Lotes das Peças de uma família”. Os demais trechos são “Animação do Leiaute”, “Geração de operador sempre ocupado”, “Lógica de uma Célula”, “Saídas do Sistema” e “Transferência de dados para o Excel”. Esses blocos são abordados nos próximos subitens.

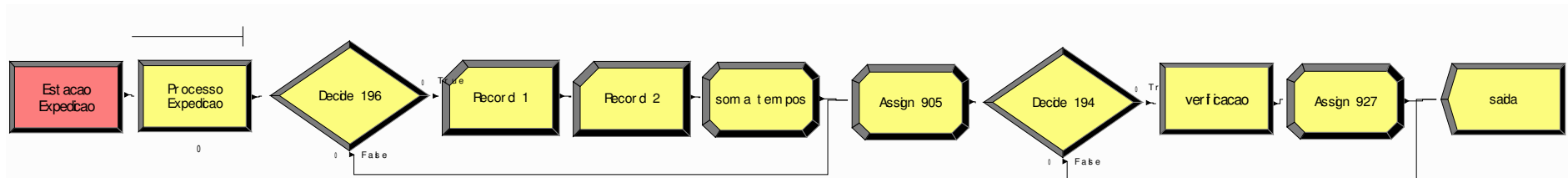


Figura 5.17 O bloco “Saída das Entidades”

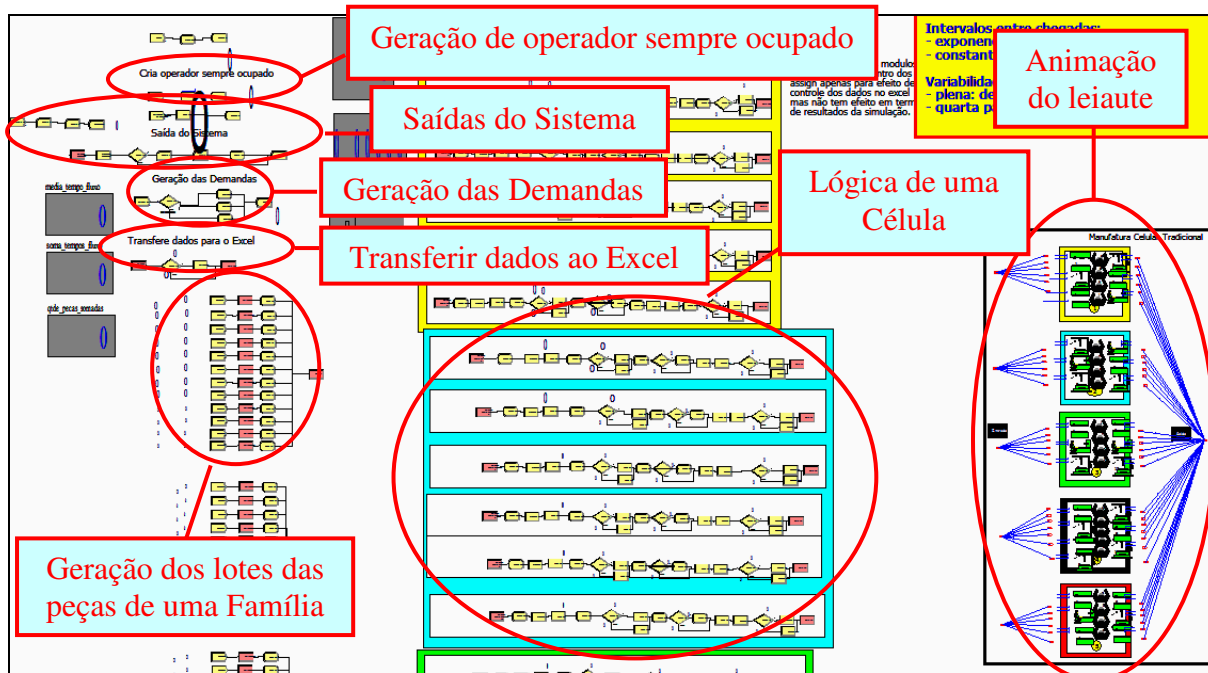


Figura 5.18 Visão geral do modelo celular tradicional

### 5.3.2 Animação do leiaute celular tradicional

A figura 5.19 mostra a animação do leiaute celular tradicional. As células na figura estão representadas pelos retângulos com seis ou sete máquinas e quatro ou cinco operadores em cada uma. Pode-se verificar que estão representadas as cinco células, as trinta e uma máquinas e os vinte e um operadores, exatamente da forma estabelecida no padrão adotado e esquematizado na figura 4.4. As linhas na figura correspondem às rotas que as entidades (os lotes de peças) percorrem da entrada do sistema para as diversas células, e delas para a saída do sistema.

### 5.3.3 Geração de operador sempre ocupado

A figura 5.20 mostra o bloco “Geração de operador sempre ocupado”. O objetivo deste bloco é criar um operador chamado “operador\_sempre\_ocupado”, que vai permanecer ocupado até o fim da simulação. Conforme apontado no subitem 4.1.4, o número de operadores das células tradicionais é menor que a quantidade de máquinas a serem operadas, que por outro lado são todas de tipos distintos. Devido a isto, não é factível o uso das células tradicionais para  $\eta = 1$ , ou seja, quando o operador só está habilitado para operar uma única máquina. Por outro lado, para  $\eta = 2$  ou  $3$ , cada operador fica próximo da máquina para a qual ele está mais habilitado (esta sua característica operacional é denominada de “primeira habilidade”).

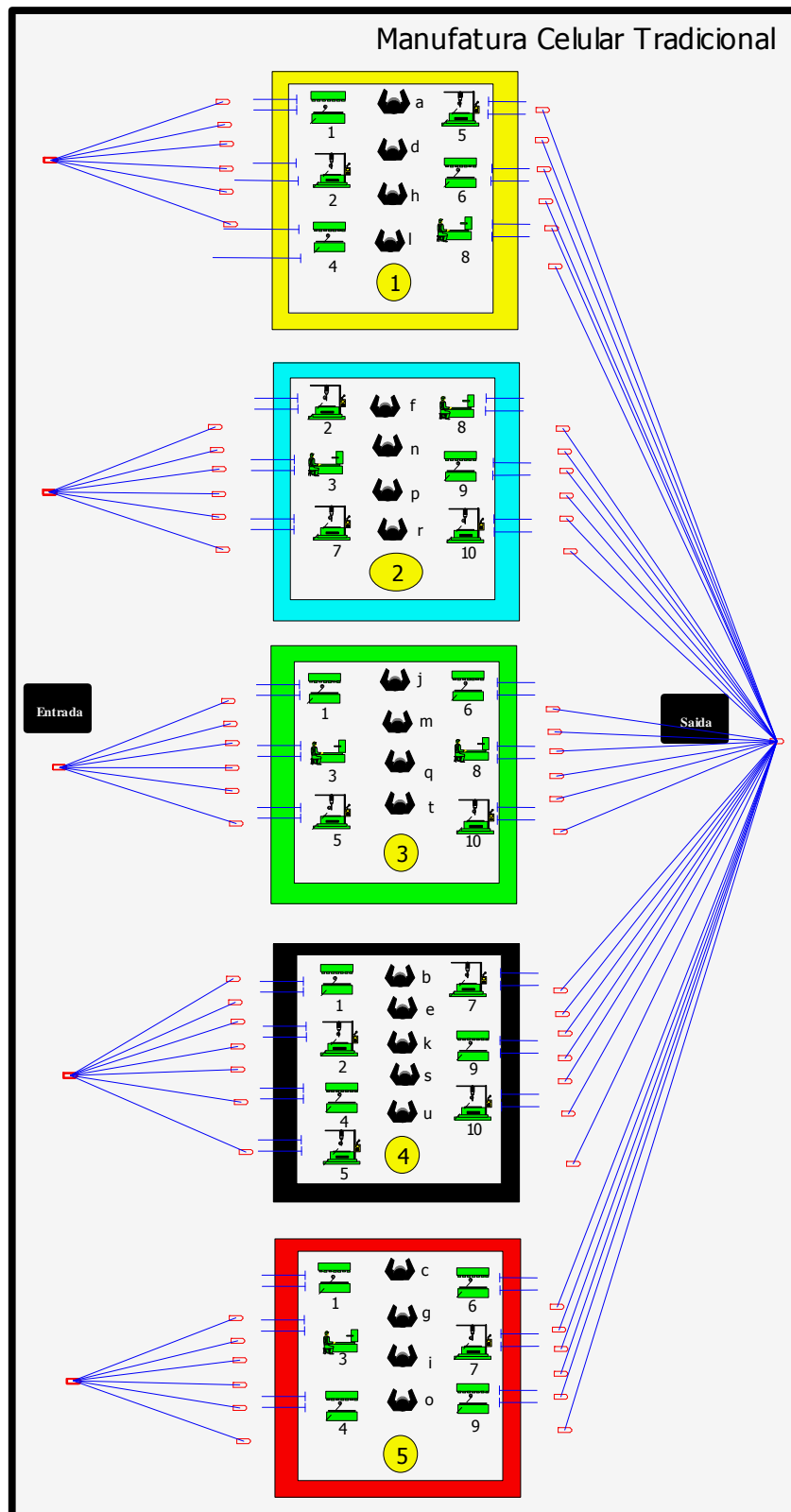


Figura 5.19 Animação do leiaute celular tradicional

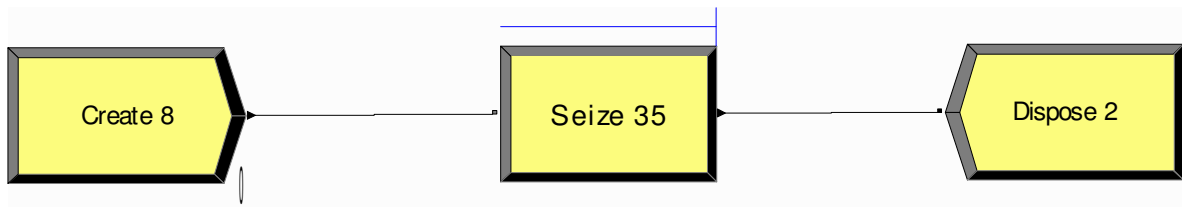


Figura 5.20 Geração de operador sempre ocupado

Assim, quando uma entidade (um lote) solicita um operador para acionar a máquina já obtida, ele dá preferência ao operador que a opera como sua primeira habilidade, pois dessa forma se economiza o tempo de movimentação do operador. São candidatos a operar cada máquina, além do operador com essa primeira habilidade, os demais operadores que têm esta habilidade como uma segunda ou terceira habilidade, mas que irão necessitar se movimentar até a máquina. Este conjunto de operadores tem uma nomenclatura própria para cada máquina a ser atendida, por exemplo, “oper\_d1\_m2” é o conjunto de operadores, para  $\eta = 2$ , que podem operar a “Depto 01\_2”, ou seja, a máquina 2 que se encontra na célula denominada “Depto 01”. Esses operadores concretamente são o operador\_d e o operador\_a, e que naturalmente se encontram na célula “Depto 01”, uma vez que nas células tradicionais não há mobilidade dos operadores entre as células, só internamente nas próprias células. Além disso, algumas das máquinas, porque elas são em maior número do que os operadores, não têm operador que as opere como a sua primeira habilidade. Na célula do “Depto 01” as máquinas “Depto 01\_5” e “Depto 01\_8” não têm esses operadores. Para esses casos se atribui o “operador\_sempre\_ocupado” como primeiro operador do conjunto de operadores para essas máquinas. Dessa forma, sempre que o operador do conjunto de operadores de uma determinada máquina não for o primeiro do conjunto significa que houve movimentação do operador, e esse tempo de movimentação deve ser considerado. Assim, a finalidade desse bloco é criar e disponibilizar um operador sempre ocupado para os conjuntos de operadores que não tenham um operador como primeira habilidade. O item A2.11 do Apêndice 2 descreve este trecho do modelo celular tradicional.

#### 5.3.4 Lógica de uma célula

A figura 5.21 mostra o trecho do modelo correspondente à lógica de uma célula na configuração celular tradicional. Este trecho do modelo se repete cinco vezes, uma para cada célula. Cada célula desse modelo é composta por seis ou sete fileiras de conjuntos de comandos, conforme se pode observar na figura 5.21. Cada uma dessas fileiras corresponde à lógica associada a cada uma das seis ou sete máquinas disponíveis em cada célula.

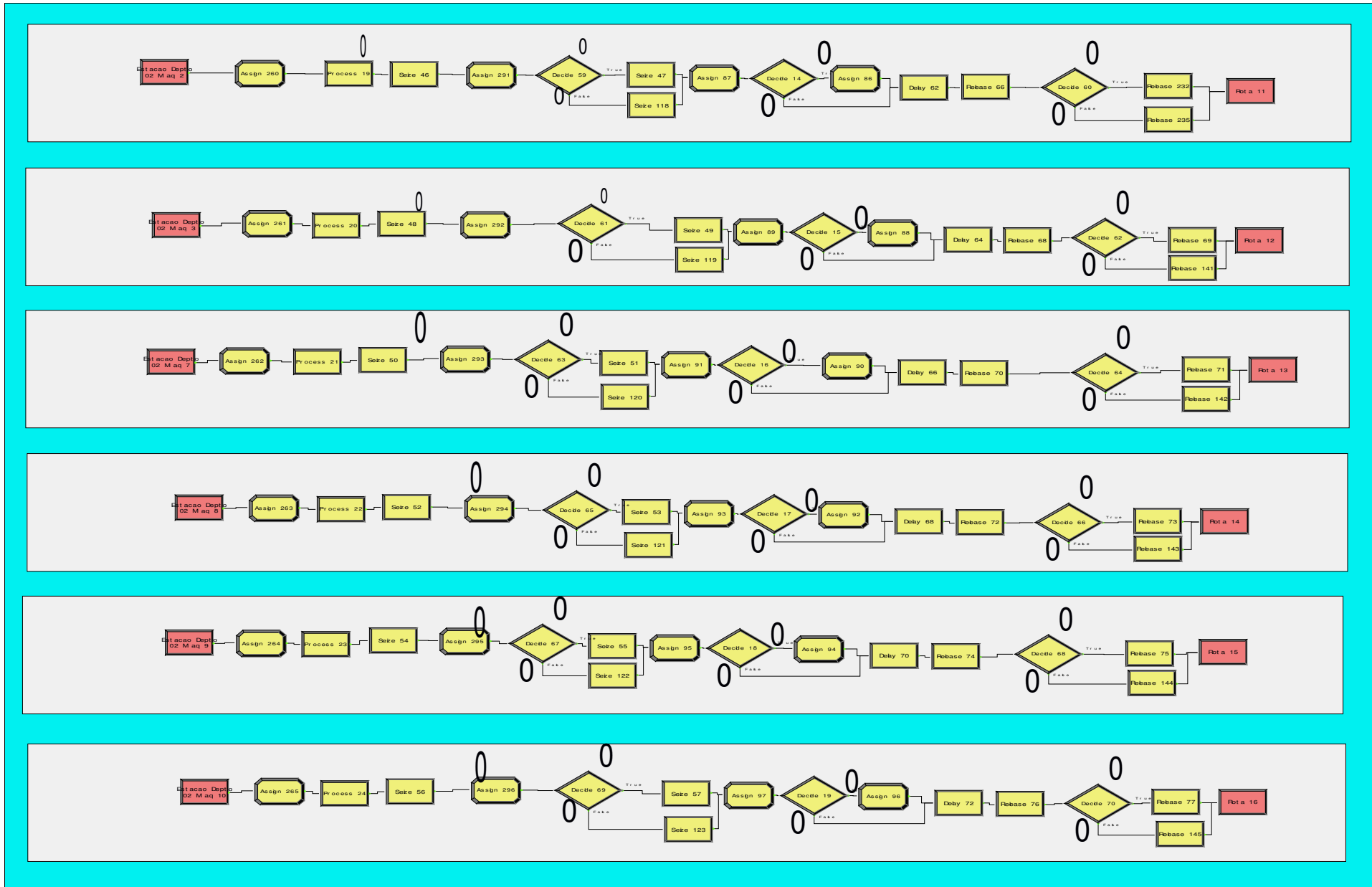


Figura 5.21 Conjunto de módulos que formam uma célula tradicional

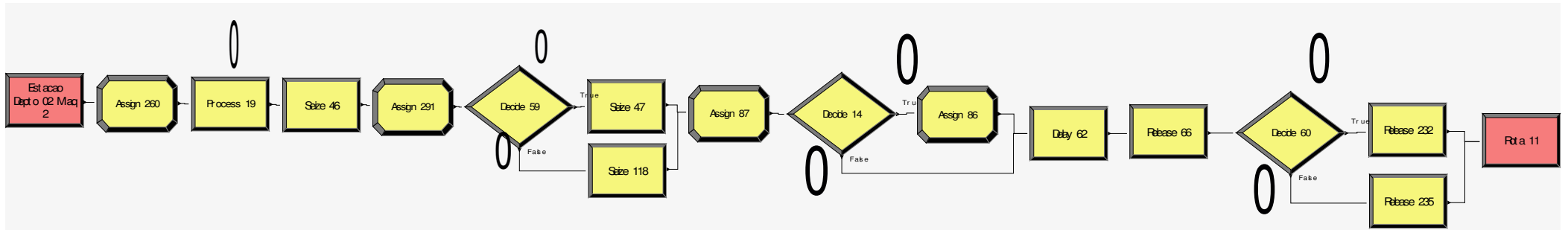


Figura 5.22 Lógica associada a cada uma das máquinas disponíveis em cada célula



O trecho do modelo apresentado na figura 5.22 representa a lógica de controle da máquina “Depto 02\_2”, isto é, a máquina da célula 2 que realiza o processo 2, além de prover um operador para operá-la. O tempo de processamento nessa máquina será definido neste bloco pela expressão “ $\text{expo}(0.1*q)$ ”. Caso mais de um lote deseje operar a máquina “Depto 02\_2”, é dada maior prioridade ao lote que esteja esperando em uma fila maior. Por outro lado, a fila é ordenada em função do menor tempo de processamento (SPT).

Na configuração celular tradicional, todos os **set** de operadores referem-se a operadores internos da própria célula. No exemplo o **set** “opr\_d2\_m2” é constituído pelos recursos “operador\_sempre\_ocupado” e “operador\_f”. Conforme mencionado anteriormente, esse primeiro operador está sempre indisponível. É atribuído o tempo de processamento de “ $\text{expo}(0.1*q)$ ” horas, um tempo de setup reduzido e um tempo de movimentação do operador de  $\text{unif}(0.05, 0.1)$  horas, caso o operador não opere a máquina como a sua primeira habilidade. A seguir a máquina e operador são liberados e o lote se dirige para a próxima etapa descrita na sua seqüência de operações, que normalmente corresponde a uma outra máquina da mesma célula, ou para a saída do sistema, caso a entidade já tenha concluído todas as operações previstas. O tempo de rota é dado pela função “ $\text{unif}(0.05, 0.1)$ ” horas. O item A2.12 do Apêndice 2 detalha com profundidade este trecho do modelo celular tradicional.

#### 5.3.5 Saídas do sistema.

A figura 5.23 mostra o trecho “Saídas do sistema” dentro da configuração celular tradicional. Há quatro blocos neste trecho do modelo: “Saída de dados”, “*Warm up*”, “Saídas de entidades” e “Transferência de dados para o Excel”, os quais são apresentados na figura 5.23 pelos números 1, 4, 2 e 3 respectivamente.

O bloco “*Warm up*” já foi apresentado no subitem 5.2.4.2, e o bloco “saída de dados” também já foi abordado e encontra-se no subitem 5.2.4.1. Como os blocos “Saída de entidades” e “Transferência de dados para o Excel” foram abordados anteriormente de maneira parcial, eles são descritos nos próximos subitens.

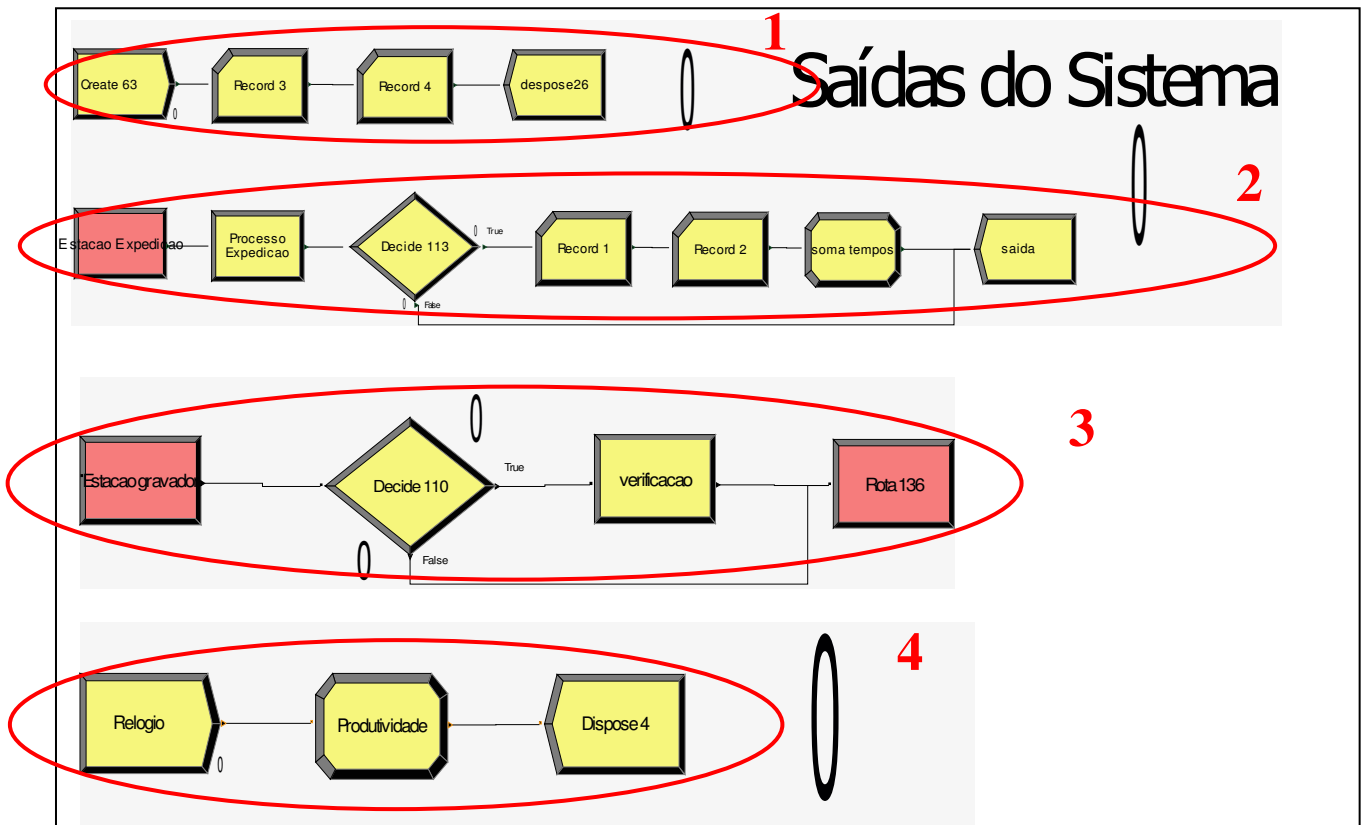


Figura 5.23 Blocos associados a saídas do Sistema

5.3.5.1 O bloco “Saída das Entidades”

A figura 5.24 mostra o bloco “Saída das Entidades”, no qual são recolhidos alguns tempos da simulação. Esses tempos são obtidos e armazenados após o tempo de *warm up* de 1800 horas. Um desses dados é o tempo de fluxo dos lotes na célula, ou seja, o tempo de fluxo (*flow Time*). Outro dado recolhido são os tempos de fila em que cada entidade (lote) permanece durante a simulação. O item A2.13 do Apêndice 2 detalha com profundidade este trecho do modelo celular tradicional.

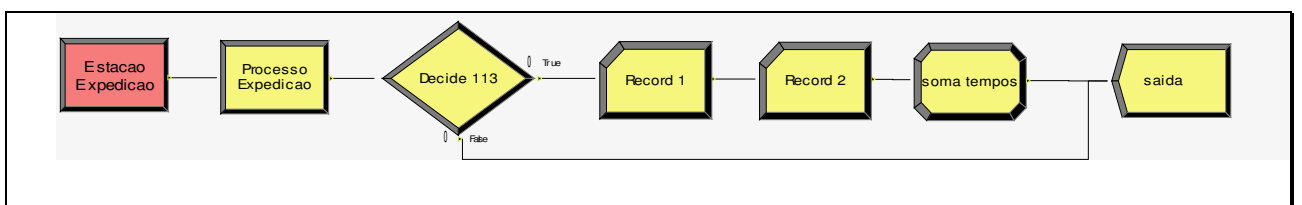


Figura 5.24 O bloco “Saída das Entidades”

### 5.3.5.2 O bloco “Transferência de dados para o Excel”

A figura 5.25 mostra o bloco “Transferência de dados para o Excel”, o qual foi considerado como bloco de saída porque a sua finalidade é obter dados na saída de cada máquina e no fim das operações de uma entidade, enviando-os para uma planilha Excel. O item A2.14 do Apêndice 2 detalha com profundidade este trecho do modelo celular tradicional.

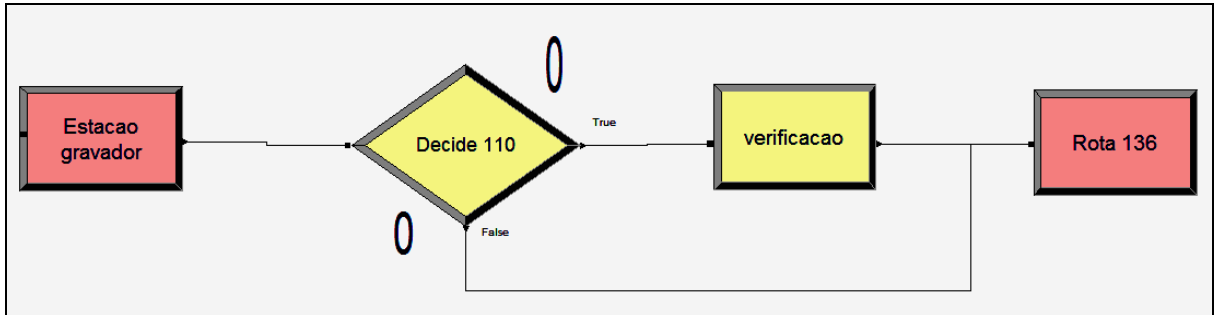


Figura 5.25 O bloco “Transferência de dados para o Excel”

## 5.4 Descrição Específica do Modelo Celular Virtual

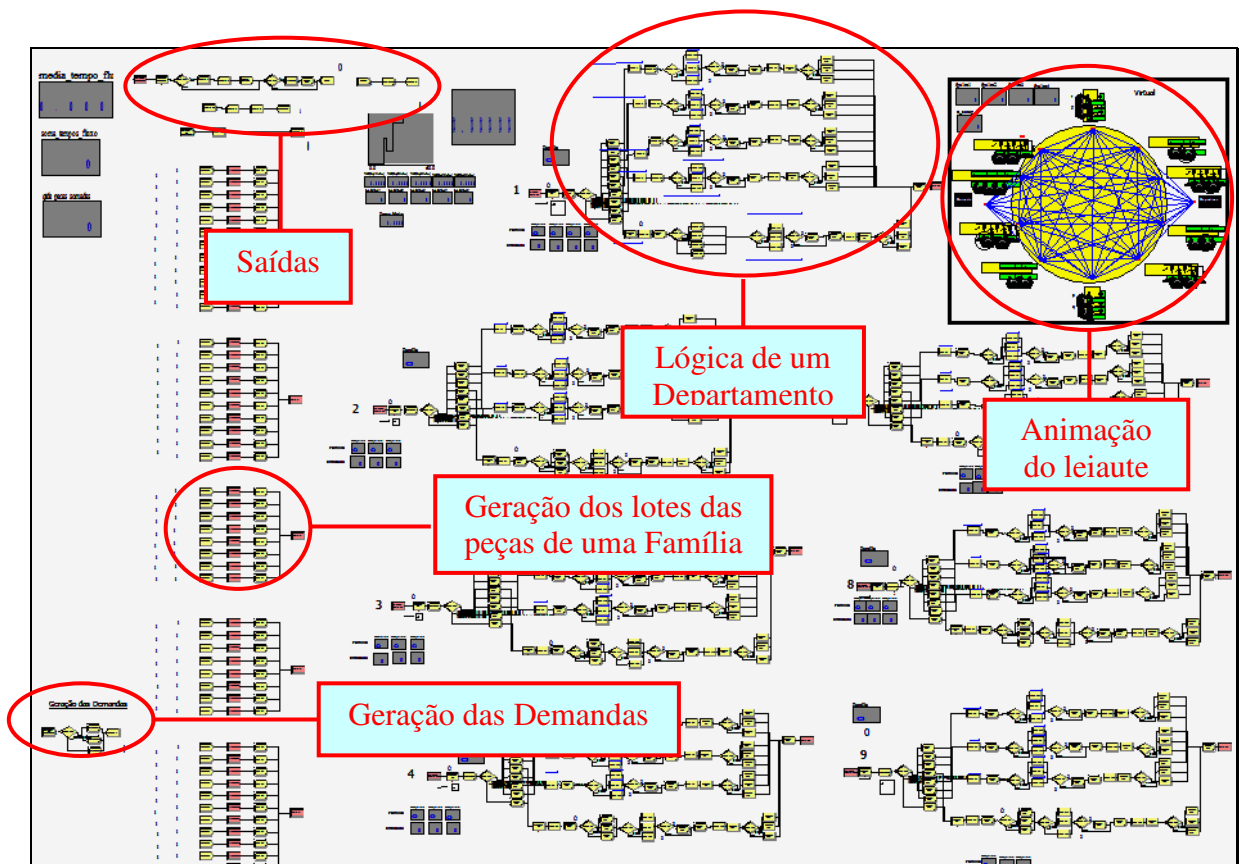


Figura 5.26 O bloco “Visão geral do modelo celular Virtual”

#### 5.4.1 Visão geral do modelo celular virtual

A figura 5.26 apresenta uma visão geral do modelo celular virtual. Procurou-se subdividir o modelo em trechos lógicos que facilitassem a sua compreensão. Há três trechos já explicados anteriormente, que são denominados “Geração de Demandas”, “Geração dos Lotes das Peças de uma família” e “Animação do leiaute”. Apesar deste último trecho (“Animação do leiaute”) não ser logicamente igual à configuração *job shop*, fisicamente ele é exatamente igual (figura 5.27), e por isso a descrição feita no subitem 5.2.2 se aplica também para o modelo celular virtual. Os demais trechos deste modelo são denominados “Lógica de um Departamento” e “Saídas”, os quais são descritos nos próximos subitens.

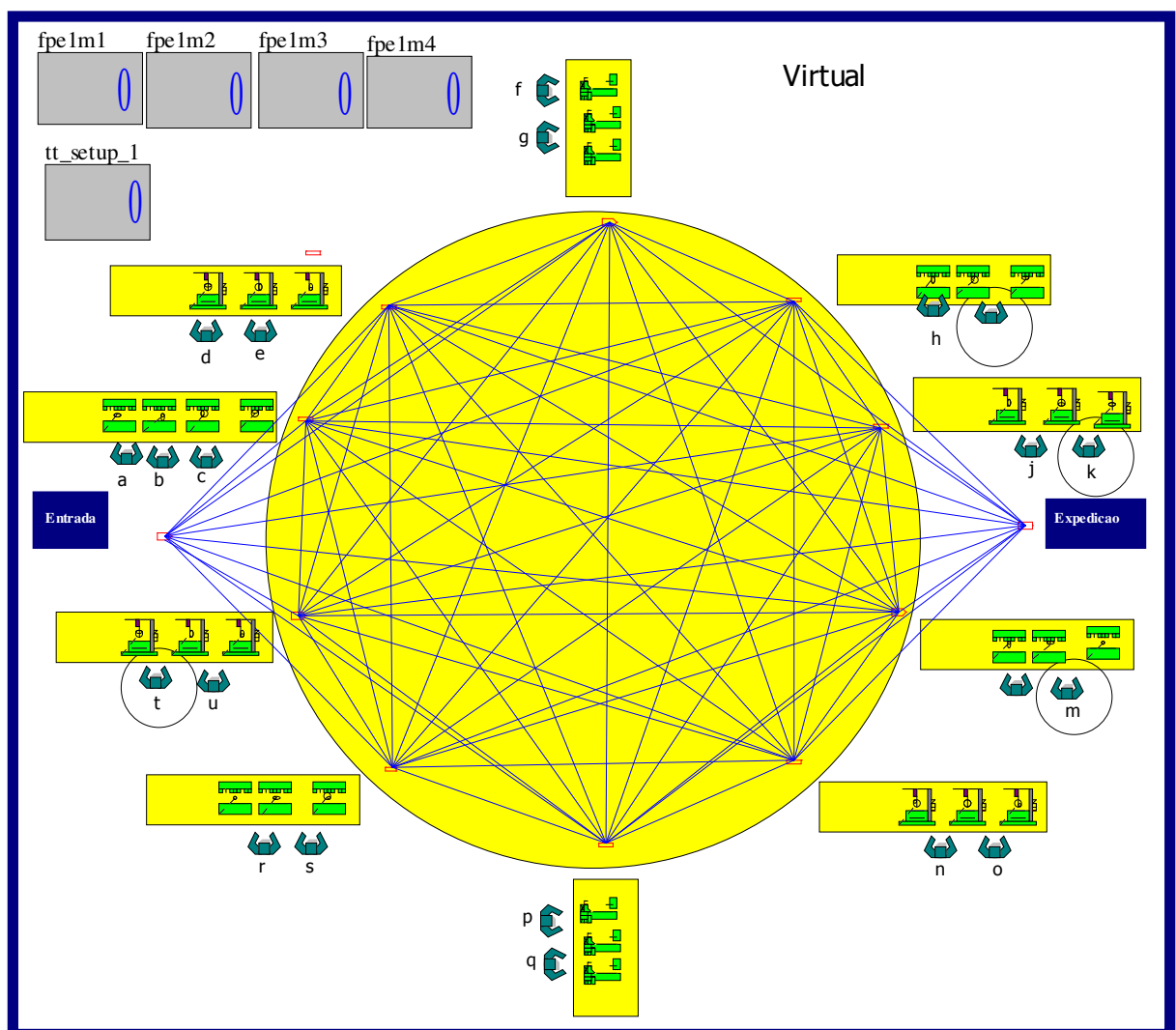


Figura 5.27 A animação da configuração celular Virtual

### 5.4.2 Lógica de um departamento

A figura 5.28 mostra o trecho do modelo correspondente à lógica de um departamento na configuração celular virtual. No exemplo sendo considerado, este departamento corresponde ao número 1, que é o único com quatro máquinas e três operadores. Este trecho do modelo se repete dez vezes, uma para cada departamento. Para uma melhor compreensão da lógica do departamento, este trecho foi subdividido em três blocos: “Entrada”, “Operando com máquina já configurada” e “Operando com máquina não configurada”, respectivamente representados na figura pelos números 1, 2 e 3. Esses blocos são descritos nos próximos subitens.

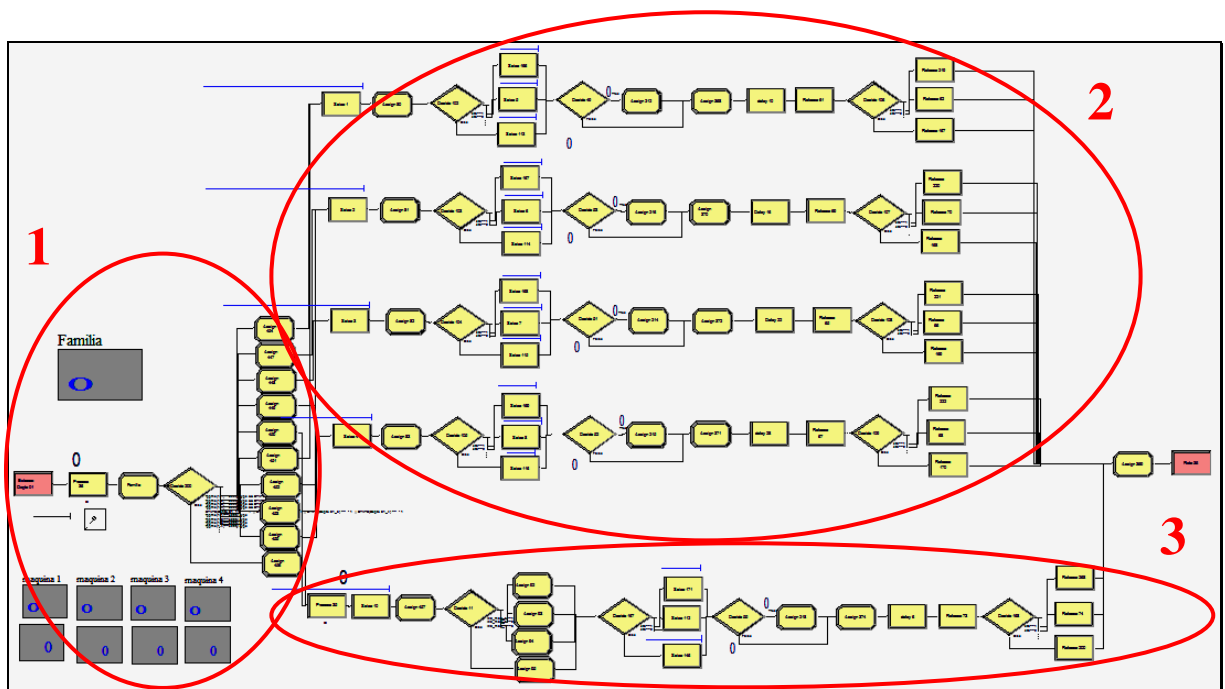


Figura 5.28 O conjunto de módulos que formam um departamento

#### 5.4.2.1 O bloco “Entrada” na lógica de um departamento.

A figura 5.29 mostra o bloco “Entrada” na lógica de um departamento, bloco este que recebe todas as entidades (lotes) que estão chegando para serem processadas nesse departamento. Ele atribui o valor da expressão “ $\text{expo}(0.1 \cdot q)$ ” ao tempo de processamento dessa entidade no departamento. Atribui o tempo de *setup* da entidade para este departamento com o valor da expressão “ $\text{reductor\_setup} \cdot \text{EXPO}(3)$ ”, que neste caso é um tempo de setup reduzido.



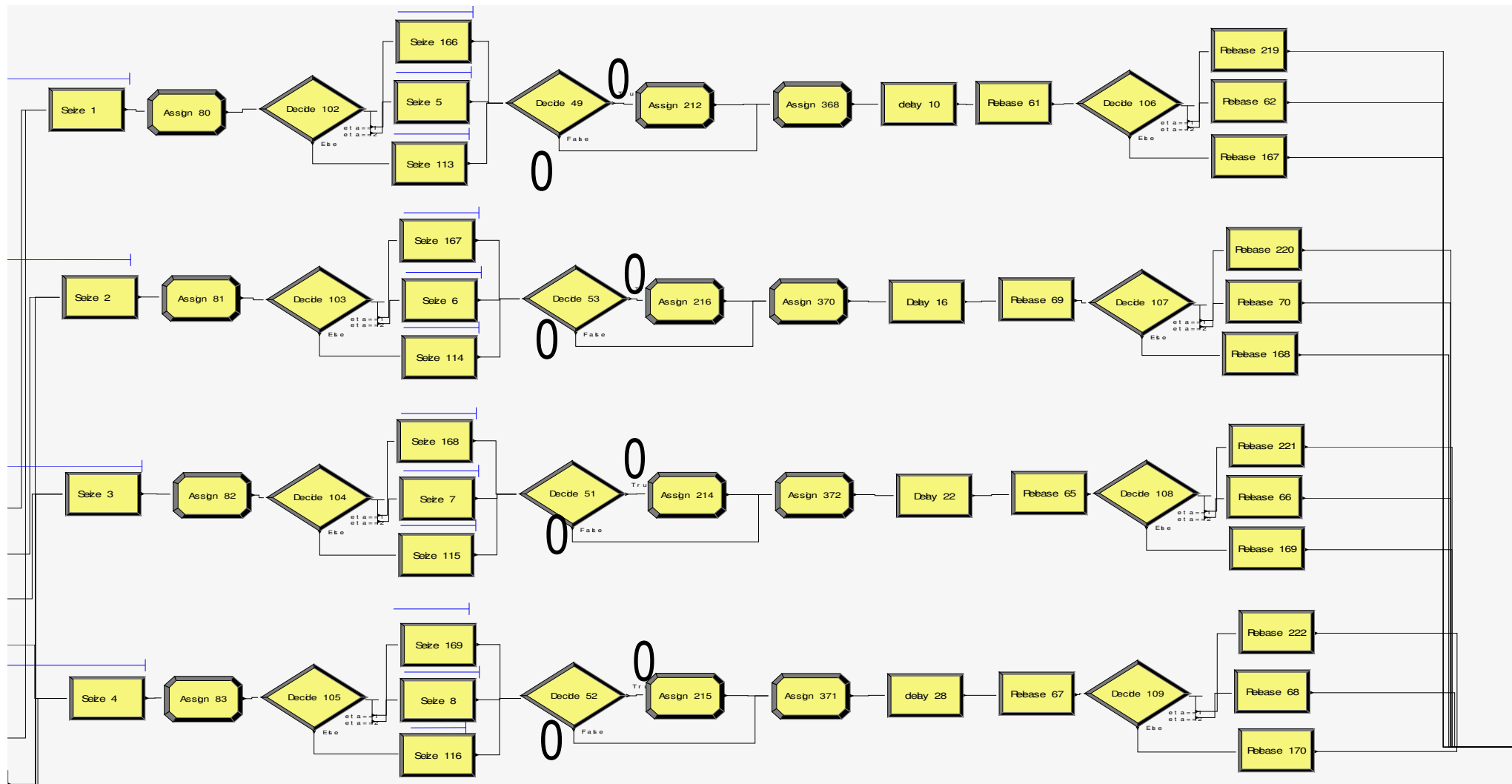


Figura 5.30 O bloco “Operando com máquina já configurada”

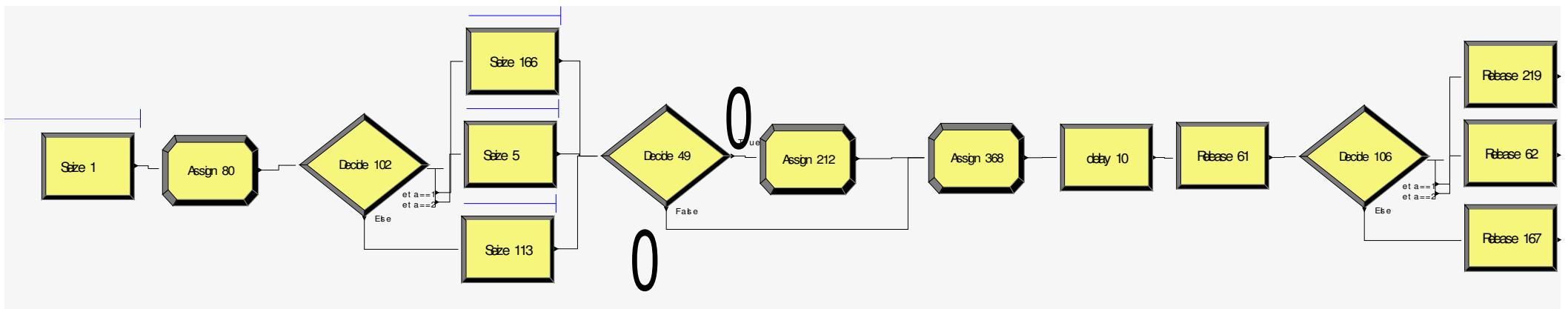


Figura 5.31 Trecho da máquina “Depto 01\_1” no módulo “Operando com máquina já configurada”



Na figura 5.30 podem-se visualizar quatro conjuntos distintos de módulos, cada um deles correspondendo a uma máquina distinta. No caso do departamento 1, tais módulos correspondem às máquinas “Depto 01\_1”, “Depto 01\_2”, “Depto 01\_3” e “Depto 01\_4”.

A figura 5.31 destaca o trecho correspondente apenas à máquina “Depto 01\_1”. Nesse trecho, a entidade (o lote) tenta ocupar a máquina “Depto 01\_1”, com prioridade alta. Caso a máquina esteja ocupada, a entidade espera na fila dessa máquina. As entidades são ordenadas nesta fila em função o menor tempo de processamento. Ainda nesse trecho, uma vez obtida a máquina, a entidade tenta obter com prioridade alta algum dos operadores do conjunto de operadores que operam essa máquina. Esse conjunto de operadores será maior ou menor (nula, dupla ou tripla) em função do grau de flexibilidade dos operadores escolhida em cada rodada de simulação. Enquanto a entidade não obtiver o recurso, ela espera uma fila ordenada em função do menor tempo de processamento. Uma vez obtido o operador, caso ele seja externo (de um departamento distinto da máquina) é atribuído um tempo de movimentação ao operador com o valor dado pela função “UNIF(0.5,1)”. Caso contrário, o operador é interno, e nesse caso, o tempo de movimentação é nulo.

A seguir, esses tempos de movimentação, de setup e de processamento, são simulados. E na seqüência o operador e a máquina são liberados e a entidade segue para a máquina, que irá realizar a próxima operação, ou até a saída da célula, caso todas as operações previstas já tenham sido realizadas, com tempo de rota dado pela expressão “UNIF(30, 60)/60” minutos. O item A2.16 do Apêndice 2 detalha com profundidade este trecho do modelo celular virtual.

#### 5.4.2.3 O bloco “Operando com máquina não configurada”

A figura 5.32 mostra o bloco “Operando com máquina já configurada” na lógica de um departamento. Conforme a lógica descrita anteriormente, todas as entidades que entram neste bloco irão controlar uma máquina que esteja livre, mas que não esteja configurada para a família da entidade livre, ou então a primeira máquina que for liberada, e neste caso é pouco provável que tal máquina vá estar configurada. Neste bloco, a entidade que entra tenta ocupar uma das máquinas desse departamento, com prioridade média. Caso todas as máquinas estejam ocupadas, a entidade espera numa fila, que é ordenada em função do tempo de processamento. A seguir, atribui-se o valor da função expo(3) horas ao tempo de *setup*, considerando-o pleno (sem redução). Na seqüência, ele tenta obter com prioridade média algum dos operadores que operam máquinas desse departamento. Esse conjunto de operadores será maior ou menor (nula, dupla ou tripla) em função do grau de flexibilidade dos operadores escolhida em cada rodada de simulação.

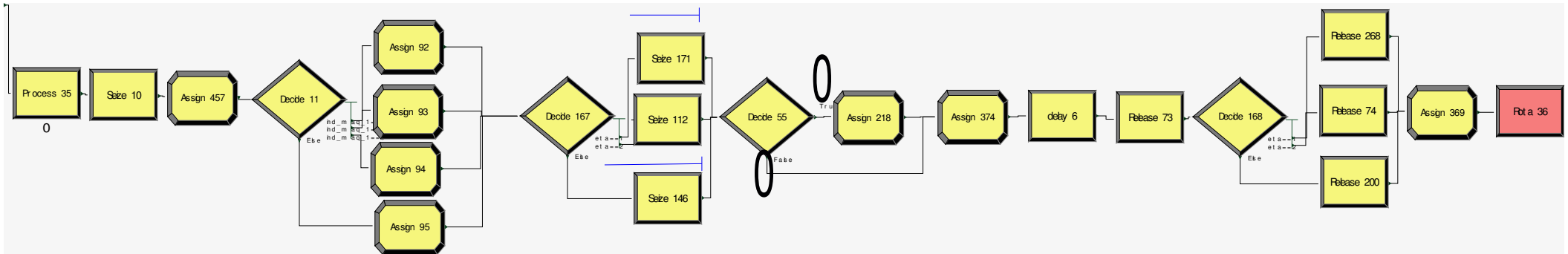


Figura 5.32 O bloco “Operando com máquina não configurada”

Enquanto a entidade não obtiver o operador, ela espera numa fila, que é ordenada pelo valor mais baixo de tempo de processamento. Uma vez obtido o operador, caso ele seja externo (de um departamento distinto da máquina) é atribuído um tempo de movimentação ao operador com o valor dado pela função “UNIF(0.5,1)”. Caso contrário, o operador é interno, e nesse caso o tempo de movimentação é nulo.

A seguir, esses tempos de movimentação, de setup e de processamento, são simulados. Na seqüência o operador e a máquina são liberados, e a entidade segue para a máquina, que irá realizar a próxima operação, ou até a saída da célula, caso todas as operações previstas já tenham sido realizadas, com tempo de rota dado pela expressão “UNIF(30, 60)/60” minutos. O item A2.17 do Apêndice 2 detalha com profundidade este trecho do modelo celular virtual.

#### 5.4.3 O trecho “Saídas”

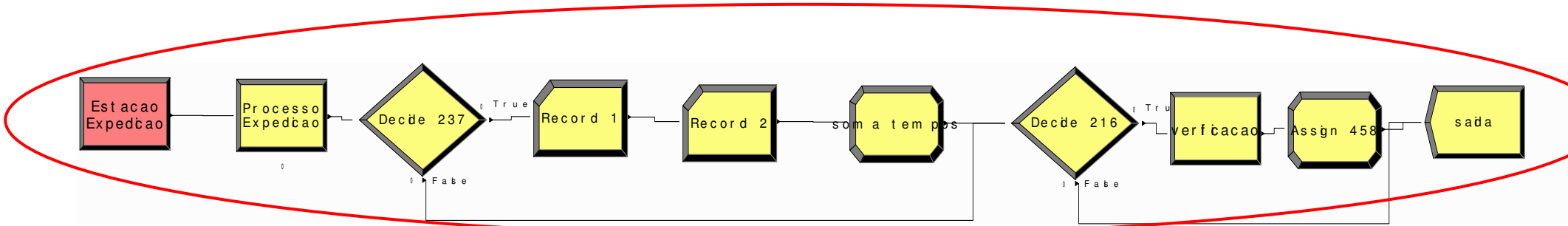
A figura 5.33 mostra o trecho “Saídas” no modelo da configuração celular virtual. Há três blocos neste trecho do modelo: “Saídas de entidades”, “Saída de dados” e “*Warm up*”. Estes blocos são representados respectivamente pelos números 1, 2 e 3 na figura 5.33. O segundo e o terceiro blocos estão descritos nos subitens 5.2.4.1 e 5.2.4.2, enquanto o primeiro foi descrito no subitem 5.2.4.3. Entretanto, como há alterações em alguns módulos do primeiro bloco, bem como a presença de alguns módulos diferentes, tais módulos são descritos no item A2.18 do Apêndice 2.

### 5.5 Conclusões

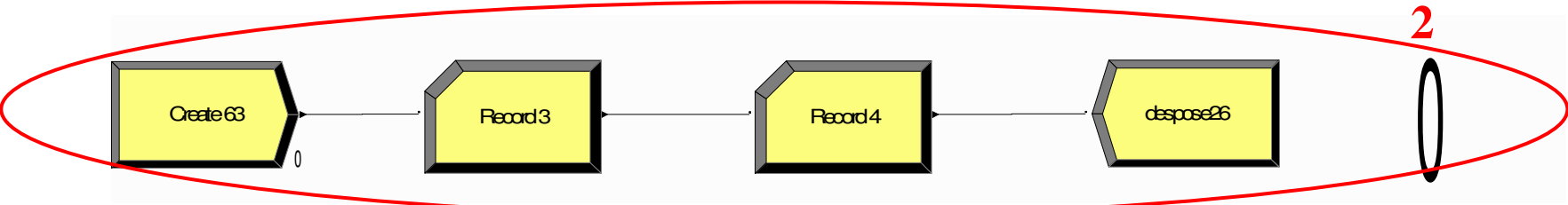
Este capítulo descreveu os modelos físicos desenvolvidos na presente tese. Foram descritas as modelagens realizadas com o software ARENA dos três tipos de configuração de sistemas manufatura contemplados neste trabalho, ou seja, o que emprega a configuração *job shop*, o que emprega a configuração celular tradicional e o último que utiliza uma configuração celular virtual. Primeiramente foram descritas características comuns aos três modelos, depois foram descritas as particularidades de cada uma das configurações. Desta forma, pôde-se verificar que as características do modelo conceitual estavam presentes no modelo físico.

O próximo capítulo aborda os métodos de verificação e validação empregados nos modelos da tese para certificar a idoneidade dos dados obtidos e a conformidade dos mesmos com o desempenho e os resultados de outros modelos empregados para representar um padrão semelhante ao adotado.

1



2



3

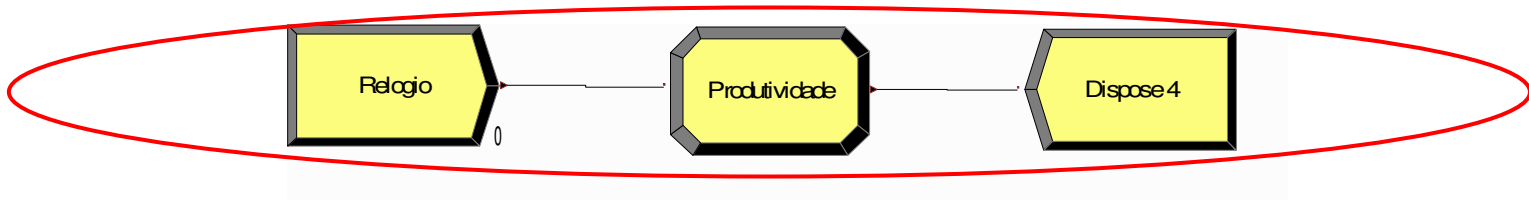


Figura 5.33 Blocos associados às saídas do sistema

## **CAPÍTULO 6**

### **VERIFICAÇÃO E VALIDAÇÃO**

Este capítulo apresenta a verificação e validação dos modelos, cujos aspectos foram verificados para auferir a consistência dos mesmos. Entre outras técnicas, foram empregados na validação, o uso de modelos determinísticos, variações sobre os dados de entrada e o uso de rotinas de rastreamento. Apesar de o conceito de validação não se aplicar aqui com propriedade - por se tratarem de modelos teóricos - admite-se uma validação num sentido mais amplo, conforme a abordagem desenvolvida no item 6.2.

#### **6.1 Verificação dos modelos**

Nos próximos subitens, são descritas as verificações realizadas para confirmar a consistência dos modelos. Conforme abordado anteriormente, foram utilizadas técnicas de verificação indicadas por Freitas Filho (2001), em particular o emprego de valores constantes, de forma a que os modelos passem a ser determinísticos e assim se possa verificar os resultados obtidos. Além disso, realizaram-se também variações sobre os dados de entrada para avaliar o desempenho do modelo com diversos parâmetros. Por último, procedeu-se a uma rastreabilidade (*trace*) de vários de seus procedimentos, com o objetivo de verificar se a simulação dos modelos realizada com o emprego do software Arena estava de acordo com o padrão adotado.

Inicialmente, são abordadas as verificações referentes às características comuns a todos os modelos, tais como: geração dos lotes de peças, geração de demanda, roteiro de fabricação etc. Depois são abordadas as verificações relativas aos casos particulares dos três modelos. A quantidade de dados e resultados é de tal ordem que é inviável transcrever e comentar todos eles numa tese, por isso, neste capítulo destacam-se algumas verificações e resultados mais significativos.

##### **6.1.1 Verificação de características comuns a todos os modelos**

Neste subitem, estão as verificações de características comuns aos três modelos, tais como geração dos lotes de peças, geração de demanda, roteiro de fabricação etc.

###### **6.1.1.1 Verificação da Geração dos Lotes de Peças e da Demanda das Peças**

Neste subitem, descreve-se a verificação dos modelos com relação ao trecho referente à geração dos lotes de peças e da demanda das mesmas. Pretende-se verificar, desse modo, se os modelos geram - para cada uma das cinquenta peças pertencentes às cinco

famílias de peças - valores de demanda anual dentro dos limites superior e inferior estabelecidos no padrão adotado e que se encontram na tabela 4.1. E depois, é verificado se as peças produzidas tendem de fato a atender essas demandas geradas. Para essa verificação foram realizadas simulações para cinco situações: três para o modelo Funcional, respectivamente com lotes de 30, 50 e 80 peças e para os modelos celular e virtual, ambos com lotes de 50 peças. A simulação foi de 3.600 horas, aproximadamente 1 ano. Usou-se o nível mais alto de variação de demanda, isto é, plena, e também do intervalo entre chegadas, ou seja, exponencial. O item A3.1 do Apêndice 3 descreve em detalhes esta verificação. Pode-se constatar que os trechos referentes à geração dos lotes de peças e das suas demandas atendem as especificações do padrão adotado.

#### 6. 1.1.2 Verificação do roteiro de fabricação, dos tempos de movimentação, *setup* etc.

Foram feitas seis simulações para a verificação dos modelos com relação ao correto funcionamento do seqüenciamento dos lotes de peças nos departamentos ou nas máquinas das células especificadas. O objetivo foi verificar se não havia nenhum erro de lógica nos três modelos desenvolvidos na tese quanto a modelagem com o software Arena. Assim, com a técnica *trace*, verificou-se se as entidades percorriam a seqüência certa de máquinas e operações previstas no padrão adotado e definido na metodologia. Foram verificados também os tempos médios de movimentação dos operadores, *setup*, processamento e movimentação das peças (ou rota). Mais uma vez, também via *trace*, procurou-se apurar se a modelagem com auxílio do software, não apresentava erros lógico e os valores reais de tempo de processamento, de movimentação, de preparação das peças durante a simulação estavam estritamente dentro do modelo adotado. O item A3.2 do Apêndice 3 descreve em detalhes esta verificação. Pode-se verificar que o roteiro de fabricação, os tempos de movimentação, tempos de setup etc. atendem as especificações do padrão adotado.

#### 6.1.1.3 Verificação do tempo total médio de fabricação de um lote de peças

Com essa verificação procura-se uma confirmação de que não há erros na lógica nos modelos desenvolvidos, pois caso houvesse os resultados totais iriam conter resultados estranhos por conta de um erro não identificado. Com essa finalidade foram realizadas diversas simulações medindo os tempos totais médios dos lotes de peças (sempre com 50 unidades) para todos os tipos de peças nos três tipos de configuração: *job shop*, celular tradicional e virtual. Essas simulações foram feitas, sempre, com um só tipo de peça por vez. O tempo de simulação foi de 3600 h, os tempos de processo, *setup*, movimentação de peças (ou rota) foram não determinísticos. O mesmo com relação à geração de demanda e o intervalo entre chegada de peças. Os lotes das peças foram agrupados em função do

número de estágios necessários para o seu processo de fabricação, ou seja, 3, 4, 5, 6 ou 7 estágios, uma vez que é este fator que mais determina qual o tempo total de processamento. Dessa forma, procurou-se verificar se todas as tarefas específicas e seus respectivos tempos estavam presentes nos modelos simulados. O item A3.3 do Apêndice 3 descreve em detalhes esta verificação. Pode-se verificar que não há erros na lógica nos modelos desenvolvidos, uma vez que os tempos médios totais obtidos foram exatamente os previstos para atender as especificações do padrão adotado.

#### 6.1.1.4 Verificação da flexibilidade dos operadores

Conforme abordado na descrição do modelo conceitual os modelos desenvolvidos seguem o padrão de modelos sujeitos a uma dupla restrição de recursos (*DRC – Dual-Resource-Constrained*), de máquinas e operadores. Os modelos contemplam operadores sujeitos a três níveis de flexibilidade,  $\eta = 1, 2$  ou  $3$ , ou seja, com habilidade para operarem uma, duas ou três máquinas. Este subitem aborda a verificação realizada com relação a esta flexibilidade, que constitui um elemento essencial desse padrão *DRC*. Através da técnica *trace*, se procura verificar se os operadores utilizam a sua flexibilidade nos seus três níveis e respeitando as regras estabelecidas na descrição do modelo conceitual no capítulo 4. Com essa finalidade foram realizadas diversas simulações com o objetivo de verificar se nos 3 modelos (*job shop*, celular tradicional e virtual) a flexibilidade dos operadores (ou *cross training*) estava correta, ou seja, conforme o estabelecido no subitem 5.1.2 e na tabela 4.7. Nesse subitem e tabela são descritos como os operadores são agrupados em **sets** ou conjuntos de operadores capazes de operar determinadas máquinas e como esses **sets** são constituídos de forma diferente em função do nível de flexibilidade escolhido ( $\eta = 1, 2$  ou  $3$ ). Assim, por exemplo, para o modelo *job shop*, o **set** de operadores que pode operar as máquinas do departamento 1 para  $\eta = 1$  (ou seja, habilidade para operar apenas uma máquina), designado como “operador\_1”, é composto pelos operadores operador\_a, operador\_b e operador\_c. Para  $\eta = 2$ , o set apto a realizar a mesma tarefa denomina-se “eta\_2\_operador\_1” e é constituído pelos operadores operador\_a, operador\_b, operador\_c, operador\_i, operador\_k, operador\_m e operador\_t. Por último para  $\eta = 3$ , o set denomina-se “eta\_3\_operador\_1” e se constitui dos mesmos operadores de “eta\_2\_operador\_1”. Os dados das diversas simulações realizadas para esta verificação foram transferidos para a planilha Excel “verificação cross training.xls” para análise. O item A3.4 do Apêndice 3 descreve em detalhes esta verificação separadamente para os 3 modelos. Pode-se constatar que o padrão de flexibilidade dos operadores especificado no padrão adotado se verifica nos três modelos.

### 6.1.2 Verificação de características específicas do modelo *job shop*

O objetivo desse subitem é verificar se a modelagem da configuração *job shop*, em particular do trecho da lógica do departamento – e de seus blocos - conforme a figura 5.10 e o subitem 5.2.3, não apresentam erros lógicos. Trata-se, então, de verificar os blocos “Entrada”, “Sistema Livre”, “Obtida uma máquina obtém operador interno” e “Obtida uma máquina busca qualquer operador”. Ao fim desse subitem é comentada ainda a verificação do atributo “Priority” no modelo *job shop*.

Para a verificação do bloco “Entrada”, foi realizada uma simulação de 500 horas, com  $\eta=3$ , lote=50 peças, com uma única replicação. Foram transferidos para a planilha os dados referentes aos primeiros 99 lotes que saíram da simulação. Os dados coletados referem-se à situação dos operadores e das máquinas nos diversos departamentos, ou seja, quantidade de recursos disponíveis e de recursos ocupados. Dessa forma, pode-se verificar cada bloco de entrada de cada departamento, confirmando-se que os lotes que chegam são encaminhados para o trecho correto dentro da lógica do departamento. O subitem A3.5.1 do Apêndice 3 descreve em detalhes esta verificação.

Para a verificação do bloco “Sistema Livre”, foi realizada uma simulação de 500 horas, com  $\eta=3$ , lote=50 peças, e uma única replicação. Procurou-se constatar a ausência de fila, já que os lotes são encaminhados para este trecho, tanto com  $\eta=2$  como com  $\eta=3$ , somente quando há simultaneamente ao menos um operador e uma máquina livres. Foi possível verificar ausência de fila e tempo de espera nulo para todos os trechos de sistema livre em todos os departamentos. O subitem A3.5.2 do Apêndice 3 descreve em detalhes esta verificação.

Por outro lado, para a verificação dos blocos “Obtida uma máquina obtém operador interno” ( $\eta = 2$  e  $3$ ) e “Obtida uma máquina busca qualquer operador” ( $\eta = 2$  e  $3$ ), foram transferidos alguns dados durante a simulação para uma planilha Excel, e através da análise desses dados se constatou de fato, em função da disponibilidade de um operador interno, que a entidade (o lote) seguia para um bloco ou outro do modelo (para “obtem um operador interno” ou “busca qualquer operador”). Pode-se verificar também que tanto o trecho “obtida uma máquina obtém o operador interno” como o trecho “Obtida uma máquina busca qualquer operador” estão funcionando corretamente. O subitem A3.5.3 do Apêndice 3 descreve em detalhes esta verificação.

A última verificação realizada no modelo *job shop* é o atributo “Priority”. A finalidade desse atributo no modelo é garantir que, quando mais de um departamento disputam um mesmo operador, a preferência recaia sobre aquele departamento que tenha a maior fila, ou seja, LNQ – *Longest, Number of Jobs in Queue*. Foi possível, através de dados obtidos



durante a simulação, constatar que este atributo atende o padrão estabelecido no modelo adotado. O subitem A3.5.4 do Apêndice 3 descreve em detalhes esta verificação.

### 6.1.3 Verificação de características específicas do modelo celular virtual

O objetivo desse subitem é verificar se a modelagem da configuração celular virtual, em particular do trecho da lógica do departamento – e de seus blocos - conforme a figura 5.28 e o subitem 5.4.2, não apresentam erros lógicos. Trata-se então de verificar os blocos “Entrada”, “Operando com máquina já configurada” e “Operando com máquina não configurada”.

#### 6.1.3.1 Bloco “Entrada”:

A finalidade da verificação do bloco “Entrada” do trecho Departamento no modelo celular virtual é reproduzir a entrada dos lotes no departamento, permitir o controle visual da quantidade de lotes entrados, atribuir valores para alguns atributos e variáveis e, por último, realizar um teste múltiplo para decidir para qual bloco cada lote deve ser encaminhado. O teste deve constatar uma das 4 possíveis situações, a) se há uma máquina livre e ajustada para a família do lote testado; b) se há uma máquina livre não ajustada para a família do lote testado; c) se há uma máquina ocupada, ajustada para a família do lote testado e; d) se só há máquinas ocupadas e não ajustadas para a família do lote testado. Para as situações “a)” e “c)” os lotes devem ser encaminhados para o bloco “Operando com máquina já ajustada”; para as duas outras situações, os lotes devem ser encaminhados para o bloco “Operando com máquina não ajustada”.

Para verificar este bloco foram transferidos para uma tabela os dados de solicitação de máquina do departamento durante a simulação, e analisadas os diversos encaminhamentos para os lotes que chegavam ao departamento, concluindo-se que ocorrem segundo o padrão desejado. O item A3.6 do Apêndice 3 descreve em detalhes esta verificação.

#### 6.1.3.2 Bloco “Operando com máquina já ajustada”:

A finalidade do bloco “Operando com máquina já ajustada” é representar a obtenção de uma máquina (livre ou não) já ajustada (para a família do lote atual) de cada departamento, bem como a obtenção de um operador habilitado para esta máquina, o correspondente processamento e o encaminhamento do respectivo lote para a próxima operação. O teste do bloco “entrada” já havia verificado a existência de uma máquina (livre ou não) já ajustada e encaminhou o lote para este bloco para que efetivamente fosse ocupada.

As operações mais significativas que este bloco realiza e representa são: *i)* obter a máquina ajustada e, caso esteja ocupada, aguardar em uma fila; *ii)* registrar e armazenar a informação sobre qual das máquinas do departamento e o respectivo operador começaram a ser ocupados; *iii)* registrar e armazenar a informação do horário em que esta máquina começou a ser ocupada; *iv)* buscar um operador interno (preferencialmente) ou externo livre, em função do “eta” (1, 2 ou 3); *v)* ordenar a fila dos lotes esperando por um operador, segundo o tempo mais curto de processamento e pelo departamento com fila mais longa; *vi)* definir se há tempo de movimentação do operador em função de se é externo ou não; *vii)* reproduzir o tempo de processamento, incluindo movimentação do operador e tempo de *setup*, neste caso reduzido; *viii)* liberar máquina e operador; *ix)* encaminhar o lote para a próxima operação. Essas propriedades já foram verificadas no subitem anterior e também na abordagem sobre os níveis de flexibilidade de operador (*cross training*).

#### 6.1.3.3 Bloco “Operando com máquina não ajustada”:

A finalidade desse bloco é muito parecida com a do bloco anterior, distinguindo-se, basicamente, por procurar obter uma máquina não ajustada. O outro bloco, após identificar uma máquina (livre ou não) que estivesse ajustada para uma família, dirigia para a obtenção apenas dessa máquina, enquanto no trecho atual, já que não há máquina ajustada, busca obter qualquer uma. Assim, busca representar a obtenção de uma máquina (livre ou não) não ajustada (para a família do lote atual) de cada departamento, a obtenção de um operador habilitado para esta máquina, o correspondente processamento e o encaminhamento do respectivo lote para a próxima operação. O teste do bloco “entrada” já havia verificado a inexistência de uma máquina (livre ou não) já ajustada e encaminhou o lote para este bloco, para que ocupasse a máquina livre, se houvesse, ou ficasse esperando por uma que fosse liberada. Assim, este bloco é idêntico ao anterior, e, por isso, foram realizadas as mesmas verificações, utilizando as mesmas planilhas.

#### 6.1.4 Verificação de características específicas do modelo celular tradicional

O objetivo desse subitem é verificar se a modelagem da configuração celular tradicional, em particular do trecho da lógica de uma célula conforme a figura 5.21 e o subitem 5.3.4, não apresentam erros lógicos. Trata-se, então, de verificar a lógica da célula e o bloco “Operando sempre ocupado”

##### 6.1.4.1 A lógica de uma Célula tradicional

Cada célula é composta de 6 a 7 blocos (conjuntos de módulos) que representam e simulam as operações das 6 ou 7 máquinas de cada célula. A finalidade de cada bloco é

representar e simular a solicitação, posse, utilização e liberação de cada máquina e operador habilitado por parte dos lotes (entidades), que pertençam a uma das famílias de peças que são processadas nesta máquina e célula. E representar a seguir o encaminhamento desses lotes para a próxima máquina da mesma célula, conforme as suas específicas seqüências de operações.

Assim, as operações que representam a simulação são: *i)* receber os lotes a serem processados por uma máquina específica alocada em uma célula específica; *ii)* gerar e armazenar um valor para o tempo de processamento específico para cada lote que passe por essa máquina específica; *iii)* contabilizar e exibir durante a simulação a quantidade de lotes que entram neste bloco; *iv)* solicitar, obter e utilizar a máquina específica desse bloco ou aguardar numa fila até que a máquina esteja disponível. *v)* gerar e armazenar o valor da prioridade do lote atual na obtenção de um operador habilitado, registrar e armazenar a informação de qual máquina está sendo utilizada pelo lote atual e qual das máquinas da célula foi utilizada; *vi)* registrar e armazenar o horário em que esta máquina e o operador começaram a ser ocupados, *vii)* buscar o operador que opera esta máquina como primeira habilidade (preferencialmente) ou outro livre em função do valor de “eta” (2 ou 3). *viii)* ordenar a fila dos lotes, esperando por um operador segundo o tempo mais curto de processamento; *ix)* definir se há tempo de movimentação do operador em função de o mesmo ter esta habilidade como sua primeira habilidade (melhor, se é de outra máquina dentro da mesma célula); *x)* reproduzir o tempo de processamento, incluindo movimentação do operador e tempo de *setup*, neste caso sempre reduzido; *xi)* liberar máquina e operador. *xii)* encaminhar o lote para a próxima operação. Novamente, o procedimento de verificação consistiu da coleta e transferência de dados durante a simulação para uma planilha de dados, e a seguir foi sendo verificado com êxito todas as operações descritas acima. O subitem A3.7.1 do Apêndice 3 descreve em detalhes esta verificação.

#### 6.1.4.2 Verificação da Geração de Operador Sempre Ocupado

Os resultados gerados na verificação do *cross-training*, assim como do subitem anterior, permitem verificar que este artifício lógico está isento de erro lógico. Ou seja, o modelo deixa de tentar usar o primeiro operador do conjunto de operadores (que seria o primeiro operador segundo a lógica adotada no modelo), porque ele se encontra sempre ocupado. A finalidade é gerar tempo de movimentação interna quando uma máquina da célula não tem operador capacitado para operá-la como sua primeira habilidade. Uma verificação mais aprofundada da Geração de Operador Ocupado e da geração de tempo de movimentação para este caso encontra-se no subitem A3.7.2 do Apêndice 3

### 6.1.5 Verificação da necessidade de *warm-up*

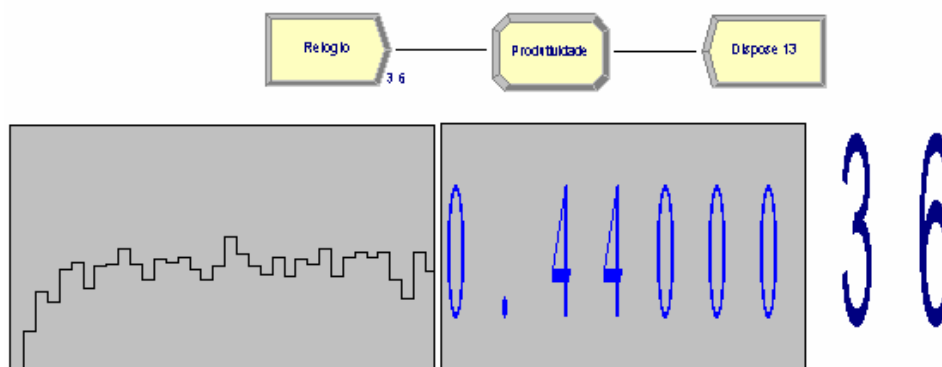


Figura 6.1 – Verificação da necessidade de *warm-up*

Um trecho denominado de “verificação da necessidade de *warm-up*” foi posteriormente acrescentado na simulação, conforme a figura 6.1. Ele indica o número médio de lotes de peças por hora, na unidade de tempo (em intervalos de 100 horas), na figura 0,44 lotes. Pode-se observar que o período transiente é relativamente curto, de forma que se verifica que o *warm-up* de 6 meses adotado no padrão (metade do tempo de simulação) é suficiente para expurgar este período. Uma descrição deste bloco encontra-se no item A3.8 do Apêndice 3.

## 6.2 Validação

O conceito de validação não se aplica aqui com total propriedade, uma vez que os modelos não estão reproduzindo um cenário real particular, mas conceitual, constituído de modelos teóricos adaptados ou reformulados a partir de pesquisas anteriores. Entretanto, admite-se uma validação num sentido mais amplo, comparando-se as características de desempenho dos modelos do novo padrão conceitual, a intensidade dos valores de resposta, o perfil gráfico dos resultados com os do padrão original. Avaliar a semelhança do padrão adotado na tese com o cenário conceitual apresentado por Suresh e Slomp (2005), é o que efetivamente se realiza a seguir, com a ajuda de gráficos.

Para facilitar a comparação e ampliar o escopo da avaliação, são apresentadas também as tabelas com os dados que deram origem a esses gráficos. Foi empregado o mesmo padrão utilizado por aqueles autores na tabela que denominam como “Mean flow ratio (COMP-LNQ-SPT)”. Nesta tabela, encontram-se os resultados da variável resposta tempo de fluxo, sob a forma de razão de fluxo médio. Esta razão é o quociente entre o tempo médio que os lotes gastam para serem totalmente processados e o tempo de 60,549 horas, que é também o tempo médio gasto, encontrado por aqueles autores, para que lotes de 50 peças no sistema *job shop* com “eta” igual a um, ao percorrem todos os processos. Além disso, corresponde também ao caso “COMP” nos critérios de prioridade LNQ e SPC,

que são também empregados nesta tese, conforme abordado no capítulo anterior. São comparados os valores e gráficos daqueles autores com alguns dos resultados e gráficos desta tese.

Para efeito dessa comparação, optou-se por empregar os modelos da tese com demanda constante, com intervalos entre chegadas também constantes, variando-se, entretanto, os níveis de flexibilidade dos operadores, os níveis de redução de *setup*, e os tamanhos de lote. Procurou-se introduzir um mínimo de variações, uma vez que já havia diversas diferenças entre o padrão da tese e o desses autores, além de pontos obscuros, confusos e mesmo ausentes no padrão original, que tornaram a replicação inviável. Entre outros, conforme abordado anteriormente, diferenças entre os fatores e os níveis adotados, ausência de dados sobre o sequenciamento de produção que afeta diretamente a carga do sistema e o tempo de fluxo. E, ainda, falta de clareza no cálculo dos intervalos entre chegadas de lotes de peças, valores ausentes em tabelas, algumas especificações de filas pouco claras e descrições ambíguas de algumas das regras de prioridade. Através dessas comparações realizadas, e apresentadas no item A3.9 do Apêndice 3, pode-se verificar a semelhança do padrão adotado na tese com o cenário conceitual apresentado por Suresh e Slomp (2005), ficando assim validados os modelos desta tese, segundo o padrão de validação adotado.

### **6.3 Conclusões**

Este capítulo apresentou a verificação e validação dos modelos empregados nesta tese. Para atestar a consistência dos modelos da tese, foram verificados aspectos comuns aos três modelos, como, por exemplo, a geração dos lotes de peças e aspectos particulares de cada um deles. Verificou-se, ainda, a necessidade do *warm-up*. Por outro lado, apesar de o conceito de validação não se aplicar aqui com propriedade - por se tratar de modelos teóricos - realizou-se uma validação num sentido mais amplo, comparando alguns resultados com os dos autores de trabalhos anteriores, em que o padrão adotado nesta tese se apóia. Dessa forma, é possível afirmar que os modelos empregados foram verificados e validados e seus resultados estão respaldados cientificamente.

## CAPÍTULO 7

### ANÁLISE DOS RESULTADOS

Este capítulo apresenta e analisa os resultados obtidos via simulação e projeto de experimentos dos três modelos desenvolvidos nesta tese, ou seja, configuração *job shop*, celular tradicional e uma nova concepção de celular virtual. Inicialmente, faz-se uma síntese dos resultados obtidos pelos autores do padrão em que se apóia esta pesquisa e, a seguir, passa-se a apresentar e avaliar os resultados obtidos para cada uma das variáveis resposta utilizadas.

#### 7.1 Análise dos resultados

Ao iniciar este capítulo, pareceu muito pertinente fazer uma síntese dos procedimentos e resultados obtidos por Suresh e Slomp (2005), uma vez que são os autores do padrão em que se apóia esta pesquisa. Ao analisarem e discutirem os resultados obtidos, concluíram que os fatores eram significativos com relação ao tempo de fluxo no sistema nos três modelos. A seguir, através de comparações visuais entre os gráficos e através dos valores das variáveis resposta, avaliaram qual dos sistemas era melhor (*outperform*). Foram bastante pontuais, não tirando conclusões sobre tendências. Por exemplo, afirmaram apenas que até tal tamanho lote tal sistema era superior. Fizeram uma especial comparação entre gráficos “Tempo de Fluxo x Tamanho de Lote” para os dois níveis de redução de *setup* e para os vários níveis de flexibilidade de operador. Compararam também as filosofias IDLE e COMP, comentadas anteriormente no item 3.1, e concluíram que, apesar de ser estatisticamente significativa a diferença de desempenho entre elas, em termos de valor as diferenças eram pequenas e marginais. Por último, com relação aos níveis de flexibilidade dos operadores, o celular tradicional teve sempre desempenho superior, exceto no caso em que simultaneamente a redução de *setup* era igual a 40% e a flexibilidade era dupla ( $\eta = 2$ ), quando então o celular virtual teve melhor desempenho.

Entretanto, ao considerar o objetivo desse capítulo, a abordagem daqueles autores não pareceu responder adequadamente à pergunta sobre quais os fatores que acarretam um desempenho superior da configuração celular virtual. O fato de os fatores serem significativos com relação ao tempo de fluxo, não quer dizer que beneficiem mais a um dos sistemas do que aos outros. Assim, apesar do intervalo entre chegadas influenciar significativamente o tempo de fluxo dos lotes, pode ser que para qualquer intervalo de entrada o celular tradicional seja sempre superior aos demais sistemas. Ou seja, a

abordagem dos autores citados não responde ao questionamento da tese: “Em que situações o sistema celular virtual é superior, podendo gerar uma vantagem competitiva?”.

Por esse motivo, resolveu-se verificar se são significativas as variações de tempo de fluxo entre 2 sistemas, isto é, (tempo de fluxo do sistema A) - (tempo de fluxo do VCM) para níveis diferentes dos fatores analisados. E, de fato, pôde-se comprovar e verificar que um fator afeta significativamente a diferença de desempenho entre dois sistemas, o que permite concluir que, quanto maior o nível desse fator, maior a diferença de desempenho de um deles com relação ao outro. Por isso, neste capítulo, avalia-se a diferença de desempenho dos sistemas no projeto de experimentos.

Conforme descrito no subitem 4.2.5, o tratamento dos dados foi realizado com auxílio do software Minitab, consistindo de vinte análises de variância, cinco para cada uma das quatro variáveis respostas. Assim, nos próximos quatro itens são apresentados e analisados os resultados obtidos para cada uma dessas variáveis.

Foi utilizado um gráfico do Minitab para análise do relacionamento entre um par de variáveis. O gráfico “*Scatterplots*” (dispersão) com regressão era indicado para a representação gráfica dos dados obtidos, pois mostra a dispersão dos valores obtidos por simulação e seus valores médios. Os pontos plotados formam linhas descontínuas verticais. Utilizou-se também a plotagem de tendência por regressão no mesmo gráfico. Foi utilizada regressão linear nos gráficos que envolviam fatores com dois níveis, isso é para representar o fator redução de *setup* e a variação nos intervalos entre chegadas de lote de peças. Foi empregada a regressão quadrática para representar graficamente o fator variabilidade da demanda, isto porque seus experimentos têm três níveis. Por outro lado, a regressão cúbica foi empregada na representação do fator tamanho do lote, porque este fator tem quatro níveis no projeto de experimentos. Como a quantidade de níveis de um experimento costuma ser pequeno, optou-se pela regressão com “*fit intercept*”, na qual as curvas passam pelos pontos médios dos pontos plotados, para evitar extrapolações nas conclusões. Desta forma, se dispôs de um recurso muito útil na análise da influência dos fatores no desempenho das variáveis resposta.

Os resultados completos com as correspondentes tabelas dos dados das análises de variância e os gráficos “*Scatterplots*” encontram-se no Apêndice 4. As principais informações das vinte tabelas e vinte e dois gráficos estão contidas na tabela 7.1, e a partir desta tabela se tecem neste capítulo de forma sintetizada os comentários sobre estes resultados.

Tabela 7.1 Síntese dos resultados

Variável resposta	Diferença	$\eta$	Variabilidade da Demanda (0%, 25%, 100% Unif. Lim)	Intervalo entre Chegadas (const, expo)	Redução do Setup (40%, 75%)	Tamanho do Lote (30, 40, 50 e 60)
Tempo médio de fluxo	VCM-JS	1	-	3 ↑ v	2 ↑ v	1 ↓ v
		2	-	3 ↑ v	2 ↑ v	1 ↓ v
		3	-	2 ↑ v	3 ↑ v	1 ↓ v
	VCM-CM	2	4 ↑ v	2 ↑ v	1 ↓ v	3 ↑ v
		3	4 ↑ v	3 ↑ v	2 ↓	1 ↑
Tempo médio de filas	VCM-JS	1	-	3 ↑ v	2 ↑ v	1 ↓ v
		2	-	2 ↑ v	3 ↑ v	1 ↓ v
		3	-	2 ↑ v	3 ↑ v	1 ↓ v
	VCM-CM	2	4 ↑ v	2 ↑ v	1 ↓ v	3 ↑ v
		3	4 ↑ v	2 ↑ v	3 ↓ v	1 ↑ v
Ocupação dos Operadores	VCM-JS	1	-	-	1 ↑ v	2 ↓ v
		2	-	-	1 ↑ v	2 ↓ v
		3	-	-	1 ↑ v	2 ↓ v
	VCM-CM	2	3 ↑	-	2 ↓	1 ↑
		3	-	-	1 ↓	2 ↑
Produtividade	VCM-JS	1	-	-	-	1 ↓
		2	-	-	-	1 ↓
		3	-	-	-	1 ↓ v
	VCM-CM	2	-	-	1 ↓	2 ↓
		3	-	-	-	-

Pode-se observar que cada linha da tabela 7.1 resume um resultado da análise de variância do experimento, para uma determinada variável resposta. Assim, a primeira linha informa que a diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 1$  (operadores habilitados a operar apenas um tipo de máquina) é afetada significativamente (com nível de significância estatística de 99%) pelos fatores intervalo entre chegadas, redução do setup e tamanho do lote.

Ainda para esta primeira linha, pode-se perceber que os fatores que influenciam mais a diferença de desempenho no tempo de fluxo entre o celular virtual e o *job shop* são o tamanho do lote, seguido da redução de *setup* e logo a seguir do intervalo entre chegadas, nessa ordem. Os números 1, 2 e 3 dessa primeira linha da tabela indicam concretamente esta ordem de maior influência na variável resposta, isto é, diferença de tempo médio de fluxo entre os dois sistemas comparados (VCM e JS). Por outro lado, as setas para cima e para baixo indicam para cada fator se o aumento do nível (por exemplo, no caso de redução de setup de 40% para 75%) acarreta vantagem à configuração celular virtual (seta para cima) ou não (seta para baixo).



Finalmente, uma letra “v” foi acrescentada nesta tabela para indicar que a configuração celular virtual teve um desempenho superior em todos os níveis do fator correspondente. Assim, nessa primeira linha, o tempo médio de fluxo foi menor na configuração celular virtual do que no *job shop*, para os quatro tamanhos de lote (30, 40, 50 e 60 peças), para  $\eta = 1$ .

## 7.2 Resultados envolvendo o tempo médio de fluxo

Na diferença de desempenho no tempo de fluxo entre o celular virtual e o *job shop*, os dados da tabela 7.1 para  $\eta = 1$  e 2 indicam que os fatores que influenciam mais são o tamanho do lote, seguido da redução de *setup* e logo a seguir do intervalo entre chegadas, nessa ordem. Eles indicam também que a variabilidade de demanda não influencia significativamente. Por outro lado, no escopo analisado, a configuração celular virtual tem um desempenho superior ao *job shop* em qualquer dos níveis dos fatores considerados. Ou seja, o tempo de fluxo do celular virtual é sempre inferior ao do *job shop*. Por outro lado, pôde-se verificar que a vantagem das células virtuais aumenta na medida em que o tamanho de lotes é menor, maior é a redução de *setup* (em outras palavras, quanto menor o *setup*) e menor a regularidade das chegadas (ou seja, quanto mais irregulares forem esses intervalos). E para  $\eta = 3$  se chega a essas mesmas conclusões, com a diferença apenas que neste caso o intervalo entre chegadas influencia mais que redução de *setup*.

Conforme abordado anteriormente, dentro do padrão adotado, o sistema celular tradicional não é viável para  $\eta = 1$ , limitando-se então a comparação entre essa configuração e o celular virtual para os outros dois níveis de flexibilidade de operador. Assim, a partir dos dados da tabela 7.1, para  $\eta = 2$ , pode-se verificar que os quatro fatores influenciam significativamente na diferença de desempenho no tempo de fluxo entre o celular virtual e o celular tradicional. A ordem dessa influência é, em primeiro lugar, da redução de *setup*, seguido do intervalo entre chegadas, do tamanho do lote e, finalmente, da variabilidade de demanda. É possível verificar também que, no escopo analisado, a configuração celular virtual tem um desempenho superior ao celular tradicional em qualquer um dos níveis desses quatro fatores. Pode-se concluir também que o celular virtual tem sua vantagem aumentada quanto menor for a redução do *setup*, maior a variabilidade dos intervalos de chegadas (irregularidade na chegada de lotes), maior for o tamanho dos lotes e quanto maior for a variabilidade da demanda atual de peças (falta de previsibilidade). Esses dados mostram a superioridade das células virtuais com relação às células tradicionais no escopo analisado. Em outras palavras, o celular virtual é mais flexível que o tradicional e por isso menos sensível às limitações desse com relação à necessidade de uma demanda constante. E tem a vantagem da flexibilidade típica do *job shop*, a sinergia do *pool* de máquinas. Mas, por outro lado, não consegue tirar tanta vantagem da redução de *setup* como o celular tradicional.

Os resultados obtidos com relação à diferença de tempo médio de fluxo entre o celular virtual e o celular tradicional para  $\eta = 3$  são muito semelhantes aos obtidos para  $\eta = 2$ . Assim, aplicam-se aqui as afirmações anteriores, com algumas pequenas diferenças. Por exemplo, a intensidade da diferença no fluxo de tempo entre os dois sistemas é bem menor, ou seja, a vantagem do celular virtual é menos acentuada. A diferença média em torno de 20 horas com  $\eta = 2$  cai para cerca de 3 horas para  $\eta = 3$  (dados disponíveis no Apêndice 4). Por outro lado, a ordem de influência é, em primeiro lugar, do tamanho do lote, seguido da redução de *setup*, do intervalo entre chegadas e, finalmente, da variabilidade de demanda. Apesar de a vantagem do celular virtual permanecer dentro do escopo analisado, neste nível de flexibilidade, o celular tradicional chega a ser superior, ainda que com uma vantagem mínima, quando a redução do *setup* é muito alta e quando o tamanho de lote é muito pequeno (dados também disponíveis no Apêndice 4). Assim, pode-se concluir que a uma maior flexibilidade de operador gera vantagem menor para o celular virtual.

### 7.3 Resultados envolvendo o tempo médio de filas

Os resultados da diferença de tempo médio de filas entre o celular virtual e o *job shop* para  $\eta = 1, 2$  e 3 são praticamente os mesmos referentes à diferença do tempo médio de fluxo. Logo, existe uma forte correlação entre essas duas variáveis resposta. Assim, os comentários do item anterior, em geral, podem também ser aplicados para o presente item. A única diferença é, para  $\eta = 2$ , na ordem de significância dos fatores. No caso do tempo médio das filas, para este nível de flexibilidade do operador, o mais significativo é o tamanho do lote, seguido do intervalo entre chegadas e, logo a seguir, da redução de *setup*.

Os resultados da diferença de tempo médio de filas entre o celular virtual e o celular tradicional para  $\eta = 2$  e 3 também são praticamente os mesmos referentes à diferença do tempo médio de fluxo. Entretanto, para  $\eta = 2$ , os valores de tempo agora são maiores. No tempo médio de fluxo, a diferença de tempo era em torno de 20 h, enquanto agora é de 30 h (conforme apresentado no Apêndice 4). E para  $\eta = 3$ , o celular virtual leva sempre vantagem sobre o celular tradicional, e a diferença média de tempo de fila é em torno de 10 horas. Para este nível de flexibilidade de operador, há uma ordem diferente de significância dos fatores. O mais significativo é o tamanho do lote, seguido do intervalo entre chegadas e, logo a seguir da redução de *setup* e da variabilidade de demanda.

Como se poderia explicar a diferença de tempo médio de fluxo ser equivalente à diferença do tempo média de fila? Neste caso, quase todas as diferenças de tempo médio de fluxo são causadas pela diferença do tempo médio de fila entre as duas configurações.

De fato, o tempo médio de processo é o mesmo nas duas configurações, uma vez que as variações relativas ao *setup* e à movimentação são relativamente pequenas.

#### 7.4 Resultados envolvendo a ocupação dos operadores

Os dados da análise de variância da tabela 7.1 indicam que os fatores que influenciam na diferença de ocupação dos operadores entre o celular virtual e o *job shop* para  $\eta = 1, 2$  e  $3$  são apenas a redução de *setup* e o tamanho do lote, nessa ordem. Por outro lado, pode-se verificar que no escopo analisado o celular virtual tem uma menor ocupação dos operadores do que o *job shop* em qualquer dos níveis dos dois fatores que são significativos em cerca de 10%. Os valores da diferença VCM-JS são negativos (conforme o Apêndice 4), indicando de fato que o celular virtual realiza suas tarefas com menor ocupação dos seus operadores do que o *job shop*. No item 7.5, verifica-se que a produtividade do celular virtual é superior à dos demais sistemas no escopo avaliado. Assim, esta baixa ocupação é eficaz, pois produz mais com menor emprego de recursos. Por isso, está se considerando aqui a menor ocupação dos operadores como desempenho superior. Por outro lado, pode-se verificar que a vantagem das células virtuais aumenta na medida em que é maior a redução de *setup* e é menor o tamanho de lotes (neste último essa tendência é bastante suave).

Com relação à diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 2$ , pelos dados da tabela 7.1, pode-se comprovar que os fatores tamanho do lote, redução de *setup* e variabilidade de demanda, nessa ordem, geram diferenças significativas na variável resposta diferença de ocupação dos operadores entre esses dois sistemas. Já para  $\eta = 3$ , apenas o fator redução de *setup* e o fator tamanho do lote, nessa ordem, geram diferenças significativas. Dentro do escopo avaliado e para ambos os níveis de flexibilidade dos operadores, os resultados indicam um desempenho médio inferior do celular virtual sobre o celular tradicional em cerca de 10% para  $\eta = 2$  e cerca de 12,5%, para  $\eta = 3$ , isto é, o celular tradicional ocupa menos os operadores. Entretanto, como a produtividade do celular virtual é um pouco maior, como se pode verificar no item 7.5, este fato comprova que o celular tradicional se beneficia mais da redução de *setup*, requerendo menos tempo de operador. Mas ele não consegue tirar proveito desse fato por ter problemas de sequenciamento (*schedule*), justamente por ter restrições de capacidade, o que comprova o item 7.3 com relação ao tempo de fila. O fator tamanho do lote é responsável pela queda do desempenho do celular tradicional, na medida em que cresce o tamanho de lote. Por outro lado, verifica-se que quanto maior a redução do *setup*, maior a vantagem do celular tradicional. Finalmente, o aumento da variabilidade de demanda gera vantagem do

celular virtual sobre o celular tradicional, uma vez que a diferença de ocupação diminui, ainda que esta redução seja muito pequena (conforme Apêndice 4).

### 7.5 Resultados envolvendo a produtividade

Pelos dados da tabela 7.1, pode-se comprovar que apenas o fator tamanho do lote gera diferenças significativas na variável resposta diferença de produtividade entre a configuração celular virtual (VCM) e a *job shop* (JS). A produtividade do celular virtual é superior em número de peças produzidas por hora, em cerca de 400 peças por hora para  $\eta = 1$  e em cerca de 1.000 peças por hora para  $\eta = 2$  e 3 (conforme Apêndice 4). A análise do gráfico do único fator que é significativo permite concluir que este desempenho superior deve-se principalmente à diferença de produtividade, muito superior para o menor tamanho de lote dentro do escopo adotado, ou seja, lote de quarenta peças (conforme Apêndice 4). De fato, para lotes menores o tempo de *setup* impacta bastante a diferença de tempo total de fluxo (conforme o item 7.2), a diferença de tempo de fila (conforme o item 7.3), e a produtividade.

Com relação à diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 2$ , pelos dados da tabela 7.1, pode-se comprovar que apenas os fatores redução de *setup* e tamanho do lote geram diferenças significativas, nesta ordem. A produtividade do celular virtual é ligeiramente superior em número de peças produzidas por hora, nesse caso cerca de 150 peças por hora (conforme Apêndice 4). Apesar de a variação ser muito pequena, pode-se perceber pelos gráficos que quanto maiores forem os tamanhos de lote ou maior a redução de *setup*, menor ou nula é a vantagem do celular virtual sobre o tradicional (conforme Apêndice 4). Por outro lado, para  $\eta = 3$ , pode-se verificar que nenhum dos fatores afeta significativamente a variável resposta.

### 7.6 Conclusão

Este capítulo apresentou e analisou os resultados obtidos via simulação e projeto de experimentos dos três modelos desenvolvidos nesta tese, ou seja, configuração *job shop*, celular tradicional e uma nova concepção de celular virtual. Apresentou uma síntese dos resultados obtidos pelos autores do padrão em que se apóia esta pesquisa. A seguir, foram apresentados os resultados obtidos com relação às quatro variáveis resposta, ou seja, as diferença de tempo médio de fluxo, de tempo de fila, de ocupação dos operadores e de produtividade entre os três modelos avaliados, para os três níveis de flexibilidade dos operadores considerados nesta pesquisa.

## CAPÍTULO 8

### CONCLUSÕES

Este capítulo tem por objetivo demonstrar os resultados da pesquisa, bem como destacar a sua contribuição aos estudos acadêmicos, o seu caráter inédito e as suas conclusões.

O primeiro objetivo específico da tese era criar três modelos de simulação para representar três sistemas da manufatura: o primeiro com tipo de processo por lote com configuração *job shop*, outro com configuração celular tradicional e o último utilizando um sistema de manufatura celular virtual. Nesse trabalho os modelos foram implementados através do software de simulação ARENA. Estes três sistemas foram modelados e encontram-se descritos conceitualmente no capítulo quatro e empiricamente no capítulo cinco. A modelagem acabou sendo realizada com a versão 9.00 desse software de simulação.

Outro objetivo era empregar um sistema fabril que envolvesse diferentes tipos de processos, por exemplo, torneamento, furação e fresamento, que considerasse ao menos três tipos distintos de máquinas, relativos a três distintos processos. O padrão adotado acabou sendo muito mais complexo do que o previsto inicialmente. Cada um dos três modelos é composto por trinta e uma máquinas, de dez tipos diferentes de processos. Além disso, considera também outro tipo de recurso, que são os operadores, concretamente vinte e um operadores habilitados para operar um, dois ou três tipos diferentes de máquinas. Todos os modelos estão sujeitos à mesma carga (os mesmos pedidos, com os mesmos roteiros de fabricação e com as mesmas demandas). Trata-se de um padrão com modelos capazes de processar cinquenta distintos tipos de peças, que necessitam passar por variados processos.

Efetivamente, foi empregado um projeto de experimentos (DOE), conforme especificado nos objetivos, onde foram identificados e validados os fatores e níveis mais significativos, assim como as variáveis resposta que melhor avaliassem o desempenho do sistema. Foram empregados variados fatores. Concretamente, além do fator sistema (*job shop*, celular tradicional e celular virtual), foram empregados também os fatores flexibilidade dos operadores, tamanho do lote, variabilidade anual da demanda, redução no tempo de *setup* e tempos aleatórios de chegada de novos pedidos. Por outro lado, foram utilizadas, além da variável resposta especificada nos objetivos, ou seja, a métrica média do tempo de fluxo, as variáveis resposta tempo de fila, ocupação dos operadores e produtividade. Para a execução dos cálculos e geração de gráficos e tabelas utilizou-se o software estatístico MINITAB versão 14.

O último objetivo também foi atingido, ou seja, comparar os resultados das configurações propostos nas diversas situações abordadas acima. Essas comparações estão descritas no capítulo sete, que trata dos resultados obtidos via simulação e projeto de experimentos dos três modelos desenvolvidos, apresentando as correspondentes análises de variância, os gráficos e as análises da influência de cada fator no desempenho dos três sistemas avaliados. Por sua vez, o capítulo seis descreve as inúmeras verificações realizadas e a validação dos modelos para assegurar a idoneidade dos resultados. Ao fim desse mesmo capítulo, a validação do modelo, através de diversas constatações, concluiu que a nova concepção de célula virtual apresentada nesta pesquisa tem um desempenho superior às demais configurações, inclusive à celular virtual do padrão original em que a pesquisa se apoiou.

Acredita-se que a presente tese seja uma contribuição, uma vez que desenvolve uma pesquisa envolvendo significativa quantidade de fatores e níveis de variação até o presente não pesquisados, com um considerável e inédito conjunto de variáveis resposta, através da técnica de projeto de experimento. A forma de comparar, por diferença de desempenho estatisticamente significativa, também foi inédita para este tipo de aplicação. E, finalmente, o emprego de uma nova concepção de célula virtual, que privilegia a flexibilidade (garante a sinergia do *pool* de máquinas) ampliando o escopo de estudo com relação aos trabalhos anteriores, abrindo um novo campo de pesquisa.

Deste trabalho, várias conclusões interessantes foram obtidas. A primeira é chave para esta pesquisa. O fato de, no escopo adotado, o celular virtual ter tido um desempenho superior aos demais sistemas para praticamente todos os níveis de todos os fatores empregados, provando que existem cenários industriais em que o emprego das células virtuais proporciona uma vantagem competitiva com relação aos demais sistemas.

Outra conclusão importante é constatar que ter trabalhado com uma gama maior de variáveis resposta do que o padrão original em que esta pesquisa se apoiou, resultou em um grande benefício. O padrão original emprega, na prática, apenas a variável resposta taxa de tempo de fluxo. Esta variável é muito importante para comparar e avaliar o desempenho dos sistemas. Entretanto, ela tem em si embutidos e misturados o tempo de processo, o de movimentação, o de *setup* e o de espera. O valor do primeiro, em média, é bem previsível; o segundo é, em sua maior parte, também bastante previsível. Os valores dos dois seguintes são consequência e medida da flexibilidade da sinergia do *pool* de máquinas, típica do *job shop*, e da economia de movimentação e de *setup*, típica da tecnologia de grupo. Por isso, o uso da variável resposta tempo de fila foi essencial para avaliar quanto de fila e de *setup* estão embutidos na variável tempo de fluxo. Assim, pôde-se verificar que a maior parte da diferença de desempenho do tempo de fluxo era devido à diferença dos tempos de fila. Ou

seja, por falta de flexibilidade e surgimento de gargalos, que seguram o sequenciamento de produção (*schedule*) na fila, e por isso o celular virtual gera muito menos fila que o celular tradicional.

Pôde-se verificar também, tanto com relação à variável tempo de fluxo como com relação ao tempo de fila, que o celular virtual é muito mais robusto que os demais sistemas. Ou seja, é tanto melhor o seu desempenho quanto maior for a variabilidade dos intervalos de chegada dos lotes de peças. Cenário muito comum nas empresas, sujeitas a freqüentes mudanças de programação de pedidos.

Talvez a conclusão mais importante seja a comprovação de que o sistema celular virtual tem o seu melhor desempenho e diferencial num cenário intermediário entre os outros dois. Assim o sistema celular virtual, diante do celular tradicional, tem melhor desempenho para lotes menores e pior para níveis maiores de redução de *setup*, e exatamente o oposto diante do *job shop*. Este fato abre uma enorme perspectiva de aplicação das células virtuais. Podem ser usadas quando haja dúvidas de qual dos dois sistemas extremos é melhor utilizar, porque obteria as vantagens de ambos e tem as desvantagens sempre em grau menor do que os extremos. Em especial, a migração do *job shop* para o celular virtual não incorre em elevados custos nem em rupturas de fornecimento na cadeia de suprimento. A partir dessa nova configuração, poder-se-ia avaliar de forma concreta e prática os benefícios e desvantagens da mudança e, ainda, a possibilidade de migrar para a célula tradicional. É como uma “simulação real”!

A fila sempre representa uma restrição ou uma má utilização da capacidade de produção e, dentro do escopo da pesquisa, como o sistema celular virtual foi superior em todos os níveis frente aos dois sistemas neste aspecto, comprova-se que este cenário tira melhor partido dos recursos disponíveis. Por ter parte da flexibilidade do *job shop* (a sinergia do pool de máquinas) e parte do benefício da tecnologia de grupos (tirando proveito da redução de *setup* através da produção de famílias de peças), a manufatura celular virtual constitui um cenário muito interessante para a produção em lote, uma vez que, muitas vezes, é muito difícil ter e manter um cenário estritamente *job shop*, abrindo mão de se beneficiar em parte com as reduções de *setup* que se obtêm ao privilegiar as famílias de peças. Igualmente, é também muito difícil ter um cenário estritamente celular tradicional, quando uma maior flexibilidade de recursos geraria ganhos ante as naturais oscilações de demanda e o ciclo de vida, cada vez mais curto, dos produtos.

Entretanto, o nível mais baixo de flexibilidade dos operadores confere ou aumenta a vantagem do celular virtual com relação ao celular tradicional, sendo, então, outro cenário interessante para a escolha da aplicação da célula virtual.

Por outro lado, a menor ocupação dos operadores do sistema celular tradicional com relação ao celular virtual, pelo fato de o primeiro se beneficiar mais da redução de *setup*, pode induzir erroneamente a pensar que teria, por esse motivo, um desempenho superior. A associação e análise dos fatores, diferença de ocupação dos operadores, diferença de produtividade e diferença de tempo de fila, permitiu tirar uma conclusão diferente. Com frequência, ocorre esta menor ocupação de operadores, simultaneamente com uma comparativamente menor produção. A freqüente produtividade superior do celular virtual decorre da sua flexibilidade, enquanto o celular tradicional perde produtividade por baixa eficácia no seqüenciamento de produção, cujos lotes permanecem retidos em filas. Então, a mais baixa ocupação, antes, é certa ociosidade. Esse quadro acentua-se ainda mais quando aumenta a variabilidade de demanda, pois mais se desestabiliza o seqüenciamento de produção do sistema celular tradicional, aumentando a ociosidade e caindo ainda mais a produtividade, conferindo ao celular virtual um desempenho muito superior.

Por último, a análise da variável resposta diferença de produtividade indicou que o sistema celular virtual apresenta desempenho especialmente superior para tamanhos menores de lotes de peças. Apesar da considerável diferença de desempenho, este resultado teoricamente era esperado com relação ao *job shop*, em virtude do peso do *setup* completo nos pequenos lotes. Entretanto, surpreendeu que o melhor desempenho do celular virtual com relação ao celular tradicional tenha sido também com o menor lote do escopo do estudo.

Para efeito de síntese das conclusões descritas neste capítulo, é listado a seguir um resumo dos resultados obtidos na presente pesquisa:

- Os três modelos de simulação foram implementados corretamente e conforme o especificado.
- O sistema fabril empregado envolveu diferentes tipos de processos.
- Empregou-se a técnica de projeto de experimentos conforme especificado.
- Foi realizada a comparação dos resultados das configurações propostas.
- Dentro do escopo do estudo, a configuração celular virtual teve um desempenho superior aos demais sistemas para praticamente todos os níveis de todos os fatores.
- O emprego de uma gama maior de variáveis resposta resultou em um benefício significativo para a análise dos resultados.
- A configuração celular virtual tem mais flexibilidade que o celular tradicional, gerando menos filas.



- A configuração celular virtual é muito mais robusto que as demais configurações.
- O sistema celular virtual tem o seu melhor desempenho e diferencial num cenário intermediário entre as outras duas configurações.
- O sistema proposto utiliza melhor a capacidade.
- Para cenários com mais baixo nível de flexibilidade dos operadores o celular virtual amplia mais a sua vantagem competitiva.
- Em função da sua maior flexibilidade, a configuração proposta está menos sujeita à ociosidade forçada de recursos devido a filas e gargalos.
- O sistema celular virtual apresenta desempenho superior especialmente com lotes menores.

Fica claro, ao fim dessas considerações, que existem diversos cenários industriais em que, comprovadamente, o sistema celular virtual, em particular a nova concepção desenvolvida nesta tese, apresenta um desempenho expressivamente superior em comparação com os sistemas tradicionais, constituindo, assim, um significativo potencial de vantagem competitiva.

Para futuros trabalhos poderia ser muito interessante estender esta pesquisa para toda a cadeia de suprimentos, permitindo uma visão mais holística da influência das diversas configurações de manufatura na cadeia produtiva. Por outro lado, poderia ser interessante também ampliar o problema, considerando na configuração *job shop* que os equipamentos de cada departamento, ainda que sendo do mesmo processo, tivessem características diferentes, introduzindo mais um fator de sinergia freqüente nesse tipo de configuração. Dessa forma, nas demais configurações, essa nova consideração traria também novas possibilidades e restrições, ampliando e tornando mais realista a abordagem. Outra possibilidade interessante de estudo futuro seria considerar a duplicação de máquinas em células tradicionais, fato que é corriqueiro nas indústrias. Assim, o modelo de célula tradicional tiraria partido desta possibilidade de aumento de flexibilidade.

Poder-se-ia considerar também uma abordagem distinta. Nos últimos anos, o modelo celular tradicional tem evoluído para uma célula com menos máquinas, porém mais versáteis e flexíveis, do tipo CNC (de Controle Numérico Computadorizado). Como o número de empresas que estão adotando essa solução é cada vez mais significativo, para futuras abordagens esse fato tem de ser levado em conta. De forma semelhante, isto tem aumentado a flexibilidade, em decorrência da presença cada vez maior de dispositivos e aparelhos especiais para dar suporte à atividade do operador em células de manufatura. Isto

pode gerar um aumento das habilidades e competências dos operadores nas células. E por isso esse fato é relevante para a continuidade da pesquisa.

Outra variação interessante que se poderia fazer no experimento seria o emprego da distribuição normal em algumas das variáveis aleatórias utilizadas como fatores de controle. Caberia, por exemplo, empregar esta função no fator redução de setup, que deixaria de ser calculado por uma função exponencial. Ou então, o fator variação nos intervalos entre chegadas de lote de peças, deixaria de ser gerado por uma exponencial, ou ainda, o fator variabilidade da demanda anual ( $d_i$ ) de cada peça, deixaria de ser dado por uma função uniforme. A função uniforme é muito indicada para descrever processos aleatórios, em especial na área de fabricação.

Outro tema interessante para futuros trabalhos é o emprego de uma gama maior de regras de prioridade, considerando-as como níveis distintos de um dos fatores do próprio experimento, uma vez que influenciam decisivamente nos resultados.

Por outro lado, além das regras de prioridade, seria muito interessante acrescentar ainda mais alguma parte de um sistema administrativo de produção. Assim, o emprego de mais elementos de um MRP II (Planejamento dos Recursos de Manufatura – *Manufacturing Resource Planning*), tais como o módulo MRP (Planejamento das Necessidades de Materiais – *Material Requirements Planning*) ou o MPS (Plano Mestre de Produção - *Master Production Scheduling*), faria que os modelos retratassem melhor a realidade dos sistemas de manufatura.

## REFERÊNCIAS BIBLIOGRÁFICAS

- BOSE, S.J., **An Introduction to Queueing Systems**, Kluwer/Plenum Publishers, 2002.
- BERRY, M.J.A. AND LINOFF, G., **Data Mining Techniques for Marketing, Sales, and Customer Support**. 2. ed. New York, NY: John Wiley and Sons, 1997. 454 p.
- CARSON, J. S., Introduction to modeling and simulation. **Proceedings of the 2005 Winter Simulation Conference**, p. 22.
- CERVO, Amado L.; BERVIAN, Pedro A. **Metodologia Científica**. 5.ed. São Paulo: Prencite Hall, 2002.
- CHIAVENATO, Idalberto **Teoria geral da administração**. 6.ed. Rio de Janeiro: Ed. Campos, 2001.
- CURRAL, J., *Statistics Packages: A General Overview*. **Maths&Stats**. November, 1994. Disponível em : <[http://www.ltsn.gla.ac.uk/articles/94\\_11/overview\\_packages.html](http://www.ltsn.gla.ac.uk/articles/94_11/overview_packages.html)>. Acesso em: 25 outubro 2005.
- DROLET, J. R., **Scheduling virtual cellular manufacturing systems**, 1989. Dissertação, Purdue University, Indiana.
- DROLET, J. R.; ABDULNOUR, G.; RHEAULT, M., The Cellular Manufacturing Evolution. **Computers & Industrial Engineering**, UK, outubro 1996. Vol 31, No. 1/2, pp. 139 – 142.
- ELMARAGHY, H. PATEL, V. & ABDALLAH, I. B., Scheduling of Manufacturing Systems under Dual-Resource Constraints Using Genetic Algorithms. **Journal of Manufacturing Systems**, 2000. Vol 19, No. 3, pp. 186 – 201.
- FERREIRA, J. C. E., **Apostila da Disciplina Sistemas Integrado de Manufatura (EMC5255)** Universidade Federal de Santa Catarina. 2000
- FREITAS FILHO, P. J., **Introdução à Modelagem e Simulação de Sistemas: com Aplicações em Arena**. Florianópolis: Visual Books Editora, 2001. 322 p.
- GIL, A. C., **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 1991.
- HOLLAND, J. H., **Adaptation in Natural and Artificial Systems**. Cambridge, Massachusetts: MIT Press, 1975.

JENSEN, J. B., The Impact of Resource Flexibility and Staffing Decisions on Cellular and Departmental Shop Performance. **European Journal of Operational Research** 2000. No.127, p. 279-296.

KANNAN, V. R., Efficiency vs. Flexibility: A Comparison of Group Technology and Virtual Cellular Manufacturing Systems, **Southeastern Chapter of the Institute for Operations Research and the Management Sciences Conference**, Myrtle Beach, SC, October 1997a

KANNAN, V. R., A Simulation Analysis of the Impact of Family Configuration on Virtual Cellular Manufacturing. **Production Planning & Control**, 1997b, vol. 8, No. 1, p. 14-24.

KANNAN, V. R., Analyzing the Trade-off Between Efficiency and Flexibility in Cellular Manufacturing Systems. **Production Planning & Control**, 1998, vol. 9, No. 6, p. 572- 579.

KANNAN, V. R.; GHOSH, S., Cellular Manufacturing Using Virtual Cells. **International Journal of Operations & Production Management**, vol. 16, No.. 5, 1996, pp. 99-112.

KO, K.; EGBELU, P. J., Virtual Cell Formation. **International Journal of Production Research**, 2003, vol. 41, No. 11, p. 2365–2389.

LAW, A. M., How to build valid and credible simulation models. **Proceedings of the 2008 Winter Simulation Conference**, p. 40-41.

LAW, A. M.; MCCOMAS, M. G., Simulation of Manufacturing Systems. **Proceedings of the 1997 Winter Simulation Conference**, p. 86-89.

MAK, K. L.; WANG, X. X., Production Scheduling and Cell Formation for Virtual Cellular Manufacturing Systems. **International Journal of Advanced Manufacturing Technology**. Londres: 2002. vol. 20, p. 144–152.

MARIA, A. Introduction to Modeling and Simulation. **Proceedings of the 1997 Winter Simulation Conference**, p. 7-13.

MARTINS, P. G.; LAUGENI, F. P., **Administração da Produção**. São Paulo: Ed. Saraiva, 2002. 445 p.

MCLEAN, C. R., BLOOM, H. M., HOPP, T. H., "The Virtual Manufacturing Cell". **Proceedings of Fourth IFAC/IFIP Conference on Information Control Problems in Manufacturing Technology**, Gaithersburg, MD, October 1982.

MORRIS, J. S. AND TERSINE, R. J., A Simulation Analysis of Factors Influencing the Attractiveness of Group Technology Cellular Layouts, **Management Science**, Vol. 36, No. 12, December 1990, p. 1567–1578.

MONTGOMERY, D. C., **Design and Analysis of Experiment**. 4. ed. New York: J. Wiley. 1997. 704 p.

PEGDEN, C. D.; SHANNON, R. E.; SADOWSKI, R. P., **Introduction to Simulation Using Siman**. 2. ed. McGraw-Hill Companies. Março 1995.

SÁNCHEZ, P. J., Fundamentals of simulation modeling. **Proceedings of the 2007 Winter Simulation Conference**, p. 55-57.

SANCHES, S. M., Better Than a Petaflop: The Power of Efficient Experimental Design. **Proceedings of the 2008 Winter Simulation Conference**, p. 73-84.

SARKER, B.; LI, Z., Job Routing and Operations Scheduling: a Network-Based Virtual Cell Formation Approach. **Journal of the Operational Research Society**. 2001. No. 52, p. 673-681

SILVA, Edna L. de; MENEZES, Estera Muszkat. **Metodologia da pesquisa e elaboração de dissertação**. 4.ed. rev. at. Florianópolis: Laboratório de Ensino à Distância da UFSC, 2005.

SLACK, N.; CHAMBERS, S.; JOHNSTON R., **Administração da Produção**. 2. ed.. São Paulo. Editora Atlas. 2002

SURESH, N. C.; SLOMP, J., Performance Comparison of Virtual Cellular Manufacturing with Functional and Cellular Layouts in DRC settings. **International Journal of Production Research**. 1 March 2005. Vol. 43, No. 5, 1 March 2005, p. 945–979.

TUBINO, D. F., **Manual de Planejamento e Controle da Produção**. São Paulo: Atlas. 1997

VAKHARIA, A.; MOILY, J. P.; HUANG, Y., Evaluating Virtual Cells and Multistage Flow Shops: an Analytical Approach. **The International Journal of Flexible Manufacturing Systems**. Boston: 1999. No. 11, p. 291–314.

WANG, Y. H.; YIN, C. W.; ZHANG, Y., A multi-agent and Distributed Ruler Based Approach to Production Scheduling of Agile Manufacturing Systems. . **International Journal of Computer Integrated Manufacturing**, 2003, Vol. 16, No. 2, p. 81-92.

WHITLEY, D., A Genetic Algorithm Tutorial. **Statistics and Computing**.1994. No. 4, p.:65-85.

## GLOSSÁRIO

**Células de manufatura** - configuração física de manufatura que tem como característica a presença de recursos transformadores variados, isto é, com processos distintos, voltadas a operar sobre famílias de peças.

**Ciclo de vida de um produto** - Tempo de vida de um produto, que inclui os processos de transformação do produto desde a extração ou produção da matéria prima até o descarte final após o consumo.

**Crítérios de prioridade** - Conjuntos predefinidos de regras podem ser utilizados para estabelecer a prioridade em que as tarefas e pedidos pendentes devem ser executados.

**Cross training** - flexibilidade de um operador, medida pelo número de tipos distintos de máquinas que ele pode operar (de um a três).

**Famílias de peças** - peças que têm características semelhantes de processamento.

**Flow shop** - leiaute por produto.

**Leiaute celular tradicional** - constituído basicamente por células de manufatura.

**Leiaute de manufatura** - Distribuição física de elementos fabris (estações de trabalho, máquinas, equipamentos etc) em um determinado espaço físico.

**Leiaute job shop, funcional ou por processo** - leiaute onde processos similares são localizados juntos um do outro.

**Manufatura celular virtual** - configuração lógica de um sistema manufatura em que as máquinas, embora agrupadas de forma semelhante ao job shop, operam virtualmente como células.

**Mix de produção** - combinação das quantidades e dos tipos de produtos a serem fabricados.

**Modelo de configuração** - especificação e representação utilizada em uma simulação de uma configuração individual e específica de produção, seja celular, celular virtual ou *job shop*.

**Padrão de referência** - características e definições estabelecidas pelos autores de trabalhos de pesquisa que comparam diferentes modelos de configuração de produção.

**Projeto de experimentos** - consiste de uma seqüência de experimentos em que são feitas alterações simultâneas em diversos fatores – segundo determinados níveis, tais como, alto, médio e baixo - de modo que se possa avaliar estatisticamente os possíveis efeitos dessas alterações nas variáveis resposta envolvidas no processo.

**Roteiro de fabricação** - seqüência de processos a que um produto (peças e matérias-primas) deve se submeter, estabelecido pela engenharia de fabricação.

**Seqüenciamento da Produção** (*Scheduling*) - lida com a alocação de recursos a operações. É um processo de tomada de decisão visando otimizar um ou mais objetivos, como minimizar número de pedidos atrasados, tempo de processamento dos pedidos, entre outros.

**Setup** – conjunto de todas as tarefas necessárias, desde o momento em que se tenha completado a última peça do lote anterior até o momento em que, dentro do coeficiente normal de produtividade e qualidade, é produzida a primeira peça do lote posterior.

**Simulação de um sistema** – é a execução de um modelo do sistema, entendido como uma representação da construção e funcionamento desse sistema.

**Tempo de fluxo** - tempo total de escoamento da produção de uma peça nos seus diversos processos de fabricação.

**Throughput** - tempo total para produzir toda a ordem de produção.

**Tipos de processos** - classificação dos sistemas produtivos com relação ao volume e continuidade das peças produzidas.



## APÊNDICES

## APÊNDICE 1

### OBTENÇÃO DA CONFIGURAÇÃO DOS ROTEIROS DE PRODUÇÃO

Conforme apontado anteriormente, Suresh e Slomp (2005) não descreveram de maneira clara a metodologia empregada para a definição dos roteiros de fabricação das peças, nem especificaram os roteiros das peças. Segundo eles, o que causou isto foi a necessidade de serem concisos. Entretanto, essa informação é essencial ao se descrever um experimento e ao se estabelecer um padrão a ser reproduzido no modelo de simulação. Por esse motivo, neste apêndice se descreve a metodologia empregada nesta tese para a obtenção da configuração dos cinquenta roteiros de fabricação correspondentes a cada tipo de peça.

Apesar de Suresh e Slomp (2005) não explicitarem essas informações, eles estabeleceram algumas restrições e especificações a respeito das peças, que se encontram na tabela 4.1, e que são reproduzidas na tabela A1.1 para facilitar a visualização.

Tabela A1.1 – Tipos de peças

	<b>Famílias de Peças</b>				
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Nº de tipos de peças	12	10	8	8	12
Nº de operações	4 – 6	3 – 6	4 - 6	4 – 7	4 - 6
Tipos de máquinas	1, 2, 4, 5, 6, 8	2, 3, 7, 8, 9, 10	1, 3, 5, 6, 8, 10	1, 2, 4, 5, 7, 9, 10	1, 3, 4, 6, 7, 9
Demanda anual	248 - 2.232	310 – 2.790	395 – 3.555	395 – 3.555	248 – 2.232

Fonte: adaptado de Suresh e Slomp (2005)

Os roteiros contidos na tabela A1.2 foram criados aleatoriamente, entretanto obedecendo os seguintes requisitos:

1. Ser coerente com as restrições da tabela A1.1
2. As cargas decorrentes dos roteiros devem estar razoavelmente balanceadas, ou seja, um departamento não deve ficar demasiadamente sobrecarregado e outro muito ocioso, com uma oscilação média em torno de 5%.
3. O número médio de operações dos tipos de peças de uma família deve ser aproximadamente igual à média dos valores limites do intervalo do número de operações. Assim, por exemplo, o número médio de operações da família 1 deve ser próximo de cinco, uma vez que os limites do intervalo são quatro e seis. Pretende-se com este procedimento proporcionar que a carga do sistema não fique nem muito baixa nem muito alta.

A tabela A1.2 sintetiza os roteiros de cada peça adotados nesta tese, onde o recurso de sombreamento foi utilizado para facilitar a identificação das peças e suas famílias. Assim, por exemplo, a peça do tipo P40 faz parte da família 4 e a sua seqüência de fabricação é o processamento sucessivo pelos departamentos 1, 2, 4, 5, 7, 9 e 10.

Tabela A1.2 – Os roteiros de fabricação das 50 peças

Família 1												Família 2								Família				
P 10	P 11	P 12	P 13	P 14	P 15	P 16	P 17	P 18	P 19	P1A	P1B	P 20	P 21	P 22	P 23	P 24	P 25	P 26	P 27	P 28	P 29	P 30	P 31	P 32
1	2	1	1	1	4	6	1	6	1	1	8	2	2	10	2	8	7	8	2	8	3	1	1	5
2	4	2	4	4	5	8	2	8	2	2	1	3	7	2	3	9	8	9	9	9	7	3	3	6
4	5	4	6	5	6	1	5	1	4	4	2	7	8	3	7	10	9	2	10	10	8	5	5	8
5	6	5	8	6	8	2	6	2	5	6	4	8	9	7		2	10	3		2	9	6	8	10
	8	6		8	1		8	4		8	5	9	10			3	2			3		8	10	
		8			2			5			6	10				7						10		

3			Família 4							Família 5														
P 33	P 34	P 35	P 36	P 37	P 40	P 41	P 42	P 43	P 44	P 45	P 46	P 47	P 50	P 51	P 52	P 53	P 54	P 55	P 56	P 57	P 58	P 59	P 5A	P 5B
5	6	1	8	1	1	1	1	4	10	1	10	1	1	1	9	1	9	9	7	1	1	1	7	3
6	8	3	10	3	2	2	4	5	9	2	1	2	3	3	4	3	7	7	9	3	3	3	6	4
8	10	6	1	5	4	4	7	7	7	4	2	4	4	6	3	4	6	6	1	4	4	4	4	7
10	1	8	3	6	5	7	9	9	5	5	5	5	6	7	1	6	3	4	3	6	6	6	3	9
1	3		5		7	9	10		4	7	7		7	9		7	1		4	7		7	1	
3			6		9	10			2	9			9			9			6			9		
					10				1															

Pode-se observar que esses roteiros obedecem ao primeiro item dos requisitos, uma vez que seguem as restrições da tabela A.1.1, tanto no número de peças por família, como no número de operações, como também nos tipos de máquinas.

A tabela A1.3 mostra o cálculo das cargas decorrentes dos roteiros de fabricação. Pode-se verificar que elas estão razoavelmente balanceadas, conforme estabelece o segundo item dos requisitos, ou seja, nenhum departamento está demasiadamente sobrecarregado ou ocioso. A oscilação média de fato é em torno de 5%. Para obter-se esta percentagem de 5,06%, foram utilizadas apenas as variações de nove dos dez departamentos, uma vez que o primeiro departamento tem uma capacidade superior aos demais tanto em máquinas como em operadores, conseqüentemente admitindo mais carga.

Por último, os roteiros gerados atendem também a especificação de que a média de operações dos tipos de peça de uma família deveria ser aproximadamente igual à média dos valores limite do intervalo do número de operações. De fato, a média de operações das famílias 1, 3 e 5 é cinco, da família 2 é quatro e meio e da família 4 é cinco e meio.

Tabela A1.3 – Balanceamento das Cargas

Departamentos	Tipos de peças da família que usam o departamento						% sobre o Total	Variação média
	Família 1	Família 2	Família 3	Família 4	Família 5	Totais		
1	11	0	7	7	10	35	14,23%	
2	10	8	0	6	0	24	9,76%	-2,37%
3	0	7	7	0	10	24	9,76%	-2,37%
4	10	0	0	6	9	25	10,16%	-6,64%
5	9	0	7	6	0	22	8,94%	6,16%
6	10	0	7	0	8	25	10,16%	-6,64%
7	0	7	0	7	10	24	9,76%	-2,37%
8	10	8	6	0	0	24	9,76%	-2,37%
9	0	8	0	6	9	23	9,35%	1,90%
10	0	7	7	6	0	20	8,13%	14,69%
<b>Totais</b>						246	100,00%	
<b>Média</b>							9,53%	5,06%

## APÊNDICE 2

### DETALHAMENTOS DA DESCRIÇÃO DOS MODELOS FÍSICOS

Este apêndice tem por finalidade detalhar alguns elementos descritos no capítulo 5 – Descrição dos Modelos Físicos. Ele servirá como subsídio para um aprofundamento maior na compreensão da modelagem realizada com auxílio do software de simulação Arena.

Sempre que possível, a descrição dos módulos é feita da esquerda para a direita, de cima para baixo. Ao descrever módulos, ou conjunto de módulos que aparecem repetidos nos modelos descritos, visando facilitar e tornar mais concisa a explicação, optou-se muitas vezes por descrevê-los como aparecem em uma célula, ou departamento, deixando para o leitor a generalização para todo o modelo. As funções, os parâmetros e os módulos do software Arena são representados neste texto sempre em negrito, enquanto os nomes desses módulos, das variáveis, dos atributos das entidades, dos recursos, dos conjuntos de recursos etc. foram escritos entre aspas.

#### A2.1 Geração das demandas de cada tipo de peça

A figura 5.5, capturada da tela do modelo desenvolvido no software Arena, mostra o trecho dos modelos responsável pela geração das demandas de cada tipo de peça. Os nomes dos módulos do software Arena, caso não sejam atribuídos a cada um deles, são gerados automaticamente. Assim, o nome “Decide 195” desta figura corresponde ao nome gerado automaticamente pelo software para identificar este específico módulo **Decide**. Os módulos desse trecho e suas configurações são:

- O módulo **create** “gerador de demanda” cria uma única entidade lógica no instante zero da simulação.
- O módulo **Decide** “Decide 195” testa se a variável “demand\_tipo” é igual a um ou dois, e em caso afirmativo envia a entidade lógica respectivamente para o módulo **assign** “demandas1” ou “demandas2”, caso contrário envia para “demandas3”
- O módulo **assign** “demandas1” atribui valores constantes para as 50 variáveis de demanda, cada uma armazenando os valores de demanda anual de cada um dos 50 tipos de peças que são processadas durante a simulação. Tais valores correspondem àqueles de variação nula de demanda que se encontram na segunda coluna da tabela 4.3. A figura A2.1 mostra a caixa de diálogo desse módulo.

- Os módulos **assign** “demandas2” e “demandas3” atribuem valores aleatórios para as 50 variáveis de demanda, segundo a semente geradora do software Arena. Nesses casos os valores de variação são respectivamente os que aparecem nas colunas ¼ e plena de demanda da tabela 4.3.
- O módulo **dispose** “Dispose 12” efetua o encerramento deste trecho do modelo, eliminando a entidade lógica criada anteriormente.

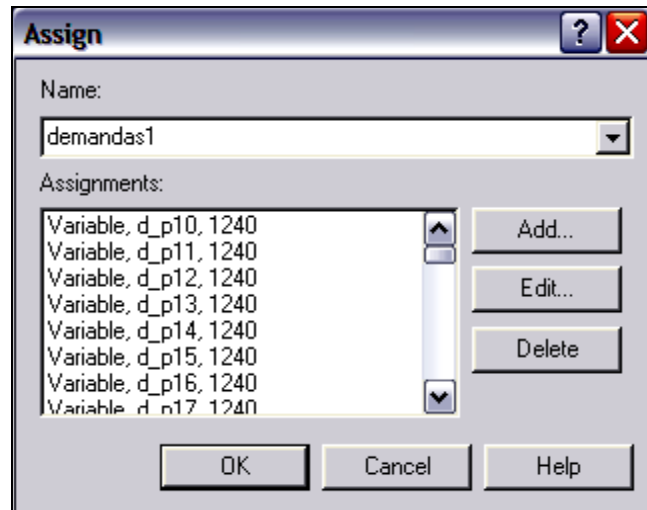


Figura A2.1 Caixa de diálogo do módulo **assign** “demandas1”

## A2.2 Geração dos lotes de peças e atribuição das seqüências de produção para cada peça

A figura 5.6 mostra o conjunto de módulos responsáveis pela geração das peças de uma das famílias de peças. Cada linha com três módulos é responsável pela criação de cada uma das peças dessa família. O caso apresentado na figura corresponde à geração de todos os tipos de peças da Família 1 que, conforme abordado anteriormente, tem doze tipos diferentes de peças. Como o padrão adotado é composto de cinco famílias de peças, há cinco conjuntos semelhantes a esse em cada um dos três modelos desenvolvidos nessa tese.

A figura 5.7 mostra o conjunto de módulos responsáveis pela geração dos lotes de um tipo de peça, sendo que neste exemplo específico a peça criada foi a “peca\_10” pertencente à família 1. Os módulos e suas configurações são:

- Módulo **create** “Peca\_10” – cria os lotes de um tipo específico de peça pertencente a uma das cinco famílias (neste exemplo as peças 10 pertencentes à

família 1), com intervalo entre chegadas dado pela expressão “EXPO((q/d\_p10)\*3600)\*e\_expo+((q/d\_p10)\*3600)\*e\_const” em horas, onde:

- “q” é a variável que contém o nível de tamanho do lote selecionado;
  - “d\_p10” é a variável que armazena a demanda gerada para um tipo de peça (no exemplo a peça 10);
  - “e\_expo” e “e\_const” são as variáveis associadas aos dois níveis do fator variabilidade dos intervalos entre chegadas de peças, assumindo alternadamente os valores 0 e 1 definem regimes constantes ou exponenciais desses intervalos;
  - 3600 são as horas de simulação (um ano aproximadamente).
  - O item “entidades por entrada” igual a 1 indica a entrada de uma entidade por vez, e no caso as entidades são os lotes de peças. Não há limite de quantidade de lotes a serem gerados exceto em função do tempo de execução de cada replicação que é de 3600 horas. A primeira geração ocorre às  $EXPO((q/1240)*3600)/2$  horas, onde “q” é o tamanho do lote, 1240 é a demanda anual média das peças da família 1, e para as outras famílias os valores médios são os mesmos da segunda coluna da tabela 4.3.
- Módulo **Station** “Entrada\_10” – define que as entidades estão em uma determinada estação, que no exemplo considerado é a estação “Entrada10\_St”.
  - Módulo **Assign** “Peca 10” – fornece ao atributo “tipo” (tipo de peça) o valor que identifica a peça (para o primeiro tipo de peça da família 1 atribuí o valor 10, ao segundo 11 etc.). Atribui também ao parâmetro “inicio” da entidade (que nesse caso é o lote) o valor **TNOW** (que uma função que retorna o tempo cronológico desde o início da simulação) correspondente a chegada desse lote no sistema. Atribui também à entidade uma figura para animação. E atribui ainda à entidade o roteiro de fabricação específico do seu tipo de peça, sendo que no exemplo considerado (peça 10), esta atribuição é feita com base no atributo “Entity.Sequence”, o qual contém um roteiro específico para cada um dos cinquenta tipos de peça, segundo a tabela 4.2, que no exemplo corresponde à seqüência “Seq\_10”. A figura A2.2 mostra o módulo **Sequence** do **template Advanced Transfer** com algumas das cinquenta seqüências de produção. Nesta mesma figura aparece uma janela que mostra a seqüência da peça 10 (o “Seq\_10”), a qual define que a entidade deve passar pelas estações “Depto 01\_st”, “Depto 02\_st”, “Depto 04\_st” e finalmente pela estação “Depto 05\_st”, nessa ordem. Os roteiros de cada **Sequence** seguem o padrão adotado, que neste caso encontra-se especificado na tabela 4.2.

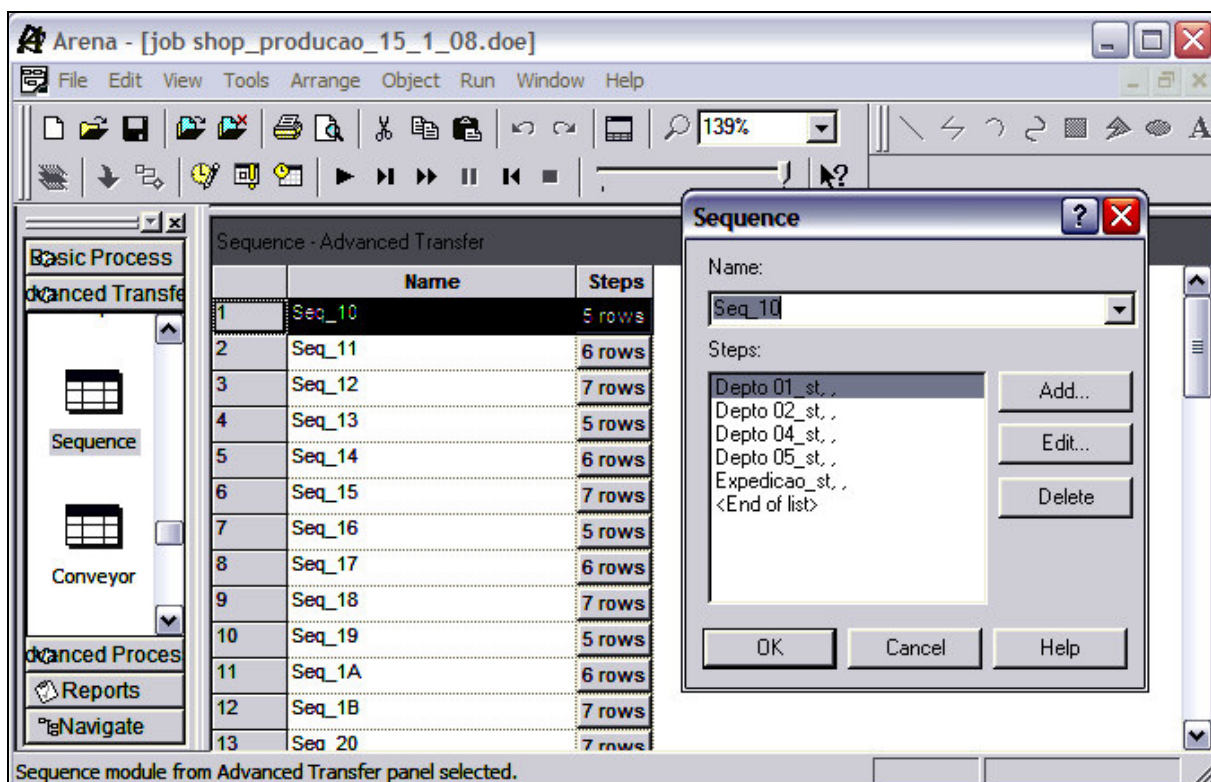


Figura A2.2 Detalhes do módulo **Sequence** em uma tela parcial do software Arena

- Módulo **Route** “Rota 1” – encaminha a entidade para o primeiro destino estabelecido no seu atributo “Entity.Sequence”, sendo que no exemplo considerado, este atributo é o Seq\_10, e neste atributo o primeiro destino é a estação “Depto 01\_st”, com tempo de rota igual a 10 minutos.

### A2.3 Descrição Detalhada do bloco “Entrada” na lógica de um departamento da Configuração *Job Shop*

A figura A2.3 mostra o trecho do modelo correspondente ao bloco “Entrada” na lógica de um departamento. Os módulos e suas configurações são:

- Módulo **Station** “Estação Depto 01” - define que as entidades estão em uma determinada estação, sendo que no exemplo este corresponde à estação “Depto 01\_st”.
- Módulo **Assign** “Assign 231” – atribui à entidade um atributo para marcar o tempo inicial de entrada no departamento. No exemplo este corresponde a “tempo\_ent\_1\_1” com valor **TNOW** (o tempo cronológico). A finalidade é verificar o modelo, estudando e avaliando os tempos no bloco “Sistema Livre”.



Também para efeito de verificação do modelo atribui-se à variável “t\_e\_1\_1” o valor de “tempo\_ent\_1\_1”.

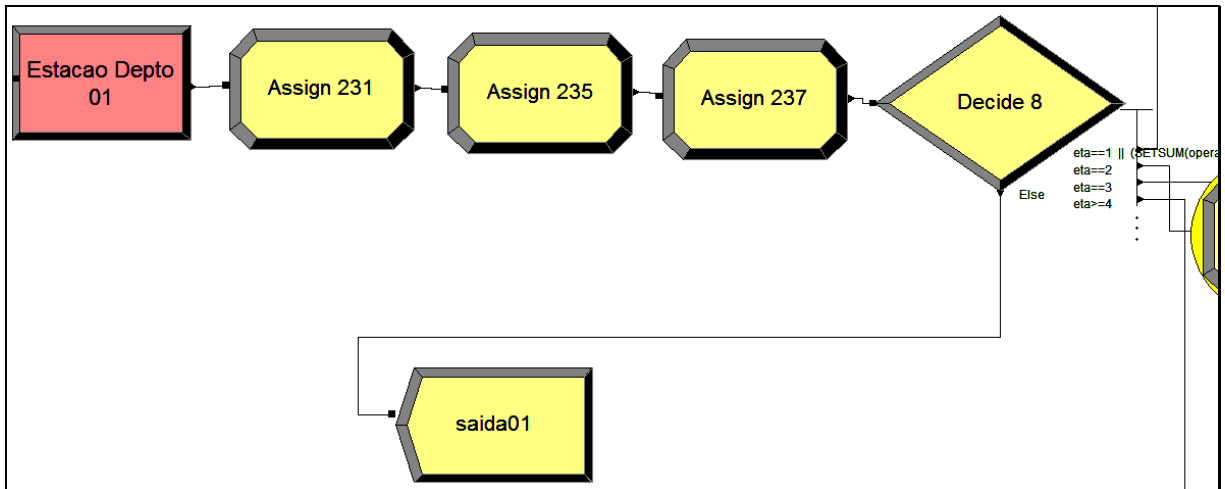


Figura A2.3 O bloco “Entrada” dentro da lógica de um departamento

- Módulo **Assign** “Assign 235” - atribui à entidade um atributo para marcar o tempo inicial de entrada no departamento, e no exemplo este atributo é “tempo\_ent\_1\_2”, cujo valor é **TNOW**. A finalidade consiste em verificar o modelo, estudando e avaliando os tempos no bloco “Espera de máquina com operador Livre”.
- Módulo **Assign** “Assign 237” - atribui à entidade um atributo para marcar o tempo inicial de entrada no departamento, e no exemplo este atributo é “tempo\_ent\_1\_3”, com valor **TNOW**. A finalidade é verificar o modelo, estudando e avaliando os tempos no bloco “Espera de máquina e de operador”. Este módulo atribui também os seguintes parâmetros:
  - “valor\_eta\_1” com valor de “eta”;
  - “oper\_ocup\_1” com o valor da função “**setsum**(operador\_1,5)” (**SETSUM(set de recursos)** retorna o número de recursos do **set** que estão ocupados);
  - “oper\_exist\_1” com o valor da função “**nummem**(operador\_1)” (a função **NUMMEM(set de recursos)** retorna o número de recursos que existem neste **set**);
  - “maq\_ocup\_1” com o valor de “**setsum**(Depto 01,5)”;
  - “maq\_exist\_1” com o valor de “**nummem**(Depto 01)”.
- Módulo **Decide** “Decide 8” – este módulo testa condições múltiplas, as quais são descritas a seguir:

1. A primeira é “eta==1 || (SETSUM(operador\_1, 5) < NUMMEM(operador\_1) && SETSUM(Depto 01, 5) < NUMMEM(Depto 01))”. Ou seja, caso “eta” seja igual a 1, isto é, flexibilidade nula, **OU** havendo operador **E** máquina disponíveis, encaminha-se a entidade (o lote de peças) para o bloco “Sistema Livre”. Esta segunda parte do “OU” corresponde à especificação do padrão adotado que estabelece que quando uma ordem chega a um departamento, se houver máquina e operador disponíveis, imediatamente se inicia o processamento.
2. Caso contrário, referente ao item 1, testa-se se “eta==2”, ou seja, se o *Cross training* ( $\eta$ ) for igual a 2 (isto é, flexibilidade dupla). Em caso afirmativo encaminha-se a entidade para o trecho em que se encontram os blocos “Espera de máquina com operador Livre” e “Espera de máquina e de operador”, referentes ao *Cross training* ( $\eta$ ) igual a 2, ou seja, o trecho que antecede imediatamente os blocos 3 e 4 da figura 5.10.
3. Caso contrário, no que se referem os itens 1 e 2 acima, faz-se o teste “eta==3”, ou seja, se o *Cross training* ( $\eta$ ) for igual a 3 (flexibilidade tripla). Em caso afirmativo encaminha-se a entidade para o trecho em que se encontram os blocos “Espera de máquina com operador Livre” e “Espera de máquina e de operador” referentes ao *Cross training* ( $\eta$ ) igual a 3, ou seja, o trecho que antecede imediatamente os blocos 3' e 4' da figura 5.10.
4. Caso contrário (referente aos itens 1, 2 e 3 acima), encaminha-se a entidade para o módulo **Dispose**. Este caso nunca ocorre, pois os casos possíveis de ocorrer já estão contemplados nas opções anteriores. Esta opção foi deixada para um eventual aumento da quantidade de níveis do fator “flexibilidade do operador”.
  - Módulo **Dispose** “saída01” - retira a entidade da simulação. Este caso também nunca ocorre, pois os casos possíveis de ocorrer já estão contemplados nas opções anteriores.

#### A2.4 Descrição Detalhada do Bloco “Sistema Livre” na Lógica de um Departamento.

A figura A2.4, que corresponde ao bloco 2 da figura 5.10, mostra o trecho do modelo correspondente ao bloco “Sistema Livre” na lógica de um departamento.

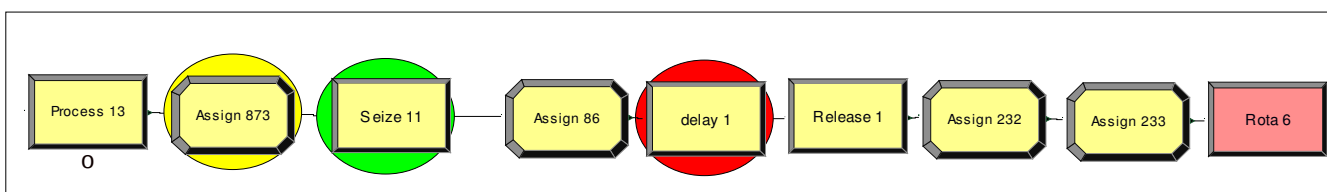


Figura A2.4 O bloco “Sistema Livre” na lógica de um departamento

Os módulos e suas configurações são:

- Módulo **Process** “Process 13” – Este módulo é incluído apenas para permitir monitorar o número de entidades que estão entrando neste bloco. O tempo de processamento (*delay*) é zero, e não retém nem ocupa nenhum recurso.
- Módulo **Assign** “Assign 873” – Fornece a um atributo o valor do tempo de processamento da entidade em uma das máquinas do departamento em questão, e no caso do exemplo o correspondente é o atributo “tempo\_processamento\_1”, pois trata-se do departamento 1. O valor atribuído foi “ $\text{expo}(0.1 * q)$ ”. Este módulo também executa as seguintes ações:
  - Fornece ao atributo “tempo\_processamento” o valor de “tempo\_processamento\_1”;
  - Atribui à variável “t\_p1” o valor “tempo\_processamento\_1”;
  - Fornece ao atributo “area\_enviado\_1” o valor zero;
  - Fornece ao atributo “opcao\_busca\_op\_1” o valor três;
  - Fornece ao atributo “p\_int\_livre\_1” o valor “ $\text{nummem}(\text{operador}_1) - \text{setsum}(\text{operador}_1, 5)$ ”.
- Módulo **Seize** “Seize 11” – a entidade solicita, com prioridade alta (*High*), dois recursos, e tal solicitação é contemplada, pois já se tinha verificado no módulo “Decide 8” no “bloco de entrada” que os dois recursos estavam livres. Cada um dos dois recursos fazem parte de um conjunto de recursos (**set**) distinto, isto é:
  - Conjunto de máquinas alocadas no departamento, sendo que no caso do exemplo, este conjunto corresponde ao **set** “Depto 01” (conjunto das quatro máquinas alocadas no departamento 1). É dada prioridade (**preferred Order**) à seleção da máquina pertencente ao conjunto, isto é, seleciona-se a primeira máquina livre por ordem de cadastramento na planilha deste **set**. Assim, no

exemplo considerado, é selecionada a primeira máquina livre do **set** “Depto 01”, na ordem de cadastro, ou seja, “Depto 01\_1”, “Depto 01\_2”, “Depto 01\_3” ou “Depto 01\_4”, conforme mostra a figura 5.3. A informação de qual recurso dentro do conjunto de recursos havia sido selecionado é armazenada no atributo “ind\_maq” (nome definido na caixa de diálogo do módulo **seize**, no campo **save attribute**), que armazena o índice do recurso selecionado dentro do **set**, e no caso do exemplo considerado, quando a máquina “Depto 01\_2” é selecionada, o valor armazenado é igual a 2.

- Conjunto de operadores alocados no departamento, que no exemplo em questão corresponde ao **set** “operador\_1” (conjunto dos três operadores alocados no departamento 1, ou seja, operador\_a, operador\_b ou operador\_c). É selecionada também a regra de prioridade **preferred Order** com o índice do operador escolhido armazenado no atributo “ind\_opr”.
- Caso não for possível reter estes dois recursos imediatamente, a entidade move-se para a fila “NOVA1111”. As entidades nesta fila são ordenadas pela regra SPT (**Shortest Processing Time**), de forma crescente, em função do menor valor (**Lowest Attribute Value**) do atributo “tempo\_processamento”.
- Módulo **Assign** “Assign 86” – este módulo executa as seguintes ações:
  - Armazena no atributo “ini\_1” o valor da função **Tnow**;
  - Atribui à variável “e\_1\_numero2” o valor “Entity.Type” (que identifica a família – 1, 2.. 5 – a que pertence o lote de peças);
  - Armazena no atributo “ini\_opr\_1” o valor da função **Tnow**.
- Módulo **Delay** “delay 1” – este módulo teve como objetivo reter a entidade durante “tempo\_processamento\_1+expo(3)” horas, simulando o tempo de processamento e o tempo de *setup* pleno.
- Módulo **Release** “Release 1” – através deste módulo, a entidade libera os dois recursos retidos no módulo **Seize** “Seize 11”, sendo que no exemplo em questão estes recursos são os seguintes: (a) o recurso de índice “ind\_maq” do **set** “Depto 01”; (b) o recurso de índice “ind\_opr\_1” do **set** “operador\_1”.
- Módulo **Assign** “Assign 232” – A finalidade deste módulo consiste em permitir exibir os tempos de utilização dos recursos neste bloco durante e depois da simulação. Este módulo executa as seguintes ações:
  - Atribui ao atributo “fini\_1” o valor da função **Tnow**, ao atributo “tempo\_saida\_1\_1” o valor “**Tnow**-tempo\_ent\_1\_1”, ou seja, armazena o

tempo que os recursos ficam efetivamente ocupados, permitindo-se obter posteriormente o tempo médio de ocupação desses recursos neste bloco;

- Atribui à variável “t\_s\_1\_1” o valor da função **Tnow**;
- Atribui à variável “placar\_1\_1” o “valor tempo\_saida\_1\_1”.
- Módulo **Assign** “Assign 233” – este módulo executa as seguintes ações:
  - Atribui a uma variável o tempo acumulado gasto por todas as entidades que saíam por esse bloco durante o tempo que permaneceram nele, sendo que no exemplo em questão a variável “Total\_tempo\_saida\_1\_1” recebe o valor “Total\_tempo\_saida\_1\_1+tempo\_saida\_1\_1”;
  - Atribui a uma variável a função de contador de saídas desse bloco em cada departamento, e no exemplo considerado a variável “Conta\_tempo\_saida\_1\_1” recebe o valor “Conta\_tempo\_saida\_1\_1+1”; ele também calcula continuamente o valor médio do tempo de uso dos recursos no bloco “Sistema Livre” de um departamento, atribuindo a uma variável o valor calculado, e no exemplo em questão é atribuído o valor “Total\_tempo\_saida\_1\_1/conta\_tempo\_saida\_1\_1” à variável “Media\_tempo\_saida\_1\_1”.
- Módulo **Route** “Rota 6” – este módulo encaminha a entidade (isto é, o lote de peças) para a próxima operação (ou à saída, no caso de encerramento de todas as operações) segundo o seu roteiro de fabricação, definido no seu atributo “Entity.Sequence”. O tempo de rota é dado pela função “UNIF(30, 60)” minutos.

## A2.5 Descrição Detalhada do Bloco “Blocos intermediários” na lógica de um departamento

Os módulos deste bloco e suas configurações são:

- Módulo **Assign** “Assign 884” – atribui a um atributo o tempo de processamento, sendo que no exemplo em questão armazena-se no atributo “tempo\_processamento\_1” o valor de “EXPO(0.1\*q)”. Este módulo também atribui à entidade uma referência de localização, e no exemplo considerado tal atributo é “area\_enviado\_1”, que recebe o valor 2. Finalmente, o atributo “tempo\_processamento” recebe o valor de “tempo\_processamento\_1”.
- Módulo **Seize** “Seize 33” – Solicita uma máquina do departamento em uma fila (no exemplo, “nova1”) até que uma das máquinas do processo desejado fique

livre. Essa fila do departamento está ordenada pelo SPT, assim a entidade da fila que tiver o menor tempo de processamento (se houver mais de uma solicitante) será a primeira a ocupar o recurso. A entidade solicita, com prioridade média (**Medium**), um recurso (máquina) do conjunto de recursos (**set**) de máquinas alocadas no departamento (no exemplo, este correspondia ao **set** “Depto 01”). É dada prioridade **preferred Order** na seleção da máquina pertencente ao conjunto, isto é, seleciona-se a primeira máquina livre por ordem de cadastramento na planilha desse **set**. A informação de qual recurso pertencente ao conjunto de recursos foi selecionado é armazenada então no atributo “ind\_maq”.

- Módulo **Station** “retorno 1” – esta é a porta de reentrada eventual da entidade que consegue ocupar (**seize**) uma das máquinas, porém não consegue depois obter (via **seize**) um operador.
- Módulo **Decide** “Decide 38” – este é um módulo de decisão com 2 possibilidades (falso ou verdadeiro) com relação à expressão “**SETSUM**(operador\_1,5) < **NUMMEM**(operador\_1)”. Assim, caso haja naquele momento algum operador disponível no **set** de operadores do próprio departamento (no exemplo, “operador\_1”), a entidade move-se para o bloco “Obtida a máquina obtém operador interno”, caso contrário ela move-se para o bloco “Obtida a máquina busca qualquer operador”. Todas as entidades que chegam neste trecho têm o  $\eta$  igual a 2 ou 3 e não conseguiram simultaneamente ocupar (via **seize**) uma máquina e obter um operador interno. Isto satisfaz a especificação do padrão adotado, que determina que se não houver operador disponível, será procurado um operador capacitado e livre em outro departamento.

## A2.6 Descrição Detalhada do Bloco “Obtida a máquina obtém operador interno”

Este bloco é constituído por 9 módulos, os quais são detalhados abaixo.

- Módulo **Assign** “Assign 77” – este módulo executa as seguintes ações:
  - Armazena no atributo “ini\_1” o valor da função **Tnow**;
  - Atribui à variável “numero” o valor “Entity.Type” (que identifica a família – 1, 2,.. 5 – a que pertence o lote de peças);
  - Armazena no atributo “opcao\_busca\_op\_1” o valor 1;

- Armazena no atributo “Op\_int\_livre\_1” a quantidade de operadores livres, que no caso do exemplo, corresponde à expressão “nummem(operador\_1)-setsum(operador\_1,5)”.
- Módulo **Assign** “Assign 67” – este módulo armazena no atributo “ini\_1” o valor da função **Tnow**. Além disso, ele armazena no atributo “Priority” o valor da expressão “1/(0.5+NQ(nova1)+NQ(nova11111))”, ou seja, este módulo faz com que a entidade que está numa fila maior tenha uma maior prioridade.
- Módulo **Seize** “Seize 34” – a entidade efetua um **seize** em um operador do **set** “operador\_1” (operadores internos do departamento, no exemplo o departamento 1) com regra **Preferred Order**, armazenando no atributo “ind\_opr\_1” a ordem no **set** do operador obtido. A obtenção de um operador interno evita tempos de ida e volta de operador(es) de outro departamento. Quando a entidade entra neste bloco (“Obtida a máquina obtém operador interno”) é porque há ao menos um operador interno disponível. Assim a fila desse módulo **Seize** (no exemplo considerado a fila “nova111”) nunca é usada.
- Módulo **Assign** “Assign 68” – este módulo armazena no atributo “ini\_opr\_1” o valor da função **Tnow**, e no atributo “Priority” o valor 2 (isto é, volta a estabelecer uma prioridade média).
- Módulo **Delay** “delay 52” – retém a entidade simulando o tempo de processamento e de *setup* pleno através da expressão “tempo\_processamento\_1 + expo(3)”
- Módulo **Release** “Release 53” – a entidade libera os dois recursos retidos anteriormente (no exemplo considerado uma máquina do conjunto “Depto 01” e um operador do conjunto “operador\_1”), mediante o emprego da regra de liberação “specific Member” e o uso dos índices dos respectivos **set** “ind\_maq” e “ind\_opr”.
- Módulo **Assign** “Assign 236” – este módulo armazena no atributo “fini\_1” o valor da função **tnow**, e também armazena no atributo “t\_sai\_1\_2” o valor da expressão “**tnow** - tempo\_ent\_1\_2”, ou seja, armazena o tempo que os recursos ficam efetivamente ocupados, permitindo-se obter posteriormente o tempo médio de ocupação desses recursos neste bloco. Atribui também à variável “placar\_1\_2” o valor “t\_sai\_1\_2”, ou seja, permite exibir durante e depois da simulação os tempos de utilização dos recursos neste bloco.
- Módulo **Assign** “Assign 234” – este módulo executa as seguintes ações:

- Atribui à variável “Total\_tempo\_saida\_1\_2” o valor da expressão “Total\_tempo\_saida\_1\_2 + tempo\_sai\_1\_2”;
- Atribui à variável “Conta\_tempo\_saida\_1\_2” o valor da expressão “Conta\_tempo\_saida\_1\_2 + 1”;
- Atribui à variável “Media\_tempo\_saida\_1\_2” o valor da expressão “Total\_tempo\_saida\_1\_2 / conta\_tempo\_saida\_1\_2”, ou seja, este módulo calcula continuamente o valor médio do tempo de uso dos recursos no bloco “Sistema Livre”.
- Módulo **Route** “Rota 36” – este módulo encaminha a entidade (o lote de peças) para a próxima operação (ou à saída, se todas as operações já foram encerradas) segundo o seu roteiro de fabricação, definido no seu atributo “Entity.Sequence”. O tempo de rota é dado pela função “UNIF( 30, 60)” minutos.

## A2.7 Descrição Detalhada do Bloco “Obtida a máquina busca qualquer operador”

Este bloco é constituído de vinte e um módulos, os quais são descritos abaixo.

- Módulo **Assign** “Assign 65” – este módulo executa as seguintes ações:
  - Armazena no atributo “ini\_1” o valor da função **tnow**;
  - Armazena no atributo “Priority” o valor da expressão “1 / (0,5 + **NQ**(nova1) + **NQ**(nova11111))”. Ou seja, faz com que a entidade que está numa fila maior tenha uma maior prioridade (a qual corresponde a um valor menor). Com isso se define o departamento para onde o operador deve ser transferido. O operador é transferido para o departamento solicitante que tenha a maior fila (LNQ - *Longest number of jobs*), desde que o operador seja habilitado para este trabalho e haja alguma máquina disponível neste departamento.
  - Armazena no atributo “Priority\_1” o valor do atributo “Priority”;
  - Armazena no atributo “opcao\_busca\_op\_1” o valor 2, registrando que a entidade passa por esse bloco dentro de departamento 2;
  - Armazena no atributo “Op\_int\_livre\_1” o valor da expressão “**nummem**(operador\_1) - **setsum**(operador\_1,5)”;
  - Armazena no atributo “fila Depto 01” a quantidade de entidades na fila “nova1” dada pela função **NQ**(nova1). A fila “nova1” é a fila das entidades que estão esperando por uma máquina do departamento 1, que procura efetuar o **seize** no módulo **seize** “seize 33” do bloco “intermediário”;



- Armazena no atributo “fila\_eta\_#\_operador\_1” o valor da função “**NQ**(nova11111)”, ou seja, o número de entidades na fila nova11111;
- Armazena no atributo “prioridade\_fila\_1” o valor da expressão “((Priority\_1)\*\*16) \* tempo\_processamento\_1”. Esta expressão busca garantir que quando dois departamentos diferentes estiverem requisitando um mesmo operador, a prioridade é dada em primeiro lugar àquele departamento que tenha a maior fila, e em caso de empate neste critério a prioridade é dada àquele que tenha o menor tempo de processamento.
- Módulo **Seize** “Seize 12” - a entidade tenta obter, através de um **seize**, um operador do **set** “eta\_2\_operador\_1” (operadores internos do departamento 1 e também de outros departamentos habilitados; caso  $\eta$  for igual a 3, o **set** seria “eta\_3\_operador\_1”), com regra “Cyclical”, ou seja, no caso de vários operadores estarem disponíveis vai alternando de forma cíclica qual deles será solicitado (esta opção foi incluída porque pode haver casos de *loops* no programa quando a opção é “preferred order”). Após o operador ser selecionado, armazena-se o seu número de ordem dentro do **set** no atributo “ind\_opr\_1”. A prioridade frente a outros comandos “**seize**” é dada pelo atributo “Priority\_1”. No exemplo considerado, os operadores que fazem parte deste **set** para as operações do departamento 1 são: “operador\_a”, “operador\_b”, “operador\_c”, “operador\_i”, “operador\_k”, “operador\_m” e “operador\_t”. Sendo apenas os três primeiros do departamento 1, os demais são operadores externos habilitados também a operar no departamento 1, e para uma melhor visualização esses últimos quatro aparecem envoltos por circunferências na figura 5.9. Enquanto não se consegue obter (via **seize**) um desses operadores, a entidade permanece na fila “nova11111”. Conforme abordado no módulo anterior (isto é, **Assign** “Assign 65”), as prioridades são dadas em primeiro lugar a entidades em recursos com filas maiores, e depois para aquelas cujo tempo de processamento é menor.
- Módulos **Route** “Rota 34” e “Rota 35”, **Assign** “Assign 81”, “Assign 82”, “Assign 83”, “Assign 84” e “Assign 906”, **Decide** “Decide 96”, **Release** “Release 129”, **Delay** “Delay 128” perderam efeito nessa tese e não serão descritos. Também parte do módulo **Decide** “Decide 95”. Esses módulos modelam a “regra quando” de Suresh e Slomp (2005), que como foi abordado no item 3.1, a mesma acabou não sendo adotada nesse trabalho de pesquisa, ainda que tenha sido deixada neste modelo porque funcionava e poderia ser útil para um futuro

trabalho. Assim, o bloco “Obtida a máquina busca qualquer operador” da figura 5.13 acabou ficando como ilustrado na figura A2.5.

- Módulo **Decide** “Decide 95” – uma vez que a entidade tenha conseguido obter algum operador, este módulo testa condições múltiplas. A primeira consiste em testar a expressão “ $\text{ind\_opr\_1} < 4$ ”, isto é, se o operador obtido é interno (no departamento 1 há três operadores). As demais opções são para modelar a “regra quando” que foi desconsiderada, conforme abordado anteriormente.
- Módulo **Assign** “Assign 66” - caso o módulo **Decide** “decide 95” resulte verdadeiro, ou seja, caso a entidade tenha obtido (através de um **seize**) um operador interno, a entidade é movida para este módulo, e atribui-se o valor zero ao atributo “ $\text{mov\_externo\_1}$ ” (o operador interno não gasta tempo de movimentação).
- Módulo **Assign** “Assign 75” – caso o módulo **Decide** “decide 95” resulte falso, a entidade segue por esse módulo, onde é atribuído o valor 1 ao atributo “ $\text{mov\_externo\_1}$ ” (os operadores externos gastam tempo de deslocamento), e ao atributo “ $\text{ini\_opr\_1}$ ” atribui-se o valor da função **Tnow**.
- Módulo **delay** “delay 11” – este módulo simula o tempo de processamento, acrescido do tempo de *setup* e do tempo de movimentação do operador (esta última parcela é considerada no caso da variável “ $\text{mov\_externo\_1}$ ” ser igual a 1) através da expressão “ $\text{tempo\_processamento\_1} + \text{expo}(3) + \text{mov\_externo\_1} * \text{UNIF}(0.5,1)$ ” horas. O valor unitário para o atributo “ $\text{mov\_externo\_1}$ ” atende a especificação do padrão adotado, que estabelece que no caso de ser encontrado um operador externo, este é transferido para o departamento de destino, gastando-se um tempo de transferência igual a  $\text{Unif}(30, 60)$  minutos.
- Módulo **Release** “Release 11” – este módulo libera a máquina do **set** “Depto 01”, cujo índice é “ $\text{ind\_maq}$ ”.
- Módulo **Release** “Release 12” – este módulo libera o operador do **set** “ $\text{eta\_2\_operador\_1}$ ” (ou “ $\text{eta\_3\_operador\_1}$ ”, se  $\eta$  for igual a três) cujo índice é “ $\text{ind\_opr\_1}$ ”.
- Módulo **Assign** “Assign 239” – o atributo “ $\text{fini\_1}$ ” recebe o valor da função **tnow**, e a variável “ $\text{t\_sai\_1\_3}$ ” recebe o valor da expressão “ $\text{TNOW} - \text{tempo\_ent\_1\_3}$ ”, e finalmente o valor 2 é armazenado no atributo “**Priority**”.

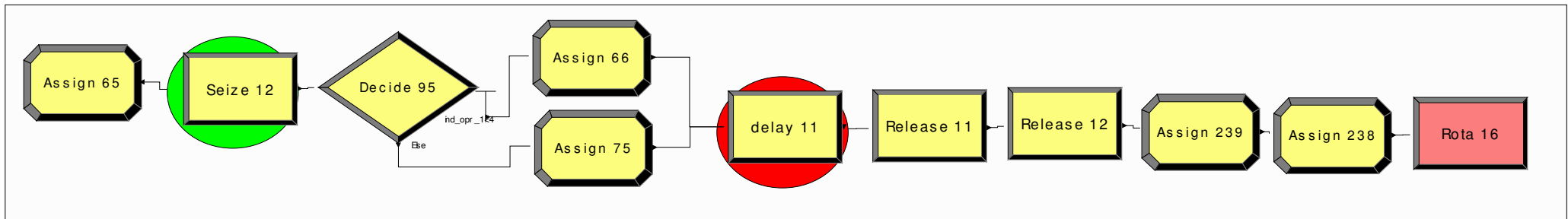


Figura A2.5 Os blocos “Obtida a máquina busca qualquer operador” simplificado

- Módulo **Assign** “Assign 238” – três variáveis auxiliam no cálculo do tempo médio da simulação neste trecho: a variável “Total\_tempo\_saida\_1\_3” recebe o valor da expressão “Total\_tempo\_saida\_1\_3 + t\_sai\_1\_3”; a variável “Conta\_tempo\_saida\_1\_3” recebe o valor da expressão “Conta\_tempo\_saida\_1\_3 + 1”; e a variável “Media\_tempo\_saida\_1\_3” recebe o valor da expressão “Total\_tempo\_saida\_1\_3 / conta\_tempo\_saida\_1\_3”.
- Módulo **Route** “Rota 16” – este módulo encaminha a entidade (isto é, o lote de peças) para a próxima operação (ou à saída, se já foram realizadas todas as operações) segundo o seu roteiro de fabricação, definido no seu atributo “Entity.Sequence”. O tempo de rota é UNIF( 30, 60) minutos.

## A2.8 Descrição Detalhada do Bloco “Saída dos dados de ocupação dos operadores e de lotes produzidos”

A seguir descrevem-se os módulos desse bloco e suas configurações.

- Módulo **Create** “Create 232” – este módulo cria uma única entidade lógica no término da simulação, às 3600 horas.
- Módulo **Record** “Record 3” – este módulo é o responsável pela coleta de dados estatísticos referentes à ocupação de todos os vinte e um operadores durante a simulação, além de também gravar, em conjunto com a planilha **statistic** do **template Advanced Process**, o resultado no arquivo “operator\_usage.dat”, no *desktop* do sistema Windows. O valor registrado é o resultado da seguinte expressão:

$$\text{“(DAVG}(\text{operador\_a.NumberBusy}) + \text{DAVG}(\text{operador\_b.NumberBusy}) + \dots + \text{DAVG}(\text{operador\_u.NumberBusy})) / 21\text{”},$$

expressão essa que é a soma das médias de ocupação de cada operador dividida pelo número total de operadores (que neste caso é igual a vinte e um).

- Módulo **Record** “Record 4” – este módulo coleta dados estatísticos do número de lotes de peças que saíram durante a simulação, e também grava, em conjunto com a planilha **statistic** do **template Advanced Process**, o resultado no arquivo “number out.dat”, no *desktop* do sistema Windows. O valor registrado é “saida.number”, que corresponde ao número de entidades que saíram pelo módulo **Dispose** denominado “saída”.

- Módulo **Dispose** “Dispose26” – este módulo retira do sistema a única entidade lógica criada no módulo **Create** “Create 232”.

### A2.9 Descrição Detalhada do Bloco “Verificação de necessidade de Warm up”

A seguir são descritos os módulos desse bloco e suas configurações.

- Módulo **Create** “Relógio” – Este módulo é o responsável pela criação de uma entidade lógica a cada 100 horas, sendo que a primeira entidade é criada às 100 horas.
- Módulo **Assign** “Produtividade” – neste módulo, a variável “Pecasporthora” recebe o valor da expressão “(Processo Expedicao.NumberOut – saidaanterior) / 100”, enquanto a variável “saidaanterior” recebe o valor “Processo Expedicao.NumberOut”. Nota-se que a produtividade é calculada pelo número de peças que foram produzidas em 100 horas, dividido por 100, ou seja, o resultado é o número médio de peças por hora. No próximo capítulo serão apresentados os resultados obtidos neste módulo.
- Módulo **Dispose** “Dispose 13” – este módulo retira do sistema as entidades lógicas criadas neste trecho da simulação.

### A2.10 Descrição Detalhada do Bloco “Saída das Entidades”

Os módulos desse bloco e suas configurações são descritos a seguir:

- Módulo **Station** “Estacao Expedicao” – este módulo recebe o último estágio das seqüências de operação de todos os lotes de peças.
- Módulo **Process** “Processo Expedicao” – usado para permitir a visualização na animação, além de possibilitar a visualização, após a simulação, do número de entidades que saíram. O tempo de processamento no módulo (isto é, o **Delay**) é igual a zero minutos, e portanto o mesmo não influi no tempo de fluxo das entidades.
- Módulo **Decide** “Decide 196” – testa se a expressão “**tnow** > 1800” é verdadeira. Na prática este módulo é utilizado para permitir que um bloco só seja acessado pelas entidades quando o tempo de *warm up* de 1800 horas for atingido. Só após este tempo os módulos “Record 1”, “Record 2” e “soma tempo” são acessados.

- Módulo **Record** “Record 1” – este módulo recolhe os tempos que cada entidade (lote) permanece na simulação (**TNOW**-início), ou seja, o tempo de fluxo (*flow Time*). O módulo **Statistic** grava esses dados de todas as entidades em um arquivo chamado “flow\_time.dat” localizado na área de *desktop* do sistema Windows.
- Módulo **Record** “Record 2” – este módulo recolhe os tempos de fila de cada entidade (lote) durante a simulação, ou seja, o atributo “Entity.WaitTime”. O módulo **Statistic** grava esses dados referentes a todas as entidades que saem do sistema, em um arquivo chamado “queue\_time.dat” na área de *desktop* do sistema operacional Windows.
- Módulo **Assign** “Soma tempos” – este módulo executa as seguintes ações:
  - atribui à variável “soma\_tempos\_fluxo” o valor da expressão “soma\_tempos\_fluxo + (tnow-início)”, ou seja, os tempos de fluxo acumulados;
  - atribui à variável “qtde\_pecas\_somadas” o valor da expressão “qtde\_pecas\_somadas+1”, ou seja, o resultado da contagem das peças que passaram por esse trecho do modelo;
  - atribui à variável “media\_tempo\_fluxo” o valor da expressão “soma\_tempos\_fluxo / qtde\_pecas\_somadas”, obtendo-se assim o tempo médio de fluxo depois do “warm up”.
- Módulo **Assign** “Assign 905” – este módulo atribui à variável “t\_e\_11\_1” o valor da função “tnow”, para verificar se de fato o desvio ocasionado pelo Módulo **Decide** “Decide 196” não estaria afetando também a segunda parte do bloco “Saída das Entidades”. Por outro lado atribui à variável “resultado” o valor da expressão “ind\_opr\_1 + ind\_opr\_2 + ind\_opr\_3 + ind\_opr\_4 + ind\_opr\_5 + ind\_opr\_6 + ind\_opr\_7 + ind\_opr\_8 + ind\_opr\_9 + ind\_opr\_10”. Esta variável foi utilizada como um primeiro teste para verificar se esses índices estavam de fato armazenando valores, o resultado esperado então era um valor maior que 0.
- Módulo **Decide** “Decide 194” – responsável por testar se a expressão “conta\_saidas < 99 && 1000 < 1” é verdadeira. Uma descrição mais detalhada deste módulo será apresentada no capítulo 7
- Módulo **ReadWrite** “Verificação” - este módulo foi introduzido para fazer verificações que serão abordadas no capítulo 6. Ele transfere os valores de várias variáveis para um arquivo Excel, sendo que as variáveis que o módulo

coleta e transfere são as seguintes: "valor\_eta\_1", "tempo\_ent\_1\_1", "opcao\_busca\_op\_1", "Op\_int\_livre\_1", "ini\_1", "ini\_opr\_1", "ind\_opr\_1", "tempo\_processamento\_1", "Priority\_1", "fila Depto 01", "fila eta\_#\_operador\_1", "prioridade\_fila\_1", e a expressão "tempo\_saida\_1\_1+t\_sai\_1\_2+t\_sai\_1\_3". No caso de outros departamentos (p.ex. departamento "2"), as variáveis seriam equivalentes (isto é, "valor\_eta\_2", "tempo\_ent\_2\_1" etc).

- Módulo **Assign** "Assign 927" – responsável por atribuir à variável "conta\_saidas" o valor da expressão "conta\_saidas+1". Esta variável faz a contagem da quantidade de entidades que passaram por este trecho do modelo.
- Módulo **Dispose** "saida" – este módulo retira as entidades do sistema em simulação.

### A2.11 Descrição do Bloco "Geração de operador sempre ocupado"

Os módulos desse bloco e suas configurações são descritos a seguir.

- Módulo **Create** "Create 8" – este módulo cria uma única entidade lógica no instante zero da simulação.
- Módulo **Seize** "Seize 35" – esta entidade lógica efetua um **seize** em um operador chamado "operador\_sempre\_ocupado", que vai permanecer assim ocupado até o fim da simulação.
- Módulo **Dispose** "Dispose 2" – este módulo retira da simulação a entidade lógica criada anteriormente no módulo **Create** "Create 8".

### A2.12 Descrição da Lógica de uma célula

Os módulos envolvidos neste bloco são:

- Módulo **Station** "Estacao Depto 02 Maq 2" - recebe as entidades que são movidas para a estação "Depto 02\_2\_st".
- Módulo **Assign** "Assign 260" – este módulo atribui o valor da expressão "expo(0.1\*q)" ao atributo "tempo\_processamento". Ou seja, ele atribui à entidade o valor do tempo de processamento da próxima operação.
- Módulo **Process** "Process 19" – este módulo foi acrescentado apenas para permitir monitorar o número de entidades que entram neste bloco. Não gasta tempo das

entidades (o tempo de **delay** é igual a zero), além de não reter nem ocupar nenhum recurso.

- Módulo **Seize** “Seize 46” – estando a máquina “Depto 02\_2” disponível, efetua-se o **seize** à entidade, caso contrário a entidade permanece na fila “nova1\_22” até que a máquina fique livre. A fila é ordenada em ordem crescente pelo atributo “tempo\_processamento” da entidade.
- Módulo **Assign** “Assign 291” – este módulo atribui ao parâmetro “Priority” o valor da expressão “ $1/(0.5+NQ(nova1\_22))$ ”, ou seja, o valor do parâmetro “Priority” é inversamente proporcional ao tamanho da fila. Além disso, teste módulo atribui também o valor da expressão “ $((Priority)**16)*tempo\_processamento$ ” ao parâmetro “prioridade\_fila”, ou seja, esse atributo recebe um valor inversamente proporcional ao tamanho da fila e ao tempo de processamento. Dar-se-á uma maior prioridade ao tamanho da fila.
- Módulo **Decide** “Decide 59” – este módulo se responsabiliza por testar se “eta” é igual a 2, ou seja, se a flexibilidade do operador é dupla, e neste caso ele envia a entidade para o módulo **Seize** “Seize 47”, caso contrário (isto é, caso “eta” for igual a 3), este módulo envia a entidade para o módulo **Seize** “Seize 118”.
- Módulo **Seize** “Seize 47” – este módulo busca efetuar um **seize** em algum dos operadores do **set** “opr\_d2\_m2”. Na configuração celular tradicional todos os **set** de operadores referem-se a operadores internos da própria célula. No exemplo o **set** “opr\_d2\_m2” é constituído pelos recursos “operador\_sempre\_ocupado” e “operador\_f”. Conforme mencionado anteriormente, esse primeiro operador está sempre indisponível. Quando o segundo operador é obtido, atribui-se o valor dois ao atributo “ind\_opr”, índice do **set**. Por outro lado, enquanto a entidade não obtiver o operador, ela espera na fila “nova2\_22”, ordenada por “prioridade\_fila”. A entidade ao tentar efetuar um **seize** em um operador segue a regra “Preferred Order”, o que confere preferência ao operador que opera esta máquina como sua primeira habilidade. Dessa forma se evitam os tempos de movimentação dentro da célula.
- Módulo **Seize** “Seize 118” – idêntico ao módulo **Seize** “Seize 47”, mas para quando “eta” é igual a 3. Assim, o operador é solicitado dentre os disponíveis no **set** “ex\_3\_opr\_d2\_m2”, composto pelos recursos “operador\_sempre\_ocupado”, “operador\_f” e “operador\_r”. Enquanto não obtiver o operador a entidade espera na fila “nova2\_22b”, ordenada por “prioridade\_fila”. Depois, o atributo “ind\_opr” da entidade recebe o índice do operador obtido.



- Módulo **Assign** “Assign 87” – este módulo zera o valor do atributo “mov\_externo”.
- Módulo **Decide** “Decide 14” - este módulo testa se a expressão “ind\_por > = 2” é verdadeira, ou seja, se o operador controlado não tem como primeira habilidade a de operar esta máquina. Neste caso, este módulo encaminha a entidade para o módulo **Assign** “Assign 86”, caso contrário a entidade é enviada para o módulo **Delay** “delay 62”.
- Módulo **Assign** “Assign 86” – este módulo atribui o valor 1 ao atributo “mov\_externo”. Ou seja, como o teste do módulo Decide “Decide 14” resultou verdadeiro, isto significa que houve movimentação de operador para atender a máquina, e este tempo tem que ser computado no próximo módulo.
- Módulo **Delay** “delay 62” – responsável por simular o tempo de processamento da máquina mais o de movimentação de operador (se for o caso), mais o tempo de *setup* reduzido através do valor da expressão “tempo\_processamento + mov\_externo \* unif(0.05, 0.1) + EXPO(3) \* setup\_reduz” horas. Caso o operador opere a máquina como a sua primeira habilidade, o valor da variável mov\_externo é nulo porque não ocorre movimentação, e assim, o tempo de movimentação, que normalmente é dada pela expressão “unif(0.05, 0.1)”, é desconsiderado. A expressão EXPO(3) \* setup\_reduz define o valor do *setup*. A variável “setup\_reduz” define o nível de redução do *setup* através de dois valores: 0,6 e 0,25, que geram respectivamente reduções de 40 % e 75% do tempo de *setup*.
- Módulo **Release** “Release 66” – libera a máquina “Depto 02\_2”.
- Módulo **Decide** “Decide 60” – este módulo testa se “eta” é igual a 2, ou seja, se a flexibilidade do operador é dupla. Caso isto ocorra, este módulo envia a entidade para o módulo **Release** “Release 232”, caso contrário (isto é, se “eta” for igual a 3), a entidade é enviada para o módulo **Release** “Release 235”.
- Módulo **Release** “Release 232” – libera um operador específico (o de índice “ind\_opr” do conjunto de operadores “opr\_d2\_m2”), que pode operar a máquina 2 na célula 2 quando eta=2.
- Módulo **Release** “Release 235” – libera o operador específico (o de índice “ind\_opr” do conjunto de operadores “ex\_3\_opr\_d2\_m2”), que pode operar a máquina 2 na célula 2 quando eta=3.

- Módulo **Route** “Rota 11” – este módulo encaminha a entidade (o lote de peças) primeiramente para o módulo **Station** “Gravador”, responsável por armazenar dados para poder realizar a verificação do modelo. Então a entidade se dirige para a próxima etapa descrita na sua seqüência de operações, que normalmente corresponde a uma outra máquina da mesma célula, ou para a saída do sistema, caso a entidade já tenha concluído todas as operações previstas. O tempo de rota é dado pela função “unif(0.05, 0.1)” horas.

### A2.13 Descrição Detalhada do Bloco “Saída das Entidades”

Os módulos e as configurações desse bloco são descritos a seguir.

- Módulo **Station** “Estacao Expedicao” – responsável pela entrada das entidades na estação “Expedicao\_st”. Este módulo recebe o último estágio das seqüências de operação de todos os lotes de peças.
- Módulo **Process** “Processo Expedicao” – este módulo permite a visualização na animação, e também do número de entidades que saíram. O tempo de processamento no módulo (o **Delay**) é igual a zero minutos, não influenciando assim no tempo de fluxo das entidades.
- Módulo **Decide** “Decide 113” – responsável por testar se a expressão “tnow > 1800” é verdadeira. Na prática, ele é utilizado para permitir que um bloco seja acessado pelas entidades somente quando o tempo de *warm up* de 1800 horas for atingido. Só após este tempo os módulos “Record 1”, “Record 2” e “soma tempo” são acessados.
- Módulo **Record** “Record 1” – recolhe os tempos em que cada entidade (lote) permanece na simulação “**TNOW** – inicio”, ou seja, o tempo de fluxo (*flow Time*). O módulo **Statistic** grava esses dados de todas as entidades em um arquivo chamado “flow\_time.dat” na área de *desktop* do sistema operacional Windows.
- Módulo **Record** “Record 2” – recolhe os tempos de fila em que cada entidade (lote) permanece durante a simulação, ou seja, o atributo “Entity.WaitTime”. O módulo **Statistic** grava esses dados de todas as entidades em um arquivo chamado “queue\_time.dat” na área de *desktop* do sistema operacional Windows.
- Módulo **Assign** “Soma tempos” – este módulo executa as seguintes ações:

- atribui à variável “soma\_tempos\_fluxo” o valor da expressão “soma\_tempos\_fluxo + (tnow-inicio)”, ou seja, os tempos de fluxo acumulados;
  - atribui à variável “qtde\_pecas\_somadas” o valor da expressão “qtde\_pecas\_somadas+1”, ou seja, conta as peças que passaram por esse trecho do modelo;
  - atribui à variável “media\_tempo\_fluxo” o valor da expressão “soma\_tempos\_fluxo / qtde\_pecas\_somadas”, obtendo assim o tempo médio de fluxo depois do *warm up*.
- Módulo **Dispose** “saida” – retira as entidades do sistema em simulação.

#### A2.14 Descrição Detalhada do Bloco “Transferência de dados para o Excel”

A seguir se descreve os módulos e as configurações desse bloco:

- Módulo **Station** “Estacao Expedicao” – responsável pela entrada das entidades na estação “gravador”. As entidades são enviadas para esta estação ao fim de cada processo em cada uma das máquinas das células.
- Módulo **Decide** “Decide 110” – este módulo testa se a expressão “eta == 10000” é verdadeira. Um detalhamento maior deste módulo será apresentado no capítulo 7.
- Módulo **ReadWrite** “Verificação” - este módulo foi introduzido para fazer verificações que serão detalhadas no capítulo 6. Ele é o responsável por transferir os valores de vários atributos para um arquivo Excel. Os atributos que o módulo coleta e transfere são os seguintes: "Entity.Sequence", "maquina", "t\_solicita\_maq", "inic\_opr", "t\_posse\_opr", "fine\_opr", "tempo\_processamento", "mov\_externo", "Priority" e "prioridade\_fila".
- Módulo **Route** “Rota 136” – responsável por encaminhar a entidade (o lote de peças) para a próxima etapa descrita na sua seqüência de operações, que normalmente corresponde a uma outra máquina pertencente à mesma célula de onde a entidade acabava de sair, ou para a saída do sistema caso todas as operações previstas já tenham sido encerradas. O tempo de rota é dado pela função “unif(0.05, 0.1)” horas.

### A2.15 Descrição Detalhada do Bloco “Entrada” na lógica de um departamento.

A seguir são descritos os módulos e as configurações desse bloco:

- Módulo **Station** “Estacao Depto 01” – recebe as entidades que chegam à estação de nome “Depto 01\_st”.
- Módulo **Process** “Process 36” – este módulo foi acrescentado para permitir monitorar o número de entidades que entram neste bloco. O tempo de **delay** é igual a zero, e não retém nem ocupa nenhum recurso.
- Módulo **Assign** “Família”. Este módulo executa as seguintes ações:
  - Atribui à variável “visual” o valor “entity.type”, de maneira a indicar no gráfico “Variable visual” a família da entidade que está passando pelo bloco;
  - Atribui o valor da função **tnow** ao atributo “t\_seiszemaq\_d1”, iniciando a contagem do tempo em que uma máquina do departamento 1 está sendo usada pela entidade;
  - Zera o atributo “ind\_opr”;
  - Atribui o valor da expressão “expo(0.1\*q)” ao atributo “tempo\_processamento”, definindo assim o tempo de processamento dessa entidade no departamento 1;
  - Atribui o valor do atributo “tempo\_processamento” ao atributo “t\_p\_1”. Atribui o valor da expressão “reductor\_setup \* EXPO(3)” ao atributo “t\_setup\_1”, que é o tempo de *setup*, neste caso com valor reduzido;
  - Atribui à variável “t\_inicio\_1” o valor da função **tnow**;
  - Zera o atributo “mov\_externo” da entidade, o que significa que não há movimentação do operador;
  - Atribui à variável “tt\_setup\_1” o valor do atributo “t\_setup\_1”. Atribui também o valor do atributo “entity.type” ao atributo “família”.
- Módulo **Decide** “Decide 200” – este módulo testa condições múltiplas. No exemplo sendo considerado, por se tratar do departamento 1, há mais testes porque são quatro máquinas, os quais são:
  1. se “fppme(1,1)==entity.type && **STATE**(Depto 01\_1) == - 1”
  2. se “fppme(1,2)==entity.type && **STATE**(Depto 01\_2) == - 1”
  3. se “fppme(1,3)==entity.type && **STATE**(Depto 01\_3) == - 1”
  4. se “fppme(1,4)==entity.type && **STATE**(Depto 01\_4) == - 1”

5. se “**STATE**(Depto 01\_1) == - 1 || **STATE**(Depto 01\_2) == - 1 || **STATE**(Depto 01\_3) == - 1 || **STATE**(Depto 01\_4) == - 1”
  6. se “fppme(1,1)==entity.type”
  7. se “fppme(1,2)==entity.type”
  8. se “fppme(1,3)==entity.type”
  9. se “fppme(1,4)==entity.type”
- Como no software Arena não há variáveis indexadas que possam armazenar as informações de qual família de peça ocupa ou ocupou, por último, cada máquina, se fez uso do conceito “expressão” do software Arena, a qual trabalha como matriz. Neste caso, a linha é o departamento e a coluna a máquina do departamento, e assim a expressão “fppme(1,2)==3” indica que a máquina 2 do departamento 1 está operando ou acabou de operar uma peça da família 3. Na planilha **Expression** do **template Advanced Process**, os dados da expressão “fppme” são, nesta ordem, as variáveis “fpe1m1”, “fpe2m1”, “fpe3m1”, ... “fpe10m1”, “fpe1m2”, “fpe2m2”,..., “fpe10m2”, “fpe1m3”, “fpe2m3”, ..., “fpe10m3”, “fpe1m4”, “fpe2m4”, ..., “fpe10m4”, totalizando quarenta variáveis. A única forma de se conseguir mudar os dados das expressões é através de variáveis constantes na sua planilha de dados. Desse modo, este módulo testa se já há alguma máquina configurada para processar uma peça de uma determinada família, e no caso da existência desta máquina, a entidade (o lote de peça) é direcionada para sofrer uma operação por esta máquina. A lógica desse **decide** é verificar, em primeiro lugar, se há alguma máquina livre em que a sua última operação tenha sido com peças da mesma família da entidade solicitante (testes de 1 a 4). Depois, o módulo testa se há alguma máquina livre, ainda que, naturalmente, não esteja com o *setup* adequado (teste 5). A seguir, o módulo testa se há alguma máquina ocupada operando um lote de peças da mesma família que a entidade solicitante, conseqüentemente já estando com o *setup* pronto. Caso contrário, a entidade espera por qualquer máquina que fique livre (condição **else** por eliminação).
    1. Módulos **Assign** “Assign 454”, “Assign 447”, “Assign 448”..., “Assign 453”, “Assign 455” e “Assign 456” – recebem, nesta ordem, as entidades procedentes respectivamente das condições de 1 a 9 do módulo “Decide 200”. Cada um desses módulos **Assign** atribui ao parâmetro “opc\_1” o valor da opção em que o teste do módulo “Decide 200” resultou verdadeiro. Assim, no exemplo considerado, se no primeiro teste do

módulo “Decide 200”, isto é, “fppme(1,1)==entity.type && **STATE**(Depto 01\_1) == - 1”, o resultado for verdadeiro, a entidade vai para o módulo **Assign** “Assign 454”, que atribui o valor 1 ao parâmetro “opc\_1”.

## A2.16 Descrição Detalhada do Bloco “Operando com máquina já configurada”

A seguir são descritos os módulos e as configurações desse bloco:

- Módulo **Seize** “Seize 1” – a entidade efetua um **seize** na máquina “Depto 01\_1”, com prioridade alta (*Priority: High(1)*). Caso a máquina esteja ocupada, a entidade espera na fila “Seize 1.Queue”. As entidades são ordenadas nesta fila pelo menor valor do atributo “tempo\_processamento”.
- Módulo **Assign** “Assign 80” – este módulo executa as seguintes ações:
  - Atribui o valor 1 ao atributo “maq\_usada\_1”, indicando, para futura verificação, que esta foi a máquina utilizada pela entidade no departamento 1;
  - Atribui o valor da função **tnow** ao atributo “t\_ini\_uso\_maq\_d1” para indicar o tempo que começou a utilizar esta máquina;
  - Atribui à variável “fpe1m1” o valor “entity.type”, para indicar qual a família que está usando ou usou por último esta máquina. Ou seja, a variável em questão recebe o número correspondente à família da entidade que acaba de efetuar um **seize** na máquina 1 do departamento 1;
  - Atribui o valor da função **tnow** ao parâmetro “ini\_1”.
- Módulo **Decide** “Decide 102” – este módulo testa se “eta” é igual a 1, 2 ou 3, ou seja, se a flexibilidade do operador é nula, dupla ou tripla.
- Módulo **Seize** “Seize 166” – no caso de “eta” ser igual a 1, este módulo tentar efetuar um **seize** com prioridade alta (*Priority: High(1)*) em algum dos operadores do **set** “operador\_1”. Quando o segundo operador é obtido atribui-se o valor dois ao atributo “ind\_opr\_1”, índice do set com regra “Preferred Order”. Por outro lado, enquanto a entidade não obtiver o recurso, ela espera na fila “Seize 166.Queue”, ordenada pelo valor mais baixo do atributo “tempo\_processamento”.
- Módulo **Seize** “Seize 5” – Idêntico ao Módulo **Seize** “Seize 166”, mas para quando “eta” é igual a 2. Neste caso, o operador é solicitado dentre os do **set** “eta\_2\_operador\_1” com prioridade alta (*Priority: High(1)*). Enquanto não obtiver o operador, a entidade espera na fila “Seize 5.Queue”, ordenada pelo valor mais

baixo do atributo “tempo\_processamento”. Depois, o atributo “ind\_opr\_1” da entidade”, que corresponde ao índice do **set** com regra “Preferred Order”, recebe o índice do operador obtido.

- Módulo **Seize** “Seize 113” – Idêntico aos módulos anteriores, mas para quando  $\eta$  for igual a 3. Assim o operador é solicitado dentre os do **set** “eta\_3\_operador\_1” com prioridade alta (*Priority: High(1)*). Enquanto não obtiver o operador a entidade espera na fila “Seize 113.Queue”, ordenada pelo valor mais baixo do atributo “tempo\_processamento”. Depois, o índice do operador obtido é atribuído ao parâmetro “ind\_opr\_1” da entidade, que é o índice do **set** com regra “Preferred Order”.
- Módulo **Decide** “Decide 49” – uma vez que a entidade tenha conseguido efetuar um **seize** em algum operador, este módulo testa a expressão “ind\_opr\_1 >= 4”. Isto é, se o operador for externo (no departamento 1 há três operadores internos), a entidade segue para o módulo “Assign 212”, caso contrário ela move-se diretamente para o módulo “Assign 368”.
- Módulo **Assign** “Assign 212” – este módulo atribui ao parâmetro “mov\_externo\_1” o valor 1, ou seja, a movimentação do operador será considerada, devido ao fato do mesmo ser externo.
- Módulo **Assign** “Assign 368” – este módulo atribui o valor da expressão “mov\_externo \* UNIF(0.5,1)” ao atributo “t\_mov\_ext\_1”, ou seja, ele calcula o tempo de movimentação do operador. Se “mov\_externo\_1” for nulo, isto significa que o operador é interno, e nesse caso o tempo de movimentação será nulo também. Este módulo atribui também o valor da função **tnow** ao atributo “t\_ini\_uso\_oper\_d1”, isto é, o tempo em que o operador está ocupado começa a ser contado.
- Módulo **Delay** “delay 10” – responsável por simular o tempo de processamento, de movimentação e de *setup* através da expressão “t\_p\_1 + t\_mov\_ext\_1 + t\_setup\_1” horas.
- Módulo **Release** “Release 61” – a entidade libera a máquina “Depto 01\_1”.
- Módulo **Decide** “Decide 106” – este módulo testa se “eta” é igual a 1, 2 ou 3, ou seja, se a flexibilidade do operador é nula, dupla ou tripla.
- Módulo **Release** “Release 219” – a entidade libera um operador do **set** “operador\_1”, mediante o emprego da regra de liberação (**release rule**) “specific

Member” e o uso dos índices (**Set Index**) “ind\_opr\_1”, ou seja, a entidade libera o operador que foi empregado no módulo “Seize 166”

- Módulo **Release** “Release 62” – a entidade libera um operador do **set** “eta\_2\_operador\_1”, mediante o emprego da regra de liberação (**release rule**) “specific Member” e o uso dos índices (**Set Index**) “ind\_opr\_1”, ou seja, a entidade libera o operador que foi empregado no módulo “Seize 5”.
- Módulo **Release** “Release 167” – a entidade libera o operador do **set** “eta\_3\_operador\_1”, mediante o emprego da regra de liberação (**release rule**) “specific Member” e o uso dos índices (**Set Index**) “ind\_opr\_1”, ou seja, libera o operador que foi empregado no módulo “Seize 113”.
- Módulo **Assign** “Assign 369” – Este módulo executa as seguintes ações:
  - Atribui o valor da função **tnow** à variável “t\_fim\_1”, indicando, para uma futura verificação, o tempo de término do uso de algum dos recursos;
  - Atribui o valor da expressão “UNIF(30, 60)/60” ao atributo “t\_rota\_1”. Assim gera o tempo de rota, com um valor entre 30 a 60 minutos, segundo uma distribuição uniforme, que a entidade (do lote) irá gastar para se movimentar até a máquina, que irá realizar a próxima operação, ou até a saída da célula, caso todas as operações previstas já tenham sido realizadas.
  - Atribui o valor da função **tnow** ao atributo “fini\_1”, que corresponde ao tempo de término do uso de algum dos recursos.
- Módulo **Route** “Rota 36” - encaminha a entidade (o lote de peças) para a próxima operação segundo o seu roteiro de fabricação, definido no seu atributo “Entity.Sequence”. O tempo de rota é dado pelo valor em horas do atributo “t\_rota\_1”. Caso todas as operações tenham sido concluídas, a entidade é encaminhada para a saída.

#### **A2.17 Descrição Detalhada do Bloco “Operando com máquina não configurada”**

A seguir são descritos os módulos e as configurações deste bloco.

- Módulo **Process** “Process 35” – Este módulo foi acrescentado apenas para permitir monitorar o número de entidades que entram neste bloco. O tempo de **delay** é igual a zero, e não retém nem ocupa nenhum recurso.



- Módulo **Seize** “Seize 10” – a entidade efetua um **seize** no **set** de máquinas “Depto 01”, com prioridade média (*Priority: Medium(2)*). Caso todas as máquinas do **set** estejam ocupadas, a entidade espera na fila “nova11111”. As entidades são ordenadas nesta fila pelo menor valor do parâmetro “tempo\_processamento”. Depois, o índice da máquina obtida é atribuído ao atributo “ind\_maq\_1” da entidade, que é o índice do **set** com regra “Preferred Order”.
- Módulo **Assign** “Assign 457” – este módulo executa as seguintes ações:
  - Atribui o valor da função **tnow** ao atributo “t\_ini\_uso\_maq\_d1”, para indicar o tempo que a entidade começou a utilizar esta máquina;
  - Atribui o valor da função **tnow** ao atributo “t\_seizeoper\_d1”, para indicar o tempo que a entidade começou a efetuar o **seize** no operador;
  - Atribui o valor do atributo “maq\_usada\_1” ao parâmetro “maq\_usada\_1”, indicando, para uma futura verificação, qual foi a máquina utilizada pela entidade no departamento 1.
- Módulo **Decide** “Decide 11” – este módulo testa se o atributo “ind\_maq\_1” é igual a 1, 2, 3 ou 4, ou seja, se a máquina obtida é a “Depto 01\_1”, “Depto 01\_2”, “Depto 01\_3” ou “Depto 01\_4” (no caso do “Depto 01” há quatro máquinas).
- Módulos **Assign** “Assign 92” e também “Assign 93”, “Assign 94” e “Assign 95” – Cada um destes módulos executam as seguintes ações:
  - Atribui à variável “fpe1m1” (ou “fpe1m1”, “fpe1m1” ou “fpe1m1”, nos módulos **Assign** sucessivos) o valor do atributo “entity.type”, ou seja, atribui à máquina utilizada a família de peças que a mesma está processando;
  - Atribui à variável “num\_entid\_e1\_maq1” (ou “num\_entid\_e1\_maq2”, “num\_entid\_e1\_maq3” ou “num\_entid\_e1\_maq1”, nos módulos **Assign** sucessivos) o valor do atributo IDENT, ou seja, associa à máquina utilizada a entidade solicitante;
  - Atribui o valor da função expo(3) ao atributo “t\_setup\_1”, considerando assim o tempo de *setup* como sendo pleno (sem redução);
  - Atribui à variável “tt\_setup\_1” o valor do atributo “t\_setup\_1”.
- Módulo **Decide** “Decide 167” – testa se “eta” é igual a 1, 2 ou 3, ou seja, se a flexibilidade do operador é nula, dupla ou tripla.

- Módulo **Seize** “Seize 166” - tenta efetuar um **seize** com prioridade média (*Priority: Medium(2)*) em algum dos operadores do **set** “operador\_1”. Quando o segundo operador é obtido atribui-se o valor dois ao atributo “ind\_opr\_1”, índice do **set** com regra “Preferred Order”. Por outro lado, enquanto a entidade não obtiver o recurso, ela espera na fila “nova1111111c”, ordenada pelo valor mais baixo do atributo “tempo\_processamento”.
- Módulo **Seize** “Seize 112” – Idêntico ao Módulo **Seize** “Seize 166”, mas para quando “eta” for igual a 2. Assim, o operador é solicitado dentre aqueles do **set** “eta\_2\_operador\_1” com prioridade média (*Priority: Medium(2)*). Enquanto não obtiver o operador, a entidade espera na fila “nova11111111”, ordenada pelo valor mais baixo do atributo “tempo\_processamento”. Depois, o índice do operador obtido é atribuído ao parâmetro “ind\_opr\_1” da entidade, índice do **set** com regra “Preferred Order”.
- Módulo **Seize** “Seize 146” – Idêntico aos módulos anteriores, mas para quando  $\eta$  for igual a 3. Assim, o operador é solicitado dentre aqueles do **set** “eta\_3\_operador\_1” com prioridade média (*Priority: Medium(2)*). Enquanto não obtiver o operador, a entidade espera na fila “nova11111111b”, ordenada pelo valor mais baixo do atributo “tempo\_processamento”. Depois, o índice do operador obtido é atribuído ao parâmetro “ind\_opr\_1” da entidade, índice do **set** com regra “Preferred Order”.
- Módulo **Decide** “Decide 55” – uma vez que a entidade tenha conseguido obter (via **seize**) algum operador, este módulo testa “ind\_opr\_1 > = 4”. Isto é, se o operador for externo (no departamento 1 há três operadores internos), a entidade segue para o módulo “Assign 218”, caso contrário ela move-se diretamente para o módulo “Assign 374”.
- Módulo **Assign** “Assign 218” – responsável por atribuir ao parâmetro “mov\_externo\_1” o valor 1, ou seja, irá incluir a movimentação do operador pelo fato do mesmo ser externo.
- Módulo **Assign** “Assign 374” – atribui o valor da expressão “mov\_externo \* UNIF(0.5,1)” ao atributo “t\_mov\_ext\_1”, ou seja, calcula o tempo de movimentação do operador. Percebe-se que, se “mov\_externo\_1” for nulo (ou seja, o operador é interno), o tempo de movimentação será nulo também. Além disso, este módulo atribui também o valor da função **tnow** ao atributo “t\_ini\_uso\_oper\_d1”, isto é, ele começa a contar o tempo em que o operador

está ocupado. Finalmente, este módulo atribui o valor da função **tnow** ao atributo “ini\_1”.

- Módulo **Delay** “delay 6” – responsável por simular o tempo de processamento, de movimentação e de *setup* através da expressão “t\_p\_1 + t\_mov\_ext\_1 + t\_setup\_1” horas.
- Módulo **Release** “Release 61” – a entidade libera uma máquina do **set** “Depto 01”, mediante o emprego da regra de liberação (**release rule**) “specific Member” e o uso dos índices (**Set Index**) “ind\_maq\_1”, ou seja, a entidade libera a máquina que foi empregada no módulo “Seize 10”.
- Módulo **Decide** “Decide 168” – este módulo testa se “eta” é igual a 1, 2 ou 3, ou seja, se a flexibilidade do operador é nula, dupla ou tripla.
- Módulo **Release** “Release 268” – a entidade libera um operador do **set** “operador\_1”, mediante o emprego da regra de liberação (**release rule**) “specific Member” e o uso dos índices (**Set Index**) “ind\_opr\_1”, ou seja, libera o operador que foi empregado no módulo “Seize 166”
- Módulo **Release** “Release 74” – a entidade libera um operador do **set** “eta\_2\_operador\_1”, mediante o emprego da regra de liberação (**release rule**) “specific Member” e o uso dos índices (**Set Index**) “ind\_opr\_1”, ou seja, a entidade libera o operador que foi empregado no módulo “Seize 112”.
- Módulo **Release** “Release 200” – a entidade libera o operador do **set** “eta\_3\_operador\_1”, mediante o emprego da regra de liberação (**release rule**) “specific Member” e o uso dos índices (**Set Index**) “ind\_opr\_1”, ou seja, a entidade libera o operador que foi empregado no módulo “Seize 146”.
- Módulos **Assign** “Assign 339” e “Rota 36”, por onde as entidades passam na seqüência, já foram descritos anteriormente.

#### A2.18 Descrição Detalhada do Bloco “Saídas de entidades”

A seguir são descritos os módulos e as configurações deste bloco.

- Módulo **Station** “Estacao Expedicao” – recebe o último estágio das seqüências de operação de todos os lotes de peças.
- Módulo **Process** “Processo Expedicao” – usado para permitir a visualização na animação e também, após a simulação, do número de entidades que saíram. O

tempo de processamento no módulo (o **Delay**) é igual a zero minutos, e portanto o mesmo não influi no tempo de fluxo das entidades.

- Módulo **Decide** “Decide 196” – testa se a expressão “ $t_{now} > 1800$ ” é verdadeira. Na prática este módulo é utilizado para permitir que um bloco seja acessado pelas entidades somente quando o tempo de *warm up* de 1800 horas for atingido. Só após este tempo os módulos “Record 1”, “Record 2” e “soma tempo” são acessados.
- Módulo **Record** “Record 1” – responsável por recolher os tempos em que cada entidade (lote) permanece na simulação (**TNOW**-início), ou seja, o tempo de fluxo (*flow Time*). O módulo **Statistic** armazena esses dados de todas as entidades em um arquivo chamado “flow\_time.dat” na área de *desktop* do sistema operacional Windows.
- Módulo **Record** “Record 2” – responsável por recolher os tempos de fila em que cada entidade (lote) permanece durante a simulação, ou seja, o atributo “Entity.WaitTime”. O módulo **Statistic** grava esses dados de todas as entidades que saem do sistema em um arquivo chamado “queue\_time.dat” na área de *desktop* do sistema operacional Windows.
- Módulo **Assign** “Soma tempos” – este módulo executa as seguintes ações:
  - atribui à variável “soma\_tempos\_fluxo” o valor da expressão “soma\_tempos\_fluxo + (tnow-início)”, ou seja, os tempos de fluxo acumulados;
  - atribui à variável “qtde\_pecas\_somadas” o valor da expressão “qtde\_pecas\_somadas+1”, ou seja, conta as peças que passaram por esse trecho do modelo;
  - atribui à variável “media\_tempo\_fluxo” o valor da expressão “soma\_tempos\_fluxo / qtde\_pecas\_somadas”, obtendo-se assim o tempo médio de fluxo depois do *warm up*.
- Módulo **Decide** “Decide 216” – testa se a expressão “(conta\_saidas < 99 && fine\_10 > 0) && 1000 < 1” é verdadeira. Uma descrição mais detalhada deste módulo será feita no capítulo 7
- Módulo **ReadWrite** “Verificação” - este módulo foi introduzido para fazer verificações que serão descritas no capítulo 6. Ele é o responsável por transferir os valores de várias variáveis para um arquivo Excel, sendo que as variáveis que o módulo coleta e transfere são: “Entity.Type”, “opc\_10”, “maq\_usada\_10”,

“t\_seiszemaq\_d10”, “t\_ini\_uso\_maq\_d10”, “t\_ini\_uso\_oper\_d10”,  
“t\_ini\_uso\_oper\_d10”, “t\_setup\_10”, “t\_p\_10”.

- Módulo **Assign** “Assign 458” – responsável por atribuir à variável “conta\_saidas” o valor da expressão “conta\_saidas+1”. Esta variável faz a contagem da quantidade de entidades que passaram por este trecho do modelo.
- Módulo **Dispose** “saida” – retira as entidades do sistema em simulação.

## APÊNDICE 3

### DETALHAMENTOS DA VERIFICAÇÃO E VALIDAÇÃO

Este apêndice tem por finalidade detalhar alguns elementos descritos no capítulo 6 – Verificação e Validação. Ele servirá como subsídio para um aprofundamento maior na compreensão dos métodos e procedimentos empregados na verificação e validação dos modelos desenvolvidos na tese.

#### A3.1 Descrição Detalhada da Verificação da Geração dos Lotes de Peças e da Demanda das Peças

Conforme abordado no subitem 6.1.1.1, descreve-se a verificação dos modelos com relação ao trecho referente à geração dos lotes de peças e da demanda das mesmas. Pretende-se verificar, desse modo, se os modelos geram - para cada uma das cinquenta peças pertencentes às cinco famílias de peças - valores de demanda anual dentro dos limites superior e inferior estabelecidos no padrão adotado e que se encontram na tabela 4.1. E depois, se as peças produzidas tendem de fato a atender essas demandas geradas. Para essa verificação foram realizadas simulações para cinco situações: três para o modelo Funcional, respectivamente com lotes de 30, 50 e 80 peças e para os modelos celular e virtual, ambos com lotes de 50 peças. A simulação foi de 3.600 horas, aproximadamente 1 ano. Usou-se o nível mais alto de variação de demanda, isto é, plena, e também do intervalo entre chegadas, ou seja, exponencial.

Conforme mencionado no capítulo 5 e no Apêndice 2, o módulo **ReadWrite** foi introduzido para fazer as verificações que são abordadas neste capítulo. Este módulo transfere os valores de várias variáveis para um arquivo Excel. Para efeito de verificação da geração dos lotes de peças e da demanda das mesmas, este módulo transfere para um arquivo Excel o valor de cem variáveis, isto é: os cinquenta valores de demanda anual gerada e armazenada nas variáveis “d\_p10”, “d\_p11”,... ”d\_p5B” descritas no subitem 5.1.1 e cinquenta variáveis que armazenam o número de lotes gerados na simulação de cada peça. Essas variáveis foram criadas automaticamente ao se introduzirem os cinquenta módulos **Create** que criam os lotes de peças durante a simulação, conforme descrito nos subitens 5.1.4 e A2.2. Essas variáveis são designadas como “Peca\_10.numberOut”, “Peca\_11.numberOut”, ... “Peca\_5B.numberOut”.

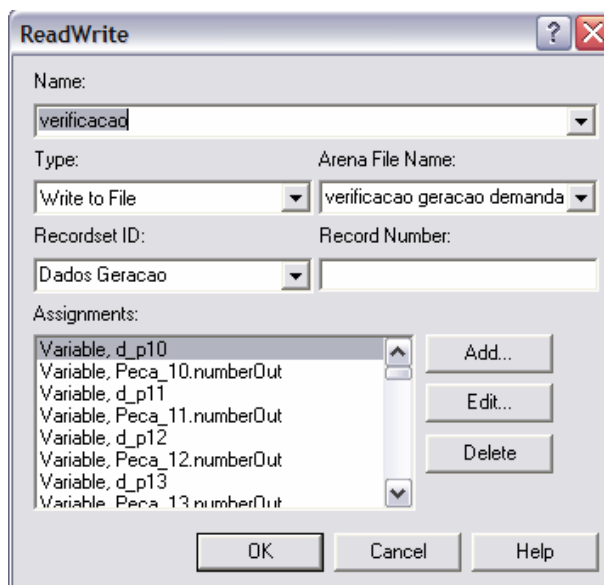


Figura A3.1 O módulo **ReadWrite**

A Figura A3.1 ilustra este módulo e as variáveis cujos valores foram transferidos para uma planilha Excel. Este módulo trabalha em conjunto com o módulo **File**, que indica a localização (disco e diretório), o nome e a guia (ou pasta), as linha e colunas da planilha Excel do software para onde os dados devem ser transferidos. A Figura A3.2 mostra este módulo e pode-se observar que, no caso concreto, o arquivo se chama “Verificação geração de peças e demanda.xls”; a guia se chama “Funcional”, e os dados devem ser transferidos para as células B3 à CW12. Por outro lado, a planilha Excel mencionada aparece parcialmente na Figura A3.3.

Foram comparados e avaliados os resultados do Gerador das Demandas e do Gerador dos Lotes de Peças nas cinco situações. Mesmo trabalhando com os valores dos modelos, que eram não determinísticos, obtiveram-se médias bem de acordo com o esperado, ou seja, dentro dos limites inferior e superior descritos na tabela 4.1 tanto para os valores gerados de demanda como para o de peças geradas que entraram no sistema. Ou seja, segundo o padrão estabelecido, entre 248 e 2.232 unidades por ano de cada um dos doze tipos de peça da família um e dos doze da família cinco. Entre 310 e 2.790 unidades por ano de cada um dos dez tipos de peça da família dois. E finalmente, entre 395 e 3.555 unidades por ano de cada um dos oito tipos de peça da família três e dos oito da família quatro, totalizando assim cinquenta distintas demandas de peça geradas no início de cada uma das replicações realizadas por altura de cada simulação de cada um dos modelos desenvolvidos.

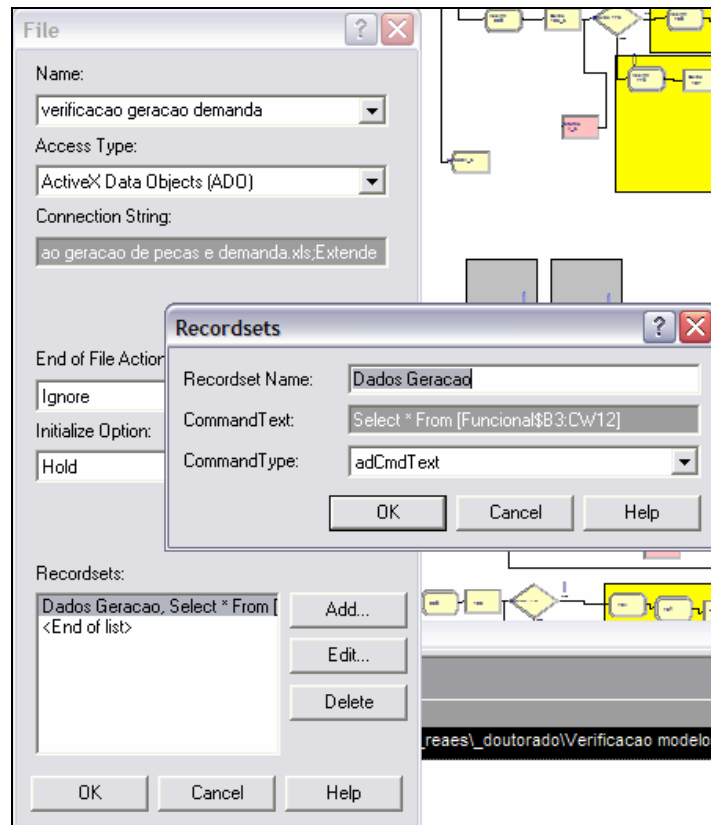


Figura A3.2 O módulo File

	A	B	C	D	E	F	G	
1	Warm-up 1800 h		10		11		12	
2	faz real=demanda/2	Demanda (peças)	Real (lotes 50 p.)	Demanda (peças)	Real (lotes 50 p.)	Demanda (peças)	Real (lotes 50 p.)	Dem
3	1	804,68	6,00	1671,76	16,00	2113,56	15,00	
4	2	884,77	10,00	780,62	13,00	1100,13	9,00	
5	3	2165,09	24,00	268,27	2,00	418,44	3,00	
6	4	826,82	6,00	842,21	3,00	756,82	7,00	
7	5	727,12	4,00	491,81	3,00	435,75	4,00	
8	6	1175,07	16,00	738,42	6,00	1881,08	21,00	
9	7	798,33	12,00	1697,14	22,00	1356,14	16,00	
10	8	919,98	11,00	2180,88	30,00	1709,38	15,00	
11	9	1202,47	13,00	346,74	3,00	1962,82	26,00	
12	10	1847,53	16,00	1235,28	11,00	1064,23	14,00	
13	média	1135,18		1025,31		1279,83		
14			1180,00		1090,00		1300,00	

Figura A3.3 A planilha Excel com os dados de demanda e geração de lotes

Todos os valores de demanda gerada ficaram dentro dos limites superiores e inferiores de valores de demanda especificados. A coluna B da planilha apresentada na Figura A3.3 contém as demandas anuais da peça dez, que pertence à família um, geradas aleatoriamente no início das dez replicações da simulação. O limite superior e inferior de



demanda anual dessa peça é respectivamente 248 e 2.232. Pode-se comprovar que todos os valores estão dentro desse intervalo.

Entretanto, uma pequena parte dos valores de peças criadas ultrapassa os limites inferior e superior (aprox. entre 5 e 15% do total de peças). Quanto menor o tamanho do lote, menor o percentual de peças que ultrapassam os limites de demanda adotados. A explicação para esse fato consiste em saber que as peças são criadas com intervalos aleatórios. E parte do cálculo desses intervalos são as demandas anuais, que são também aleatórias. Assim, quando a demanda anual gerada é muito próxima dos limites estabelecidos, pode ocorrer que a aleatoriedade dos intervalos acabe tendo um caráter acumulativo, jogando esses valores um pouco abaixo ou acima dos intervalos inicialmente previstos.

Para adequar esses valores ao padrão adotado, pensou-se em substituir a função exponencial por outra que gerasse menos variabilidade, por exemplo, a uniforme, que garantiria que todos os valores ficariam dentro do intervalo. Outra possibilidade seria tornar os limites de demanda mais amplos que o padrão original, ou seja, um novo padrão, ou usar um fator de redução, por exemplo, com auxílio de uma curva de Gauss. Mas optou-se por manter a função exponencial, pois seria melhor ter mais variabilidade na geração de peças do que menos, uma vez que, neste segundo caso, seria uma mitigação das variações normais a que todos os sistemas reais estão sujeitos.

Fora esta pequena variação, pode-se verificar que a função geradora de demanda e de entrada de peças no sistema atende plenamente o especificado, ou seja, valores dentro dos limites definidos da tabela 4.1. A Figura A3.3 mostra, nas colunas C, E e G da planilha, os valores transferidos das quantidades de peças criadas na simulação. Em função do período transiente de **Warm up**, de seis meses, os resultados obtidos devem ser multiplicados por dois. As quantidades produzidas referem-se ao número de lotes criados apenas nos últimos seis meses de cada simulação. Como a quantidade de peças nos lotes do exemplo era cinquenta, os limites inferiores e superiores de quantidades de lotes desejados eram 2,48 e 22,32 para as peças da família um (respectivamente  $(248/50)/2$  e  $(2.232/50)/2$  lotes - esses valores estão divididos por dois, pois os numeradores são os limites para demanda anual e não semestral). Dos trinta resultados mostrados na figura, apenas três apresentam pequenas variações para cima ou para baixo, que, conforma abordado anteriormente, se deve ao caráter acumulativo do cálculo aleatório da produção anual de lotes de peças. Assim, por exemplo, a produção de quantidades anuais de lotes da peça 10 em cada simulação é gerada a partir da função aleatória exponencial  $EXPO((q/d_{p10}) * 3600)$  - onde  $q$  é o tamanho do lote - que por sua vez tem como parâmetro o valor da demanda anual também gerada aleatoriamente por uma função aleatória uniforme, isto é,

$d_{p10} = \text{UNIF}(248, 2232)$ , conforme abordado nos capítulos 4 e 5). Esse cálculo eventualmente pode gerar valores um pouco abaixo ou acima dos intervalos inicialmente previstos, caso os valores aleatoriamente gerados da demanda fiquem muito próximos dos limites de intervalo.

### A3.2 Descrição Detalhada do roteiro de fabricação, dos tempos de movimentação, setup etc.

Conforme abordado no subitem 6.1.1.2, Foram feitas seis simulações para a verificação dos modelos com relação ao correto funcionamento do seqüenciamento dos lotes de peças nos departamentos ou nas máquinas das células especificadas. O objetivo foi verificar se não havia nenhum erro de lógica nos três modelos desenvolvidos na tese quanto a modelagem com o software Arena. Assim, com a técnica *trace*, verificou-se se as entidades percorriam a seqüência certa de máquinas e operações previstas no padrão adotado e definido na metodologia. Foram verificados também os tempos médios de movimentação dos operadores, *setup*, processamento e movimentação das peças (ou rota). Mais uma vez, também via *trace*, procurou-se apurar se a modelagem com auxílio do software, não apresentava erros lógico e os valores reais de tempo de processamento, de movimentação, de preparação das peças durante a simulação estavam estritamente dentro do modelo adotado. Para tanto, foram simulados individualmente lotes para as peças 12 e 20 em todos os três tipos de configuração: *job shop*, celular tradicional e virtual. A figura A3.4 mostra as variáveis coletadas para o Excel.

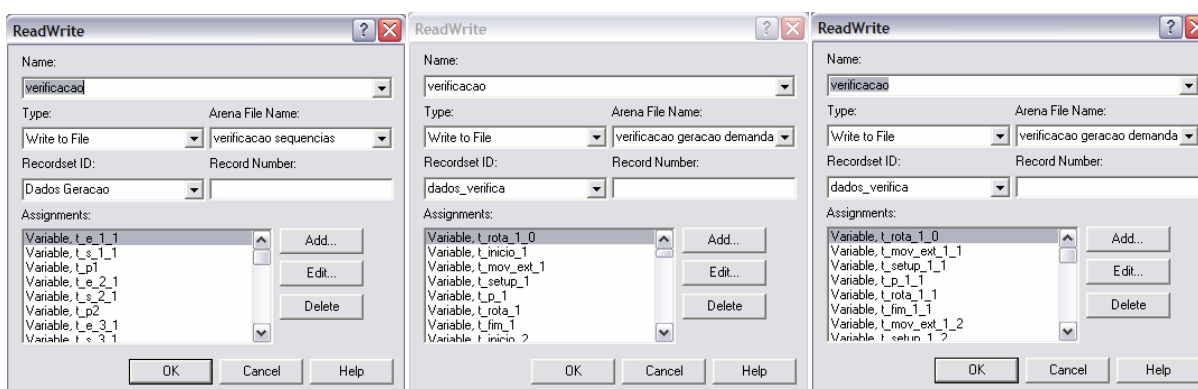


Figura A3.4 Telas da coleta de dados do *job shop*, do celular virtual e do celular tradicional

Os dados coletados no *job shop* foram “t\_e\_1\_1” (tempo de entrada no departamento 1), “t\_s\_1\_1” (tempo de saída do departamento 1) e “t\_p1” (Tempo de processamento em máquina do departamento 1). No celular virtual os dados foram “t\_rota\_1\_0” (tempo de rota ou movimentação para a próxima operação ou para a saída), “t\_inicio\_1” (tempo de chegada

no departamento 1), “ $t_{mov\_ext\_1}$ ” (tempo de movimentação de um operador externo para atender uma máquina do departamento 1) e “ $t_{setup\_1}$ ” (tempo de *setup* na máquina do departamento 1), “ $t_p_1$ ” (tempo de processamento na máquina do departamento 1). E no celular tradicional foram: “ $t_{rota\_1\_0}$ ” (tempo de rota ou movimentação para a próxima operação ou para a saída), “ $t_{mov\_ext\_1\_1}$ ” (tempo de movimentação de um operador interno para atender uma máquina da célula 1), “ $t_{setup\_1\_1}$ ” (tempo de *setup* da máquina 1 na célula 1), “ $t_p_1_1$ ” (tempo de processamento da máquina 1 da célula 1). Com os tempos de entrada ou início foi possível verificar a seqüência de produção.

A figura A3.5 mostra os dados do *job shop* transferidos para o Excel. Trata-se dos dados obtidos em 10 replicações. Pelos dados da planilha, pôde-se constatar que todas as peças seguiam exatamente o seqüenciamento estabelecido (o lote da peça 12, por exemplo, apesar da figura não permitir uma visão completa, permite observar que passou pelos departamentos: 1, 2, 4, 5, 6 e 8, nesta ordem, porque os valores dos tempos de entrada “ $t_{e\_1\_1}$ ”, “ $t_{e\_2\_1}$ ”, “ $t_{e\_4\_1}$ ”, “ $t_{e\_5\_1}$ ” etc. são não nulos e crescente, ou seja, passaram por esses departamentos e nesta ordem). Concretamente, na primeira replicação, conforme pode-se parcialmente comprovar por essa última figura, os tempos de entradas para a única entidade no *job shop* foram: “ $t_{e\_1\_1} = 0,17$ ”, “ $t_{e\_2\_1} = 17,76$ ”, “ $t_{e\_3\_1} = 0$ ”, “ $t_{e\_4\_1} = 23,36$ ”, “ $t_{e\_5\_1} = 36,28$ ”, “ $t_{e\_6\_1} = 49,59$ ”, “ $t_{e\_7\_1} = 0$ ”, “ $t_{e\_8\_1} = 53,00$ ”, “ $t_{e\_9\_1} = 0$ ” e “ $t_{e\_10\_1} = 0$ ”. Ou seja, de fato o roteiro faz com que a entidade passe pelos departamentos: 1, 2, 4, 5, 6 e 8. Os demais departamentos não receberam valor porque a entidade não passou por eles. E a ordem está confirmada, pois os tempos de entrada nesses departamentos, nessa ordem, são crescentes, isto é, 0,17, 17,76, 23,36 horas etc. A mesma verificação se dá com o celular virtual e com o celular tradicional.

Nesse mesmo conjunto de simulações, pode-se verificar também que os tempos médios de movimentação dos operadores, *setup*, processamento e movimentação das peças (ou rota) estão dentro do esperado, conforme os padrões adotados no capítulo 4. Por exemplo, a tabela 4.6 contém os tempos de movimentação de lotes e operadores e a tabela 4.5 os de *setup*. Assim, por exemplo, para o sistema *job shop*, que não tem redução de *setup*, esperavam-se os valores aleatórios da função  $\text{expo}(3)$ , com média 3. Para os sistemas celular tradicional e celular virtual, esperavam-se valores aleatórios dados pela expressão  $\text{expo}(3) * \delta$ , onde  $\delta$  é o fator de redução do *setup* que admite dois níveis, isto é, 0,60 e 0,25. Alguns desses dados estavam disponíveis na planilha e outros foram obtidos trabalhando com os dados disponíveis. Assim, por exemplo, no caso do *job shop*, os tempos de processamento foram diretamente transferidos para a planilha.

Microsoft Excel - sequencias.xls

Arquivo Editar Exibir Inserir Formatar Ferramentas Dados Janela Documents To Go Ajuda

Digite uma pergunta

D30  $\text{fx} = +\text{MÉDIA}(\text{AG16}:\text{AG26})$

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1		1			2			3			4			5		
2		T_e_l1	T_s_l1	T_p_l1	Temp_est_2_1	Temp_saida_2_1	Temp_proc_2_1	Temp_est_3_1	Temp_saida_3_1	Temp_proc_3_1	Temp_est_4_1	Temp_saida_4_1	Temp_proc_4_1	Temp_est_5_1	Temp_saida_5_1	Temp_proc_5_1
3	1	0,17	16,99	14,09	17,76	22,54	3,61	0,00	0,00	0,00	23,36	35,31	5,06	36,28	49,08	
4	2	0,17	3,13	2,81	3,99	16,47	4,48	0,00	0,00	0,00	17,20	22,33	4,95	23,03	33,74	
5	3	0,17	2,02	0,45	2,68	9,27	6,32	0,00	0,00	0,00	9,88	16,13	4,11	16,64	22,69	
6	4	0,17	8,08	1,48	8,94	14,11	4,62	0,00	0,00	0,00	14,84	30,41	15,54	30,98	39,05	
7	5	0,17	7,45	0,50	8,13	13,61	4,24	0,00	0,00	0,00	14,40	33,85	19,23	34,68	41,99	
8	6	0,17	17,97	8,66	18,67	24,92	5,99	0,00	0,00	0,00	25,49	37,20	4,97	38,19	48,87	
9	7	0,17	7,75	4,09	8,41	31,58	9,39	0,00	0,00	0,00	32,37	41,12	1,78	41,93	44,44	
10	8	0,17	8,21	6,67	8,74	24,80	12,91	0,00	0,00	0,00	25,69	30,45	2,45	31,29	43,13	
11	9	0,17	10,52	9,99	11,09	14,59	3,34	0,00	0,00	0,00	15,53	24,71	4,85	25,53	31,93	
12	10	0,17	3,90	2,65	4,43	18,51	13,69	0,00	0,00	0,00	19,07	20,08	0,30	20,98	25,71	
13																
14		1			2			3			4			5		
15		Moviment	setup	Processam.	Moviment	setup	Processam.				Moviment	setup	Processam.	Moviment	setup	Proc
16	1	0,17	2,89	14,09	0,78	1,17	3,61				0,82	6,89	5,06	0,97	7,91	4
17	2	0,17	0,32	2,81	0,86	8,00	4,48				0,73	0,18	4,95	0,70	4,79	5
18	3	0,17	1,57	0,45	0,66	0,26	6,32				0,61	2,14	4,11	0,51	4,97	1,
19	4	0,17	6,59	1,48	0,86	0,55	4,62				0,73	0,03	15,54	0,57	4,86	3
20	5	0,17	6,95	0,50	0,69	1,23	4,24				0,79	0,22	19,23	0,84	7,19	0
21	6	0,17	9,31	8,66	0,69	0,25	5,99				0,58	6,75	4,97	0,98	3,20	7
22	7	0,17	3,66	4,09	0,66	13,78	9,39				0,79	6,97	1,78	0,80	1,08	1,
23	8	0,17	1,54	6,67	0,53	3,16	12,91				0,89	2,32	2,45	0,84	0,36	11
24	9	0,17	0,53	9,99	0,57	0,16	3,34				0,95	4,33	4,85	0,81	2,47	3
25	10	0,17	1,25	2,65	0,54	0,39	13,69				0,56	0,72	0,30	0,89	1,07	3
26	Média	0,17	3,46	5,14	0,68	2,89	6,86				0,74	3,05	6,32	0,79	3,79	4
27																
28		Média das médias														
29		T_mov_ext_1a	T_setup_1a	T_p_1a	T_rota_1a											
30		0,000	3,16	5,84	0,75											

Funcional peça 12 / Funcional peça 20 / Celular peça 12 / Celular peça 10 / Celular todas peças / Virtual peça 12

Figura A3.5 Dados do job shop transferidos para o Excel

Entretanto, como podemos verificar na planilha na primeira replicação no departamento 2, o tempo de *setup* de 1,17 h foi obtido pela subtração do tempo de saída (“t\_s\_2\_1” que era igual a 22,54 h) pelos tempos de entrada (“t\_e\_2\_1” que era igual a 17,76 h) e de processamento (“t\_p2” que era igual a 3,61 h). Neste mesmo exemplo, o tempo de movimentação (ou rota) da peça era obtido pela subtração do tempo de entrada no departamento (“t\_e\_2\_1” que era igual a 17,76 h) pelo tempo de saída do departamento anterior (“t\_s\_1\_1” que era igual a 16,99 h).

Assim, o tempo médio de movimentação dos lotes da peça 12 na configuração *job shop* (“T\_rota\_1\_a”) foi de 45 minutos, correspondendo ao esperado, uma vez que era a média do valor da função uniforme (30, 60) minutos, conforme a tabela 4.6. Para os lotes do mesmo tipo de peça, o tempo médio de *setup* (“T\_setup\_1\_a”) obtido foi 3,16, ótimo para a função empregada para representá-lo:  $\text{expo}(3)$ . Já o tempo médio de processo (“T\_p\_1\_a”) obtido, 5,84, está ótimo para a função  $\text{expo}(0.1*q)$ , onde  $q$  é o tamanho de lote, que nestas replicações assumiu o valor de cinquenta unidades. Foram feitas simulações também com números maiores de quantidades de lotes e foram obtidos valores sempre próximos de cinco.

Através da figura A3.6, pode-se verificar o valor do *setup* reduzido, no caso da configuração celular tradicional. O tempo médio de *setup*, obtido neste caso para os lotes da peça 12, foi 0,898 horas (célula B17 do Excel da figura 6.6), o que representa um resultado coerente, tendo em conta que é obtido pela expressão “ $\text{setup\_reduz} * \text{expo}(3)$ ” horas e tenha sido simulado com o valor de “ $\text{setup\_reduz}$ ” de 0,25, pois  $0,25 * 3$  (se o valor gerado pela função aleatória  $\text{expo}(3)$  desse exatamente o valor médio) é igual a 0,75 horas, especialmente se leva-se em conta o emprego da função exponencial, que introduz uma elevada variabilidade dos valores. Neste caso de configuração celular tradicional, o tempo médio de movimentação das peças obtido, 0,075 h (célula D17 do Excel da figura 6.6), é muito bom já que o padrão é a função aleatória “ $\text{unif}(0,05, 0,1)$ ” horas (ou conforme a tabela 4.6, Unif (3, 6) minutos, que é o mesmo), então o valor esperado é exatamente 0,075 h.

	A	B	C	D	E	F	G	H	I	J
2		T_rota_1_0	T_mov_ext_1_1	T_setup_1_1	T_p_1_1	T_rota_1_1	t_fim_1_1	T_mov_ext_1_2	T_setup_1_2	T_p_1_2
3	1	0,167	0,000	0,541	4,545	0,066	5,089	0,000	1,724	5,21
4	2	0,167	0,000	0,673	0,258	0,097	0,933	0,000	0,044	3,00
5	3	0,167	0,000	0,949	2,342	0,054	3,294	0,000	0,536	1,25
6	4	0,167	0,000	0,693	10,714	0,058	11,410	0,000	0,007	3,08
7	5	0,167	0,000	0,637	11,303	0,067	11,942	0,000	0,054	4,35
8	6	0,167	0,000	0,899	15,242	0,054	16,144	0,000	1,687	0,81
9	7	0,167	0,000	1,408	5,826	0,099	7,237	0,000	1,742	4,32
10	8	0,167	0,000	1,936	2,288	0,083	4,227	0,000	0,579	7,35
11	9	0,167	0,000	0,501	0,610	0,053	1,114	0,000	1,082	11,0
12	10	0,167	0,000	2,054	1,800	0,056	3,857	0,000	0,179	0,55
13	média	0,167	0,000	1,029	5,493	0,069	6,525	0,000	0,763	4,10
14										
15		Média das médias								
16		T_mov_ext_1_n	T_setup_1_n	T_p_1_n	T_rota_1_n					
17		0,018	0,898	4,59	0,075					

Figura A3.6 Dados do celular tradicional transferidos para o Excel

A figura A3.7 apresenta um resumo de alguns dos dados analisados. Pode-se verificar que a configuração celular tradicional e celular virtual tem redução de *setup*. Apesar de estar operando com um só lote durante o teste de verificação, a configuração celular acaba tendo algum tempo de movimentação de operadores, pois tem menos operadores do que tipos distintos de máquinas por célula. Esse é outro resultado esperado e verificado. Os tempos de movimentação de lotes de peças também estão coerentes, uma vez que o *job shop* e o virtual apresentam tempos maiores do que o celular tradicional, que só tem movimentação interna.

	A	B	C	D	E	F	G	
1		Média das médias						
2								
3		T_mov_ext_1_n	T_setup_1_n	T_p_1_n	T_rota_1_n			
4	Funcional	0,00	3,16	5,84	0,75	Uniform(30, 60) min		
5	Celular	0,02	0,90	4,59	0,07	Uniform(3, 6) min		
6	Virtual	0,00	0,91	3,69	0,74	Uniform(30, 60) min		
7								

Figura A3.7 Resumo dos três sistemas



de valores médios, assim, a expressão fica “ $10/60 + (0,1 * 50 + 3 + 0,75) * 3$ ”, ou seja, 10 minutos de rota (conforme definido no subitem 4.1.3), acrescidos de  $0,1 * 50$  h de tempo médio de processamento (definido no subitem 4.1.1 como sendo dado pela função exponencial  $Expo(0.1*q)$  horas, onde  $q$  é o tamanho de lote, no caso igual a 50), mais 3 horas de tempo médio de *setup* (conforme tabela 4.5, para  $\delta=1$ , já que o *job shop* não tem redução de *setup*) e mais 0,75 horas de tempo médio de movimentação (da tabela 4.6), multiplicados esses 3 últimos números pelo número de estágios (na tabela 4.2 percebe-se que P23 e P27 têm 3 estágios), gerando finalmente o valor de 26,42 h (que está na tabela A3.2).

Tabela A3.2 – Tempos médios nos três sistemas por n° de estágios

n° estágios	Funcional		Celular tradicional		Virtual	
	real (horas)	estimado (horas)	real (horas)	estimado (horas)	real (horas)	estimado (horas)
3	26,45	26,42	17,80	17,71	19,6	19,7
4	34,64	35,17	23,61	23,56	25,3	26,2
5	43,89	43,92	29,45	29,41	32,1	32,7
6	51,95	52,67	35,33	35,26	38,8	39,2
7	61,00	61,42	40,65	41,11	46,1	45,7

As simulações realizadas, tanto com tempos determinísticos como com tempos não determinísticos, obtiveram resultados médios de acordo com o esperado, conforme descrito no parágrafo anterior, não parecendo haver qualquer erro no modelo, neste aspecto em particular. Levando-se em conta que os 3 modelos testados são não determinísticos, os resultados corroboram a adequação do modelo. Os determinísticos deram resultados exatos.

A tabela A3.2 mostra os resultados finais com uso de valores não determinísticos. Os resultados, em si mesmos, também mostraram-se consistentes, pois, tratando-se de lotes iguais de um só tipo de peça, os modelos celular tradicional e virtual levam vantagem significativa sobre o *job shop*, uma vez que se beneficiam do redutor de *setup*. Por outro lado, o celular tradicional leva uma pequena vantagem sobre o virtual, pois os tempos de movimentação dos lotes (ou rota) são bem menores. O celular tradicional gasta um tempo extra, ínfimo, com relação aos demais modelos, devido à necessidade de movimentação interna de operadores que, por ser inexpressiva, acaba não sendo significativa diante do tempo empregado nas demais operações.

Nesta seqüência de testes, foi realizada também a simulação de um lote de peças único no sistema virtual para verificar o tempo total com *setup* completo. Os resultados



foram exatos com valores determinísticos, verificando-se o modelo com relação a esta característica do celular virtual. Estes testes possibilitaram verificar plenamente os três modelos no que se refere aos tempos de operação, para o caso simples de produção de uma grande gama de lotes de um único tipo de peça.

Para os intervalos de tempo entre as chegadas de lotes de peças, foram empregados, no modelo *job shop*, valores exponenciais obtidos através da função  $EXPO(q/d_{pXX}) * 3600$ . Para os outros dois tipos de configuração, foram empregados valores não exponenciais, obtidos pela função  $(q/d_{pXX}) * 3600$ , onde  $q$  é o tamanho do lote (no caso testado, lote com 50 unidades) e  $d_{pXX}$  a demanda da peça  $XX$  (por exemplo, a demanda da peça 10, sendo todas as demandas de peças determinadas aleatoriamente, no início da simulação).

#### **A3.4 Descrição Detalhada da Verificação da flexibilidade dos operadores**

Conforme abordado na descrição do modelo conceitual os modelos desenvolvidos seguem o padrão de modelos sujeitos a uma dupla restrição de recursos (DRC – *Dual-Resource-Constrained*), de máquinas e operadores. Os modelos contemplam operadores sujeitos a três níveis de flexibilidade,  $\eta = 1, 2$  ou  $3$ , ou seja, com habilidade para operarem uma, duas ou três máquinas. Este item aborda a verificação realizada com relação a esta flexibilidade, que constitui um elemento essencial desse padrão DRC. Através da técnica *trace*, se procura verificar se os operadores utilizam a sua flexibilidade nos seus três níveis e respeitando as regras estabelecidas na descrição do modelo conceitual no capítulo 4. Com essa finalidade foram realizadas diversas simulações com o objetivo de verificar se nos 3 modelos (*job shop*, celular tradicional e virtual) a flexibilidade dos operadores (ou *cross training*) estava correta, ou seja, conforme o estabelecido no subitem 5.1.2 e na tabela 4.7. Nesse subitem e tabela são descritos como os operadores são agrupados em **sets** ou conjuntos de operadores capazes de operar determinadas máquinas e como esses **sets** são constituídos de forma diferente em função do nível de flexibilidade escolhido ( $\eta = 1, 2$  ou  $3$ ). Assim, por exemplo, para o modelo *job shop*, o **set** de operadores que pode operar as máquinas do departamento 1 para  $\eta = 1$  (ou seja, habilidade para operar apenas uma máquina), designado como “operador\_1”, é composto pelos operadores operador\_a, operador\_b e operador\_c. Para  $\eta = 2$ , o set apto a realizar a mesma tarefa denomina-se “eta\_2\_operador\_1” e é constituído pelos operadores operador\_a, operador\_b, operador\_c, operador\_i, operador\_k, operador\_m e operador\_t. Por último para  $\eta = 3$ , o set denomina-se “eta\_3\_operador\_1” e se constitui dos mesmos operadores de “eta\_2\_operador\_1”. Os dados das diversas simulações realizadas para esta verificação foram transferidos para a planilha Excel “verificação cross training.xls” para análise.

#### A3.4.1 Descrição detalhada da verificação da flexibilidade dos operadores no job shop

Para verificar a configuração *job shop*, foi realizada uma simulação de 500 horas, com “eta=3”, lote=50 peças, com uma única replicação. Os resultados da simulação foram transferidos para a pasta com guia denominada de “funcional passo a passo”, da planilha Excel, correspondentes aos primeiros 99 lotes que saíram da simulação (em função da restrição de 100 transferências no máximo por simulação). A tabela A3.3, aqui representada parcialmente, apresenta os resultados dos dois primeiros lotes. Os dados contêm informações relativas a quais operadores foram utilizados, em que departamentos foram empregados e aos tempos de início e fim dessa utilização. Nos departamentos em que os lotes não passaram, os dados correspondentes são nulos. Foram registrados os valores “ind\_opr\_1”, “ind\_opr\_2”,... “ind\_opr\_10”, que são os índices que identificam os recursos (os operadores) utilizados nos sets “eta\_3\_operador\_1”, “eta\_3\_operador\_2” etc. (que são o conjunto de operadores habilitados para operar respectivamente as máquinas dos departamentos 1, 2 etc.) com os correspondentes tempos de início e fim da operação de cada um desses operadores, nos diversos processos em que cada lote passou. Assim, o primeiro lote, que concluiu todas as operações previstas, passou pelo departamento 4, usando o operador índice 2 do conjunto de operadores habilitados para trabalhar neste departamento, ou seja, o operador “i”, iniciando a operação às 16,7 h (do início da simulação) e terminou às 20,4 h. Também passou pelos departamentos 6, 7 e 9, onde também foram registrados os operadores utilizados e seus respectivos tempos de início e fim de operação. O mesmo foi feito para os 98 subseqüentes lotes que concluíram suas operações.

Tabela A3.3 – Operadores e tipos de máquinas utilizadas no *job shop*

ord	Depto 1			Depto 2			Depto 3			Depto 4			Depto 5			Depto 6			Depto 7			Depto 8			Depto 9			Depto 10		
	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim	oper	inic	fim
1	0	0	0	0	0	0	0	0	0	2	17	20	0	0	0	2	15	16	3	12,7	14,4	0	0	0	2	6,91	11,8	0	0	0
2	0	0	0	0	0	0	0	0	0	5	22	29	0	0	0	1	16	21	1	11,6	15,6	0	0	0	1	6,58	10,6	0	0	0

A partir dos dados da guia “funcional passo a passo”, descritos anteriormente e utilizados parcialmente na tabela A3.3, foi possível construir a tabela A3.4, aqui representada também parcialmente, que apresenta, a cada linha, em ordem cronológica, passo a passo, cada operação que envolveu um operador. Assim, a primeira linha indica que um lote (de número de ordem 8), após dar um **seize** em uma máquina do departamento 8 (coluna “maq” na tabela), segundo a sua seqüência de produção, dá um **seize** às 0,99 h no **set** “eta\_3\_operador\_8”, que é o conjunto de operadores que operam a máquinas do departamento 8 para eta=3. Obtém o primeiro operador deste set (#op=1), ou seja, o

“operador\_p” (oper=p na tabela), que é um operador interno desse departamento. Este operador é liberado às 3,19 h.

Tabela A3.4 – Operadores e tipos de máquinas utilizadas no *job shop*

Ord	maq	#op	oper	inic	Fim
8	8	1	p	0,99	3,19
8	10	1	t	3,99	7,41
11	1	1	a	4,53	19,9
2	9	1	r	6,58	10,6
9	1	2	b	6,61	13,7
1	9	2	s	6,91	11,8
20	2	1	d	7,2	31,6
8	1	3	c	8,14	25,3
5	10	1	t	8,25	12,4
35	8	1	p	8,94	25,3
13	4	1	h	9,23	13,9
15	1	4	m	10,4	14,4
2	7	1	n	11,6	15,6
6	8	2	q	11,8	15,5
4	10	2	u	12	22,2
7	7	2	o	12,3	15,5
1	7	3	f	12,7	14,4
...	.....	.....	.....	.....	.....

A partir da tabela A3.4, foi desenvolvido o gráfico da Figura A3.8, que permite visualizar os recursos empregados nos diversos departamentos, no início da simulação, ou seja, do tempo igual a zero até a verificação da primeira ocorrência de uma flexibilidade dos operadores (ou *cross training*), que se dá no tempo 10,4 horas (a partir do início da simulação).

Este gráfico permite verificar também que os operadores internos (a, b e c) estavam ocupados operando três máquinas do departamento 1, quando surge um novo lote necessitando ser operado por uma máquina deste departamento. Por isso, foi escolhido o “operador\_m”, uma vez que é o primeiro na ordem da flexibilidade dos operadores externos (*cross training*) habilitados para operar uma máquina do departamento 1, para “eta==3” (m, t, k, i), e estava livre. Por ter atingido já o objetivo da verificação, parou-se a representação do gráfico da Figura A3.8 às 10,4 h, mas pode-se verificar também, mediante a tabela A3.3, que a próxima flexibilidade dos operadores (*cross training*) ocorre às 12,7 h de simulação para atender a terceira máquina do departamento 7, quando os seus dois operadores internos (n, o) estão ocupados. Assim, o operador “f” é utilizado, porque é a primeira opção de operador externo habilitado para operar máquinas do departamento 7 para “eta==3”, e está livre.

	Operadores																				
tempo	a	b	c	d	E	f	g	h	i	j	k	l	m	n	o	p	Q	r	s	t	u
0,99																8					
3,19																					
3,99																				10	
4,53	1																			10	
6,58	1																	9		10	
6,61	1	1																9		10	
6,91	1	1																9	9	10	
7,2	1	1		2														9	9	10	
7,41	1	1		2														9	9		
8,14	1	1	1	2														9	9		
8,25	1	1	1	2														9	9	10	
8,94	1	1	1	2												8		9	9	10	
9,23	1	1	1	2				4								8		9	9	10	
10,4	1	1	1	2				4				1				8		9	9	10	

Figura A3.8 Gráfico com as ocupações dos operadores

Pode-se observar ainda que a ordem de utilização dos operadores internos também está correta, fato que é particularmente simples de comprovar, no departamento 1 (composto por quatro máquinas). Primeiro, é acionado o operador a, depois o b e só depois o c. Exatamente a ordem prevista. O mesmo se pode concluir do departamento 9 e até mesmo dos departamentos 2, 4, 8 e 10, que, apesar de empregarem só um operador neste trecho reduzido de simulação, utilizam o primeiro operador na ordem estabelecida.

Todos esses resultados descritos podem ser acompanhados e visualizados através da animação do software Arena, na execução passo a passo (*step by step*). Desse modo, mais uma vez se verifica que o processo da flexibilidade dos operadores (*cross training*) está ocorrendo exatamente conforme o padrão adotado para o modelo *job shop* utilizado nesta tese.

#### A3.4.2 Descrição detalhada da verificação da flexibilidade dos operadores (*cross training*) no celular tradicional

Esta verificação realiza conferências, via trace, idênticas as realizadas no subitem anterior, agora para o modelo celular tradicional. Nesse modelo, uma vez que não há possibilidade de emprego de operadores externos, o **set** de operadores refere-se apenas aos operadores internos que podem operar uma determinada máquina de uma célula. Assim, o set “opr\_d1\_m1” que é apto a operar a máquina do tipo 1 da célula 1 para eta = 2 (ou seja, habilidade para operar duas máquinas – o modelo de celular tradicional adotado não é viável para eta=1, uma vez que há menos operadores do que máquinas), é composto

apenas pelo operador `operador_a`. Para  $\eta = 3$ , o set apto a realizar a mesma tarefa denomina-se “`ex_3_opr_d1_m1`” e é constituído também apenas pelo operador `operador_a`.

Para testar o *cross training* dos operadores na configuração celular, foram realizadas dez simulações com lotes de 50 peças e com uma única replicação. Em cada simulação, foram obtidos os resultados para apenas uma das 5 células (“depto 1”, “depto 2”, “depto 3”, “depto 4” ou “depto 5”) com “ $\eta = 2$ ” ou “ $\eta = 3$ ”, redundando, dessas dez combinações, os dez experimentos ou simulações.

Dos resultados dessas dez simulações, escolheu-se uma, em que se fez uma análise mais aprofundada, para verificar com detalhe o mecanismo do *cross training*. Optou-se, aleatoriamente, testar o *cross training* dos operadores que atendem as máquinas da célula “depto\_1” para “ $\eta = 3$ ”. As máquinas dessa célula são “Depto 01\_1”, “Depto 01\_2”, “Depto 01\_4”, “Depto 01\_5”, “Depto 01\_6” e “Depto 01\_8”. A quantidade máxima de transferências realizadas pelo módulo **ReadWrite** é 100. Por isso, o tempo de simulação empregado foi de 175 horas com uma só replicação (ainda assim apenas para coletar os dados da célula “depto\_1”) de forma a não ultrapassar este limite de transferência de dados. Cada conjunto de dados é coletado e transferido para a planilha Excel (gerando uma linha de dados) imediatamente após um operador ter sido liberado por uma entidade que estava sendo processada na célula estudada, no caso concreto aqui abordado, na célula “depto\_1”. Dessa forma, a ordem dos dados na tabela é cronológica, ordenada pela hora em que os operadores são liberados, ou seja, na tabela, os dados estão ordenados em função dos valores da variável “tempo\_fim”.

A tabela A3.5 exibe uma parte dos resultados obtidos nesta simulação (as 57 primeiras horas).

Na configuração *job shop*, máquinas idênticas são agrupadas na mesma área. Todos os operadores internos operam qualquer uma delas como sua primeira (ou eventualmente única) habilidade. O recurso da flexibilidade dos operadores (*cross training*) só é empregado quando todos esses operadores internos estão ocupados; nesse caso, buscam-se outros externos que também estão habilitados a operar estas máquinas. Nas células tradicionais, entretanto, as máquinas são agrupadas para atender todas as operações de uma família de peças. Assim, estão agrupadas máquinas de processos distintos. Em consequência, e particularmente no cenário escolhido, apenas um, ou mesmo nenhum, operador pode operar cada uma das máquinas das células como primeira habilidade, o que torna necessário frequentemente buscar um operador interno que tenha capacidade de operar esta máquina como flexibilidade do operador (*cross training*), ou seja, como uma segunda habilidade.

Tabela A3.5 Operadores e máquinas utilizadas no celular tradicional por ordem de saída

Lote	maquina	Tempo_inicio	indice_set	tempo_fim	operador
1	depto 01_1	4,533942342	1	6,75545711	a
2	depto 01_1	6,75545711	1	9,081168431	a
3	depto 01_1	10,37710888	1	12,69091183	a
4	depto 01_4	6,911202766	1	12,96279346	h
5	depto 01_6	13,14677166	1	14,28591298	l
6	depto 01_8	14,39532489	2	14,81675109	l
7	depto 01_4	12,96279346	1	16,80169847	h
8	depto 01_4	16,80169847	1	20,20829684	h
9	depto 01_5	16,95887978	3	22,09640406	d
10	depto 01_5	22,09640406	3	25,42534176	d
11	depto 01_1	12,69091183	1	26,52354097	a
12	depto 01_1	26,52354097	1	28,35339497	a
13	depto 01_2	26,67115545	1	28,43000219	d
14	depto 01_4	20,20829684	1	29,50683837	h
15	depto 01_6	22,24592589	1	30,88933329	l
16	depto 01_4	29,50683837	1	32,75944294	h
17	depto 01_1	28,35339497	1	34,69598031	a
18	depto 01_6	30,88933329	1	35,78607005	l
19	depto 01_8	<b>31,0186581</b>	2	39,1340083	l
20	depto 01_2	28,4962689	1	40,64854384	d
21	depto 01_2	40,64854384	1	44,57211436	d
22	depto 01_4	<b>40,79771361</b>	2	45,53246326	d
23	depto 01_6	35,78607005	2	47,26504051	h
24	depto 01_1	36,86114584	1	47,77278176	a
25	depto 01_5	45,65973525	2	49,19225788	h
26	depto 01_8	39,1340083	2	50,16060141	l
27	depto 01_8	50,16060141	2	55,19859867	l
28	depto 01_2	44,57211436	1	56,80150239	d

Tem-se ainda, neste cenário, a limitação de não se poder recorrer a operadores externos. Na tabela A3.5, faz-se uso do sombreamento para ajudar a identificar os casos em que se fez necessário o uso da flexibilidade dos operadores (*cross training*). Pode-se verificar, assim, na tabela, que as máquinas “depto 01\_1”, “depto 01\_2”, “depto 01\_4” e “depto 01\_6” têm operadores que podem operá-las como sua primeira habilidade (valor um na coluna “índice\_set”). Por outro lado, algumas vezes, essas mesmas máquinas aparecem sendo operadas por outro operador (valor diferente de um na coluna “índice\_set”), fazendo uso da flexibilidade dos operadores (*cross training*) porque o operador preferencial estava ocupado. Esse fato pode ser observado na tabela, na linha do lote 22, quando a máquina “depto 01\_4” não podendo ser operada pelo “operador\_h”, que estava ocupado operando a máquina “depto 01\_6” (linha do lote 23), acabou sendo operada pelo “operador\_d”. Por outro lado, pode-se observar também que as máquinas “depto 01\_5” e “depto 01\_8” têm que ser operadas sempre por operadores que fazem uso de sua flexibilidade (*cross training*).

Tabela A3.6 Operadores e máquinas utilizadas no celular tradicional por ordem de solicitação

Hora	Depto 01_1		depto 01_2		Depto 01_4		depto 01_5		depto 01_6		depto 01_8	
	lote	operador	lote	operador	lote	operador	lote	operador	lote	operador	lote	Operador
4,5339	1	a										
6,7555	2	a										
6,9112	2	a			4	h						
9,0812	2	a			4	h						
10,377	3	a			4	h						
12,691	11	a			4	h						
12,963	11	a			7	h						
13,147	11	a			7	h			5	l		
14,286	11	a			7	h			5	l		
14,395	11	a			7	h					6	l
14,817	11	a			7	h					6	l
16,802	11	a			8	h						
16,959	11	a			8	h	9	d				
20,208	11	a			14	h	9	d				
22,096	11	a			14	h	10	d				
22,246	11	a			14	h	10	d	15	l		
25,425	11	a			14	h	10	d	15	l		
26,524	12	a			14	h			15	l		
26,671	12	a	13	d	14	h			15	l		
28,353	17	a	13	d	14	h			15	l		
28,43	17	a	13	d	14	h			15	l		
28,496	17	a	20	d	14	h			15	l		
29,507	17	a	20	d	16	h			15	l		
30,889	17	a	20	d	16	h			18	l		
<b>31,019</b>	17	a	20	d	16	h			18	l	19	l
32,759	17	a	20	d	16	h			18	l	19	l
34,696	17	a	20	d					18	l	19	l
35,786			20	d					23	h	19	l
36,861	24	a	20	d					23	h	19	l
39,134	24	a	20	d					23	h	26	l
40,649	24	a	21	d					23	h	26	l
<b>40,798</b>	24	a	21	d	22	d			23	h	26	l
44,572	24	a	28	d	22	d			23	h	26	l

A tabela A3.6 foi construída a partir dos dados da tabela A3.5. Neste caso, os dados foram ordenados pela hora em que cada operador foi solicitado (**seize**) e não necessariamente na hora em que efetivamente foi usado, pois poderia não estar disponível quando solicitado.

Parece que o único recurso "operador\_l" está sendo usado simultaneamente nas máquinas "d1m6" e "d1m8" nos instantes 31,0187 h até 35,78607 h, mas está sendo usando apenas na máquina "d1m6" e a entidade que está segurando a "d1m8" já solicitou (deu um **seize**) no set "ex\_3\_opr\_d1\_m6" e o primeiro operador a estar disponível será o

“operador\_l” (“l” de Luis). É possível confirmar também pela animação. O mesmo ocorre no intervalo de tempo de 40,7977 a 44,57211 h com o operador “operador\_d”.

Assim, ocorrem dois tipos de uso de flexibilidade do operador: o das máquinas “depto 01\_5” e “depto 01\_8”, pois não há operadores internos com esta capacidade como a sua primeira habilidade. E a flexibilidade (*cross training*), decorrente de uma máquina precisar de um operador e este estar ocupado. É o que ocorre no momento 35,78 h. O lote 19, que ocupa a máquina “depto 01\_8”, obtém o operador “operador\_l”, único que opera esta máquina. Na seqüência o lote 23 que acaba de ocupar a máquina “depto 01\_6”, solicita um operador. O primeiro operador a ser solicitado é o operador “operador\_l”, mas como está ocupado operando a máquina “depto 01\_8”, a máquina “depto 01\_6” obtém então o operador “operador\_h” (segunda opção dentro do seu set de operadores dessa máquina) porque estava livre.

Pode-se então concluir:

1. Que os tempos finais e os (próximos) tempos iniciais do uso de uma máquina coincidem em algumas linhas da tabela A3.5, porque a máquina em questão tinha outro lote na fila. Portanto, assim que é liberada, outro lote a ocupa (dá um seize) e na seqüência ocupa também o operador. É o que se percebe nas linhas dos lotes 7, 8, 14 e 16 com relação à máquina “depto 01\_4” e o operador “operador\_h”. Essa situação pode ainda ser comprovada na simulação com relação à fila dos lotes (na máquina “depto 01\_4”, por exemplo).

2. A flexibilidade dos operadores foi verificada perfeitamente, como se pôde acompanhar pela explicação desse fenômeno no momento 35,78 h. Fato que também pôde ser observado (sombreado) nas tabelas anteriores.

3. Todos os operadores que trabalham com a máquina “depto 01\_5” ou “depto 01\_8” operam usando a flexibilidade dos operadores (*cross training*), pois o operador principal desse recurso está ausente na célula “Depto\_1”.

4. Pôde-se observar, pelos resultados, que, no sistema celular tradicional, conforme o modelo adotado, há um emprego muito expressivo da flexibilidade dos operadores.

Conforme abordado anteriormente, para testar a flexibilidade dos operadores na configuração celular, foram realizadas 10 simulações. Em cada simulação, foram obtidos os resultados para apenas uma das 5 células (“depto 1”, “depto 2”, “depto 3”, “depto 4” ou “depto 5”) com “eta==2” ou “eta==3”, redundando dessas 10 combinações os 10 experimentos ou simulações. O caso anteriormente analisado de forma mais aprofundada foi a partir da célula “depto 1” com “eta == 3”. Pôde-se observar igualmente nas 9 demais simulações que, no sistema celular tradicional, conforme o modelo adotado, há uma grande



ocorrência do emprego da flexibilidade do operador. E pôde-se verificar a adequação do modelo também no que tange às demais células, para os dois níveis de flexibilidade (“eta”) das células tradicionais.

#### A3.4.3 Descrição detalhada da verificação da flexibilidade dos operadores (*cross training*) no celular virtual

Esta verificação realiza conferências, via trace, idênticas às realizadas nos subitens anteriores, agora para o modelo celular tradicional. Com essa finalidade foram realizadas diversas simulações com o objetivo de verificar se no modelo celular tradicional a flexibilidade dos operadores (ou *cross training*) estava correta, ou seja, conforme o estabelecido no subitem 5.1.2 e na tabela 4.7. Nesse subitem e tabela são descritos como os operadores são agrupados em **sets** ou conjuntos de operadores capazes de operar determinadas máquinas e como esses **sets** são constituídos de forma diferente em função do nível de flexibilidade escolhido (**eta** = 1, 2 ou 3). Assim, por exemplo, o **set** de operadores que pode operar as máquinas do departamento 1 para eta = 1 (ou seja, habilidade para operar apenas uma máquina), designado como “operador\_1”, igual que para o modelo *job shop*, é composto pelos operadores operador\_a, operador\_b e operador\_c. Para eta =2, o set apto a realizar a mesma tarefa denomina-se “eta\_2\_operador\_1” e é constituído pelos operadores operador\_a, operador\_b, operador\_c, operador\_i, operador\_k, operador\_m e operador\_t. Por último para eta=3, o set denomina-se “eta\_3\_operador\_1” e se constitui dos mesmos operadores de “eta\_2\_operador\_1”.

Mais uma vez foi realizada uma simulação de 500 horas, com “eta” igual a 2, com lote de 50 peças e com uma única replicação. Foram transferidos para a planilha, através do módulo **ReadWrite**, os dados referentes aos primeiros 99 lotes que saíram da simulação (em função da restrição de 100 transferências por simulação). Esses dados foram gerados pelo software Arena durante a simulação do modelo, que está no arquivo, cujo nome é “Funcional\_valores\_nao\_deterministicos\_muitos\_lotes\_cross training passo a passo.doe”. Os primeiros registros (lotes saídos) encontram-se na tabela A3.7.

A primeira coluna da tabela corresponde à ordem em que os lotes saíram da simulação. As demais colunas correspondem aos dez departamentos em que se encontram as máquinas e os operadores utilizados, ou seja, ao departamento 1, 2, e 10. Esses dados correspondem aos operadores utilizados nos departamentos em que os lotes tenham passado para processar as peças. Os dados nulos significam que os lotes não passaram nos respectivos departamentos. Foram registrados nas colunas “op” de cada departamento os valores das variáveis “ind\_opr\_1”, “ind\_opr\_2”, ... “ind\_opr\_10”, que correspondem aos índices dos recursos utilizados dentro do **set**, isto é, conjunto de recursos de operadores de

cada departamento, respectivamente os sets "eta\_2\_operador\_1", "eta\_2\_operador\_2",... "eta\_2\_operador\_10". Ao lado do índice de cada operador usado, foram registrados também os tempos em que o lote conseguiu ocupar o operador, transferidos dos atributos "ini\_1", "ini\_2"..., "ini\_10" e colocados na coluna "inic" da tabela. Ao lado desses tempos, consta, ainda na coluna "fim", os tempos em que esses operadores foram liberados, transferidos dos atributos "fine\_1", "fine\_2"...., "fine\_10".

Os dados dessa tabela foram depois rearranjados e a seguir desmembrados, colocando-se os dados de cada departamento um debaixo do outro. Na seqüência foram excluídas as linhas cujos tempos fossem iguais a zero. Acrescentou-se o sufixo do nome do operador (a, b, c etc.). Aplicou-se um sombreamento nas atividades realizadas por operadores externos, ou seja, fez-se uso da flexibilidade de habilidades dos operadores. E, finalmente, ordenou-se pelo horário de início da ocupação do operador. As primeiras vinte e uma linhas dos dados resultantes encontram-se na tabela A3.8. Ou seja, cada registro passa a conter apenas os dados de uma operação, indicando o lote envolvido, o operador que a realizou, o departamento da máquina utilizada, o início e o fim de cada atividade.

A partir da tabela A3.8, foi construído o gráfico da figura A3.9. Este gráfico foi gerado tabulando-se cronologicamente, passo a passo, todas as alterações de operador ocorridas, a partir do início da simulação, ou seja, tempo igual a zero, até verificar-se a ocorrência de um *cross training*.

Tabela A3. 7 Operadores e departamentos utilizados no celular virtual por ordem da saída dos lotes

Lotes	Depto 1			Depto 2			Depto 3			Depto 4			Depto 5			Depto 6			Depto 7			Depto 8			Depto 9			Depto 10		
	op	inic	fim	op	inic	fim	op	inic	fim	op	inic	fim	op	inic	fim	op	inic	fim	op	inic	Fim	op	inic	fim	op	inic	fim	op	inic	fim
1	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	2	27,84	28,05	0	0,00	0,00	2	16,28	22,82	2	5,63	15,75	0	0,00	0,00	1	1,55	5,03	0	0,00	0,00
2	0	0,00	0,00	1	12,21	14,52	1	15,27	30,45	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	1	4,16	4,69	1	5,39	8,36	0	0,00	0,00
3	3	6,61	16,30	2	17,00	20,19	0	0,00	0,00	3	21,04	21,31	1	30,48	31,24	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00
4	1	16,13	17,25	3	18,38	23,24	0	0,00	0,00	1	28,05	28,26	2	30,86	31,62	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00
5	2	17,57	18,69	0	0,00	0,00	0	0,00	0,00	2	19,24	22,32	0	0,00	0,00	0	0,00	0,00	1	25,83	27,11	0	0,00	0,00	2	28,85	34,68	0	0,00	0,00
6	0	0,00	0,00	1	2,59	12,21	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	1	12,77	19,91	1	20,85	24,53	1	25,51	32,14	2	33,03	37,91
7	2	5,88	17,57	0	0,00	0,00	2	18,51	24,91	0	0,00	0,00	0	0,00	0,00	1	27,35	29,43	1	30,10	34,93	0	0,00	0,00	1	35,74	39,21	0	0,00	0,00
8	0	0,00	0,00	1	15,55	21,04	1	30,45	35,26	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	1	35,79	39,96	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00
9	1	1,11	3,47	0	0,00	0,00	0	0,00	0,00	1	23,58	28,05	0	0,00	0,00	2	28,67	43,52	0	0,00	0,00	2	44,51	48,39	0	0,00	0,00	0	0,00	0,00
10	0	0,00	0,00	2	39,40	48,92	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	0	0,00	0,00	1	4,34	9,05	1	9,77	15,54	1	16,51	18,38	1	21,75	23,58
11	1	42,12	47,76	0	0,00	0,00	1	48,52	49,32	0	0,00	0,00	0	0,00	0,00	1	13,02	20,24	0	0,00	0,00	2	20,90	27,62	0	0,00	0,00	1	28,62	41,48
12	4	10,52	18,88	0	0,00	0,00	4	22,47	27,48	0	0,00	0,00	3	28,26	33,31	1	34,13	51,71	0	0,00	0,00	1	0,99	3,60	0	0,00	0,00	1	4,57	9,64

Tabela A3. 8 Operadores e departamentos utilizados no celular virtual

ord	maq	#op	oper	inic	fim
12	8	1	p	0,99	3,6
9	1	1	a	1,11	3,47
1	9	1	r	1,55	5,03
14	4	1	h	2,13	19,1
6	2	1	d	2,59	12,2
42	1	1	a	3,47	16,1
2	8	1	p	4,16	4,69
10	7	1	n	4,34	9,05
12	10	1	t	4,57	9,64
2	9	1	r	5,39	8,36
1	7	2	o	5,63	15,7
7	1	2	b	5,88	17,6
3	1	3	c	6,61	16,3
26	9	2	s	6,71	12,2
17	2	2	e	7,15	8,35
13	8	1	p	7,21	9,11
15	10	2	u	8,25	9,87
10	8	1	p	9,77	15,5
13	9	1	r	9,93	11,7
12	1	4	i	10,5	18,9
15	9	3	g	10,7	13,3

Esta ocorrência se deu no tempo 10,5 horas do início da simulação. Pode-se verificar que os operadores internos, “operador\_a”, “operador\_b” e “operador\_c”, estavam ocupados operando três máquinas do departamento 1, quando surge um novo lote necessitando ser operado por outra máquina do departamento 1. Por isso, foi escolhido o “operador\_i” porque é o primeiro na ordem de cross training para “eta” igual a dois (i, k, m, t) e estava livre. Pode-se também acompanhar e visualizar esses resultados pela animação do software Arena quando se simula passo a passo (*step by step*).

Por questão de tempo, encerrou-se a geração do gráfico da figura A3.9 às 10,5 h de simulação, mas é possível verificar, também, mediante a tabela A3.8, que a próxima ocorrência do uso da flexibilidade das habilidades dos operadores ocorre às 10,7 h de simulação para atender a terceira máquina 9, quando os dois operadores internos da mesma, “operador\_r” e “operador\_s”, estão ocupados. Assim, a primeira opção de operador externo para “eta” igual a 2, o “operador\_g” é utilizada porque ele está livre.

Pode-se observar ainda que a ordem de utilização dos operadores internos também é seguida, fato particularmente simples de comprovar no departamento 1, ou seja, no uso das máquinas 1. Primeiro é acionado o “operador\_a”, depois o “operador\_b” e só depois o “operador\_c”. Exatamente na ordem prevista. O mesmo se pode concluir dos departamentos

2, 7, 9 e 10 e até mesmo dos departamentos 4 e 8 que, apesar de empregarem só um operador neste trecho reduzido de simulação, utilizam o primeiro operador na ordem estabelecida.

Foi possível, dessa forma, verificar que a flexibilidade do operador está ocorrendo conforme o previsto na configuração virtual.

tempo	A	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	R	s	t	u
0,99																8					
1,11	1															8					
1,55	1															8	9				
2,13	1							4								8	9				
2,59	1			2				4								8	9				
3,6	1			2				4										9			
4,16	1			2				4								8	9				
4,34	1			2				4						7		8	9				
4,57	1			2				4						7		8	9	10			
4,69	1			2				4						7				9	10		
5,03	1			2				4						7					10		
5,39	1			2				4						7				9	10		
5,63	1			2				4						7	7			9	10		
5,88	1	1		2				4						7	7			9	10		
6,61	1	1	1	2				4						7	7			9	10		
6,71	1	1	1	2				4						7	7			9	9	10	
7,15	1	1	1	2	2			4						7	7			9	9	10	
7,21	1	1	1	2	2			4						7	7	8		9	9	10	
8,25	1	1	1	2	2			4						7	7	8		9	9	10	10
8,35	1	1	1	2				4						7	7	8		9	9	10	10
8,36	1	1	1	2				4						7	7	8			9	10	10
9,05	1	1	1	2				4							7	8			9	10	10
9,11	1	1	1	2				4							7				9	10	10
9,64	1	1	1	2				4							7				9		10
9,77	1	1	1	2				4							7	8			9	9	10
9,87	1	1	1	2				4							7	8			9	9	
9,93	1	1	1	2				4							7	8			9	9	10
10,5	1	1	1	2				4	1						7	8			9	9	10

Figura A3.9 Gráfico com as ocupações dos operadores no celular virtual

Concluimos esta seção verificando que todos os modelos para os três tipos de configuração (*job shop*, celular tradicional e celular virtual) atendem às características estabelecidas de flexibilidade de operador.

### A3.5 Verificação de Características Específicas do Modelo *Job Shop*

Nos próximos subitens estão descritos os procedimentos realizados no modelo *job shop*, para verificar os distintos blocos que compõem a lógica dos Departamentos. - conforme a figura 5.12 e o subitem 5.2.3.

#### A3.5.1 Descrição detalhada do bloco “Entrada” do trecho Departamento dentro do modelo *job shop*

Conforme foi abordado no subitem 6.1.2, para a verificação desse bloco foi realizada uma simulação de 500 horas, com  $\eta=3$ , lote=50 peças, com uma única replicação. Foram transferidos para a planilha os dados referentes aos primeiros 99 lotes que saíram da simulação. Os dados coletados referem-se à situação dos operadores e das máquinas nos diversos departamentos, ou seja, quantidade de recursos disponíveis e de recursos ocupados. Dessa forma, pode-se verificar se o módulo Decide que há em cada bloco de entrada de cada departamento, no caso do Departamento 1, o “Decide 8”, está encaminhando os lotes que chegam para o trecho correto dentro da lógica do departamento, para todos os dez departamentos. A tabela A3.9 apresenta os dados coletados. Cada linha corresponde aos dados de um lote que foi processado nessa simulação. Os dados nulos de uma linha significam que o lote não passou pelos respectivos departamentos indicados na coluna da tabela. Assim, o lote da linha de ordem 1 foi o primeiro a sair da simulação e não passou pelos departamentos 1, 2 e 3, mas apenas pelo departamento 4. Por questão de simplificação na tabela, estão sendo apresentados apenas os primeiros 5 dos 99 lotes em que se obtiveram resultados e somente de 4 dos 10 departamentos.

Tabela A3. 9 Operadores e departamentos utilizados no exemplo do *job shop*

Ordem	Departamento 1						Departamento 2					Departamento 3					Departamento 4							
	Eta	Oper. Ocup.	Oper. Exist.	Maq. Ocup.	Maq. Exist.	Área	Eta	Oper. Ocup.	Oper. Exist.	Maq. Ocup.	Maq. Exist.	Área	Eta	Oper. Ocup.	Oper. Exist.	Maq. Ocup.	Maq. Exist.	Área	Eta	Oper. Ocup.	Oper. Exist.	Maq. Ocup.	Maq. Exist.	Área
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	1	2	1	3	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	2	2	2	3	3
3	3	2	3	2	4	0	0	0	0	0	0	0	3	2	2	1	3	3	0	0	0	0	0	0
4	3	2	3	2	4	0	0	0	0	0	0	0	3	1	2	1	3	0	0	0	0	0	0	0
5	3	3	3	1	4	3	3	2	2	1	3	3	0	0	0	0	0	0	3	2	2	3	3	3

Os dados recolhidos na tabela A3.9 correspondem ao número de operadores existentes (“Oper. Exist.”), a quantidade de operadores ocupados (“Oper. Ocup.”), ao número de máquinas existentes (“Maq. Exist.”), ao de máquinas ocupadas (“Maq. Ocup.”) e à área para a qual o lote foi transferido. Quando o lote é enviado para a área LIVRE, a coluna área recebe o valor zero. Esses dados foram transferidos ao Excel pelo módulo do software Arena ReadWrite, conforme indica a figura A3.10. Por sua vez, os atributos “valor\_eta\_1”, “oper\_ocup\_1”, “oper\_exist\_1”, “maq\_ocup\_1”, “maq\_exist\_1”, (e os demais atributos idênticos referentes aos outros nove departamentos) foram atribuídos ao lote imediatamente antes do módulo **Decide** desse bloco. O atributo “area\_enviado\_1” foi atribuído à entidade logo após o módulo **Decide**, quando a entidade já está no novo bloco.

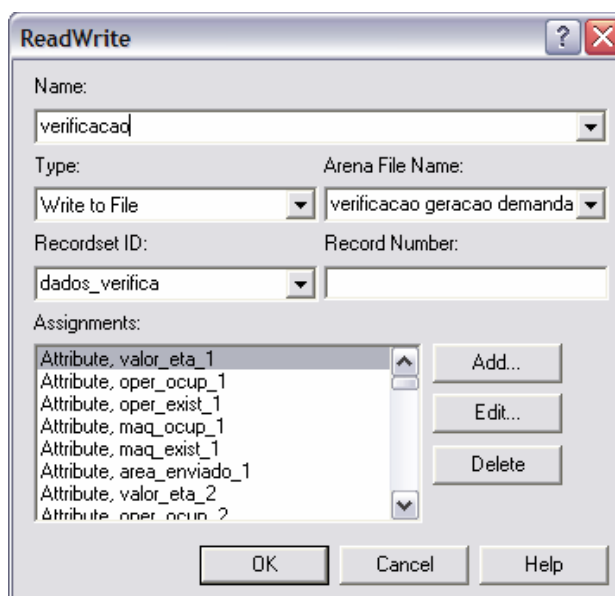


Figura A3.10 Módulo ReadWrite dos os dados dos recursos

Os dados da tabela A3.9 são rearranjados e disponibilizados na tabela A3.10. Por questão de simplificação, estão sendo apresentados apenas 16 dos 481 resultados obtidos. Quando um lote era encaminhado para o bloco do modelo denominado “Sistema Livre”, é porque havia simultaneamente máquina livre, ou seja, a diferença entre D e C era positiva, e havia operador livre, isso é, a diferença entre B e A também era positiva. Na tabela elipses destacam-se casos em que ocorreram essas circunstâncias e, de fato, o valor da área era zero, ou seja, os lotes foram para o bloco “Sistema Livre”. Caso contrário, os lotes eram direcionados a procurar máquinas e operadores disponíveis, segundo as regras de flexibilidade alta ou baixa em outros blocos. Na rodada de simulação do exemplo, o valor de

$\eta$  era 3. Foi realizada idêntica simulação para  $\eta = 2$ , obtendo-se também os resultados esperados.

Tabela A3.10 Disponibilidade de operadores e departamentos utilizadas no *job shop*

Todos os Depto							
eta	Oper. Ocup.	Oper. Exist.		Maq. Ocup.	Maq. Exist.		area
	A	B	B-A	C	D	D-C	
3	2	3	1	3	4	1	0
3	1	2	1	2	3	1	0
3	2	3	1	2	4	2	0
3	1	2	1	1	3	2	0
3	1	3	2	2	4	2	0
3	0	2	2	1	3	2	0
3	2	3	1	1	4	3	0
3	1	2	1	0	3	3	0
3	3	3	0	4	4	0	3
3	2	2	0	3	3	0	3
3	1	2	1	3	3	0	3
3	3	3	0	3	4	1	3
3	2	2	0	2	3	1	3
3	3	3	0	2	4	2	3
3	2	2	0	1	3	2	3
3	3	3	0	0	4	4	3

A3.5.2 Descrição detalhada do bloco “Sistema Livre” do trecho Departamento dentro do modelo *job shop*

Conforme foi abordado no subitem 6.1.2, para a verificação desse bloco foi realizada uma simulação de 500 horas, com  $\eta=3$ , lote=50 peças, uma única replicação. Para verificar se este bloco do modelo estava operando segundo o previsto, procurou-se constatar a ausência de fila, já que para este trecho, tanto com  $\eta=2$  como com  $\eta=3$ , só se dirigem os lotes quando há simultaneamente ao menos um operador e uma máquina livres.



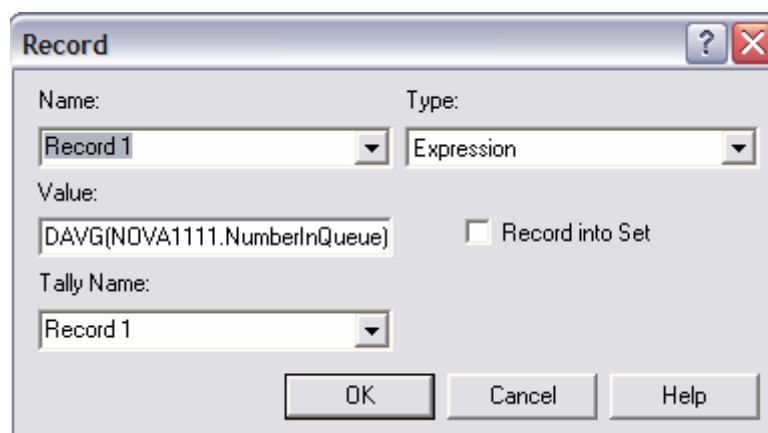


Figura A3.11 Módulo Record transferindo tamanho médio da fila “Nova1111”

Foi possível verificar ausência de fila e tempo de espera nulo para todos os trechos de sistema livre em todos os departamentos. As filas para as entidades que solicitam simultaneamente máquina e operador neste trecho são “nova1111”, “nova2222”,... “nova101010”, respectivamente dos departamentos “depto 01”, “depto 02”,... “depto 10”.

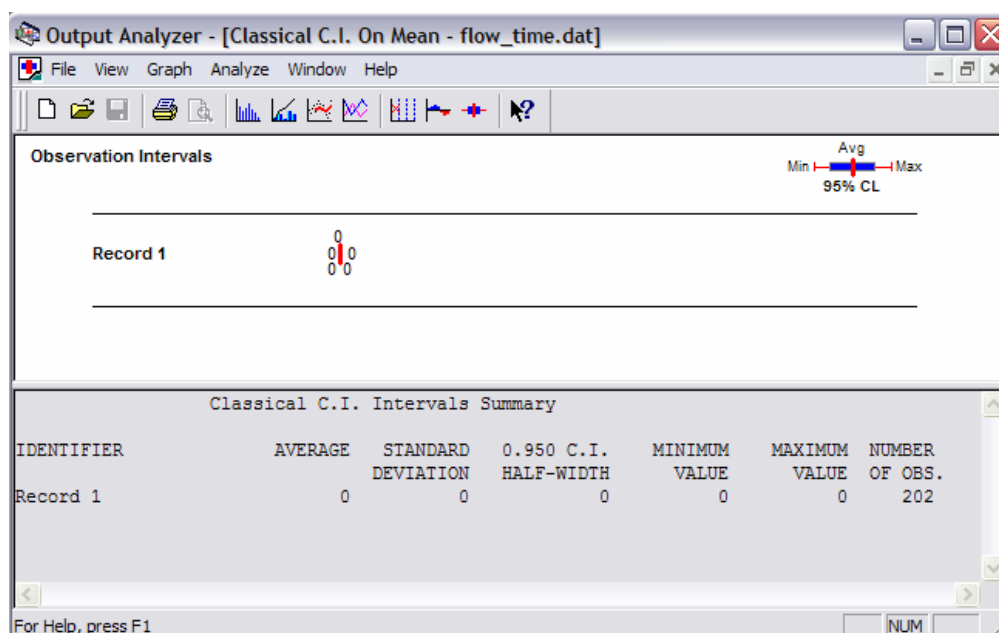


Figura A3.12 O Programa Output Analyzer analisando o tamanho médio da fila “Nova1111”

O relatório padrão do software Arena reporta que não houve filas nem tempo de espera nestes trechos do modelo. E foi empregada também a função “DAVG(NOVA1111.NumberInQueue)”, conforme a Figura A3.11, que retorna o valor médio do número de entidades (lotes) na fila “nova1111”, assim como nas demais filas dos blocos

“Sistema Livre”. E ainda foi utilizada a função “TAVG(nova1111.WaitingTime)”, que retorna o tempo médio de espera na fila “nova1111”, assim como nas demais filas dos blocos “Sistema Livre”. O programa Output Analyzer, conforme a figura A3.12, analisou os dados gerados pelas 202 entidades que saíram da simulação e constatou que não houve fila. Por outro lado, já tinha sido verificado no subitem 6.1.1.2 os tempos de processamento, *setup* etc. desse bloco.

### A3.5.3 Descrição detalhada da verificação dos blocos “Obtida uma máquina obtém operador interno” ( $\eta = 2$ e 3) e “Obtida uma máquina busca qualquer operador” ( $\eta = 2$ e 3)

Através do módulo **ReadWrite** transferiram-se os seguintes atributos para o Excel: “valor\_eta\_1”, “tempo\_ent\_1\_1”, “opcao\_busca\_op\_1”, “Op\_int\_livre\_1”, “ini\_1”, “ini\_opr\_1”, “ind\_opr\_1”, “regra”, “fila opr ext 1”, “fila velha opr ext 1”, “mov\_externo\_1”, “expug\_1” e a expressão “tempo\_saida\_1\_1+t\_sai\_1\_2+t\_sai\_1\_3” (e esses mesmos atributos para os outros nove departamentos). A figura A3.13 apresenta os dados assim que transferidos para a planilha. A seguir, esses dados foram reordenados, de forma a deixá-los de forma cronológica pela hora de chegada dos lotes nos departamentos, pela eliminação dos valores nulos e dos que chegaram após as primeiras 50 horas de execução, e lançamento dos nomes dos operadores. Este novo formato é o que está contido na figura A3.14.

Depto 1														Depto 2								
ord	eta	tempo entra	opcao busca op	oper intern livres	inicio máq	Inicio oper	indice oper	regra	fila oper ext	fila velha oper ext	mov ext	expur	Delta tempo	eta	tempo	opcao busca op	oper intern livres	inicio máq	Inicio oper	indice oper	n	
1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	3	25.291	3	1	25.291	25.291	3	1	0	0	0	0	0.6048	3	26.512	1	1	27.907	27.907	2		
3	3	6.6072	3	2	6.6072	6.6072	2	1	0	0	0	0	7.1092	3	14.43	3	1	14.43	14.43	2		
4	0	0	0	0	0	0	0	1	0	0	0	0	0	3	27.485	1	1	31.594	31.594	1		
5	3	23.105	3	1	23.105	23.105	1	1	0	0	0	0	2.2652	0	0	0	0	0	0	0	0	0
6	3	4.5339	3	3	4.5339	4.5339	1	1	0	0	0	0	15.352	0	0	0	0	0	0	0	0	0

Figura A3.13 Dados transferidos para verificar blocos dos departamentos no *job shop*

A primeira coisa que se verificou foi se, de fato, em função da disponibilidade de um operador interno, a entidade (o lote) seguia para um bloco ou outro do modelo (para “obtem um operador interno” ou “busca qualquer operador”). As conclusões desta verificação são:

1. Pode-se verificar que o trecho “sistema livre” está funcionando corretamente pela planilha da figura A3.14 tomando como exemplo a atividade 1, ou seja, a primeira linha da

planilha. Essa verificação foi possível através do atributo “opcao\_busca\_op\_#” que, neste caso, tem valor 3, o que significa que quando o lote chegou havia, simultaneamente, máquina e operador livres, ocupando-os imediatamente. Comprovam esta situação o atributo “Op\_int\_livre\_#” (quantidade de operadores internos livres) maior que zero, ou seja, os dois operadores internos estavam livres. E o atributo ind\_opr\_# comprova que efetivamente foi ocupado um dos operadores internos: na atividade 1 foi ocupado o primeiro operador interno do departamento 8, ou seja, “operador\_p”. O que comprova que ambos (operador e máquina) foram ocupados imediatamente é o fato de terem o mesmo valor os atributos “tempo\_ent\_#” (tempo de entrada do lote no departamento #), “ini\_#” (Tempo de início de ocupação da máquina pelo lote) e “ini\_opr\_#” (tempo de início da ocupação do operador pelo lote). Na atividade 1, por exemplo, todos estes tempos foram às 0,991 hora.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	Ativid	Depto	tempo entra	opcao busca op	oper intern livres	inicio máq	Inicio oper	oper	regra	fila oper ext	fila velha oper ext	mov ext	expur	Delta tempo	indice oper	tempo saída
216																
217	1	8	0,991	3	2	0,991	0,991	p	1	0	0	0	0	2,1981	1	3,1891
218	2	10	3,9896	3	2	3,9896	3,9896	t	1	0	0	0	0	3,421	1	7,4106
219	3	1	4,5339	3	3	4,5339	4,5339	a	1	0	0	0	0	15,352	1	19,886
220	4	9	6,5774	3	2	6,5774	6,5774	r	1	0	0	0	0	4,0598	1	10,637
221	5	1	6,6072	3	2	6,6072	6,6072	b	1	0	0	0	0	7,1092	2	13,716
222	6	9	6,9084	3	1	6,9084	6,9084	s	1	0	0	0	0	4,8845	2	11,793
223	7	2	7,1976	3	2	7,1976	7,1976	d	1	0	0	0	0	24,397	1	31,594
224	8	1	8,1402	3	1	8,1402	8,1402	c	1	0	0	0	0	17,149	3	25,289
225	9	10	8,2535	3	2	8,2535	8,2535	t	1	0	0	0	0	4,1023	1	12,356
226	10	8	8,9393	3	2	8,9393	8,9393	p	1	0	0	0	0	16,409	1	25,348
227	11	4	9,2314	3	2	9,2314	9,2314	h	1	0	0	0	0	4,7185	1	13,95
228	12	1	10,377	2	0	10,377	10,377	m	1	0	0	1	0	5,0173	4	15,394
229	13	1	10,408	1	1	13,716	13,716	b	1	0	0	0	0	4,6968	2	15,105
230	14	7	11,626	3	2	11,626	11,626	n	1	0	0	0	0	3,9943	1	15,621
231	15	8	11,758	3	1	11,758	11,758	q	1	0	0	0	0	3,7633	2	15,521
232	16	10	11,989	3	1	11,989	11,989	u	1	0	0	0	0	10,214	2	22,203
233	17	7	12,287	3	1	12,287	12,287	o	1	0	0	0	0	3,2285	2	15,516
234	18	7	12,689	2	0	12,689	12,689	f	1	0	0	1	0	2,4367	3	15,126

Figura A3.14 Dados reordenados para verificar blocos dos departamentos no *job shop*

2. Pode-se verificar que o trecho "obtida uma máquina obtém o operador interno" está funcionando corretamente também pela planilha da figura A3.14 na atividade 13. Essa verificação foi possível através do atributo “opcao\_busca\_op\_#” que, neste caso, tem valor 1, o que significa que quando o lote chegou não havia simultaneamente máquina e operador livres. O lote solicita (**seize**) então uma máquina. Após um tempo de espera, uma máquina foi obtida e neste momento havia também um operador interno livre. Comprovam esta

situação o atributo “ind\_opr\_#” que efetivamente demonstra que foi ocupado um dos operadores internos. Nessa atividade foi ocupado o segundo operador interno do departamento 1, ou seja, o “operador\_b”. Os 3 operadores internos - a, b e c - e as 4 máquinas do depto 1 estavam ocupados quando o lote chegou neste depto as 10,408 h. Quando a máquina 2 desse departamento foi liberada as 13,716 h pela atividade 5, também foi liberado o operador “operador\_b”. Comprovam que foram liberadas a máquina 2 e o operador “operador\_b” as 13,716 h o resultado da hora de encerramento da atividade 5. Imediatamente ambos os recursos foram ocupados pelo lote da atividade 13, o que demonstram os valores de tempo dos atributos “ini\_#” (Tempo de início de ocupação da máquina pelo lote) e “ini\_opr\_#” (tempo de inicio da ocupação do operador pelo lote) dessa atividade 13, ambos com valor de 13,716 h.

3. Pode-se verificar que o trecho “Obtida uma máquina busca qualquer operador” está funcionando corretamente também através da figura A3.14 nas atividades 12 e 18. Essa verificação foi possível através do atributo “opcao\_busca\_op\_#”, que neste caso tem valor 2, o que significa que quando o lote chegou não havia simultaneamente máquina e operador interno livres. O lote solicita (**seize**), então, uma máquina, obtém este recurso e como não há um operador interno livre, solicita qualquer operador habilitado para operar esta máquina (interno ou externo - o primeiro habilitado que estiver livre). Nessas duas atividades citados (12 e 18), assim que foi obtida a máquina, havia um operador externo habilitado disponível, que foi imediatamente ocupado. Comprovam esta situação o atributo “ind\_opr\_#”, que efetivamente demonstra que foi ocupado um dos operadores externos. Na atividade 12 foi ocupado o operador externo “operador\_m” e na atividade 18 o operador externo “operador\_f”. Tanto na atividade 12 como na 18, quando o lote chegou aos respectivos departamentos (1 e 7) havia máquina e operador externo habilitado disponíveis. Comprovam essas imediatas disponibilidades o fato de serem os mesmos os valores de tempo dos atributos “tempo\_ent\_#” (tempo de entrada do lote no departamento #), “ini\_#” (tempo de início de ocupação da máquina pelo lote) e “ini\_opr\_#” (tempo de inicio da ocupação do operador pelo lote), 10,377 h no caso da atividade 12, e 12,689 h na atividade 18.

#### A3.5.4 Descrição detalhada da verificação do “Priority” no *job shop*

Conforme foi abordado no subitem 6.1.2, a finalidade do atributo “Priority” no modelo é garantir que, quando mais de um departamento disputam um mesmo operador, a preferência recaia sobre aquele departamento que tenha a maior fila, ou seja, LNQ–*Longest, Number of Jobs in Queue*. O modelo em cada departamento tem quatro módulos **Seize**, cada um com a sua respectiva fila. Qual ou quais dessas filas devem estar incluídas na avaliação de prioridade? A figura A3.15 indica em destaque esses quatro módulos

envolvidos para  $\eta$  igual a 2. O mesmo vale para  $\eta$  igual a três, com relação aos seus respectivos módulos **Seize**. Para  $\eta$  igual a 1, não cabe esta análise, pois, pela ausência de flexibilidade dos operadores, não há disputa de operadores por diferentes departamentos.

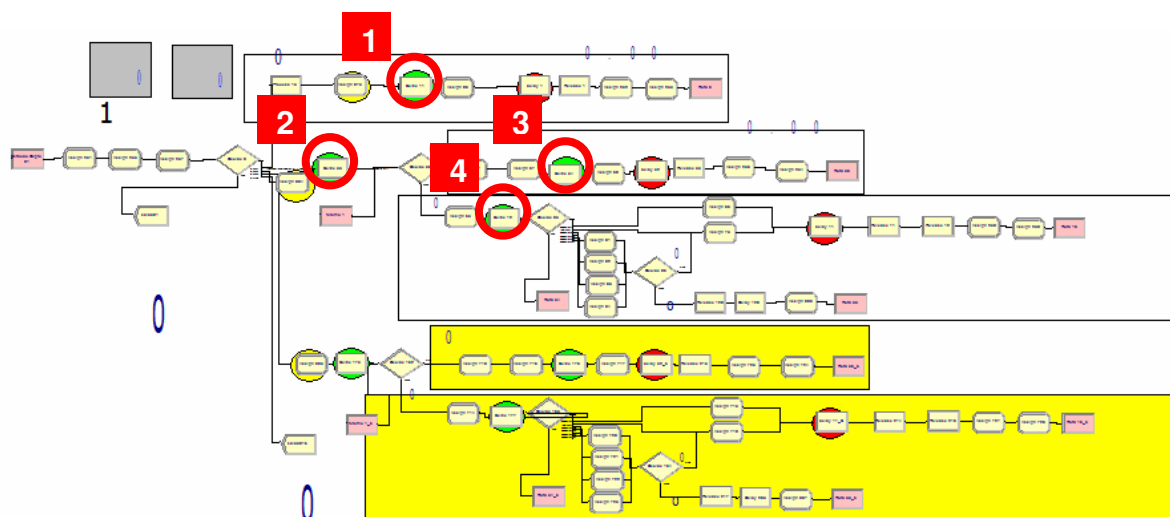


Figura A3.15 Os quatro módulos **Seize** dos departamentos para  $\eta=2$  no *job shop*

Seguindo os destaques da figura, o primeiro desses módulos é o “Seize 11”, cuja fila é “nova1111”, módulo esse que não disputa operador. Ele é usado apenas quando  $\eta=1$  ou há máquinas e operadores internos disponíveis; assim, nunca se forma fila. O segundo módulo é o “Seize 33”, cuja fila é a “nova1”. Este módulo fica no bloco “intermediário” do Departamento e a sua fila serve para conter os lotes esperando uma máquina disponível. O terceiro módulo é o “Seize 34” e sua fila é a “nova111”. Está localizado no bloco “obtida a máquina obtém o operador interno”. Quando a entidade entra neste bloco é que há um operador interno disponível; logo, nunca forma fila. O último desses módulos é o “Seize 12” com a fila “nova11111”. Está no bloco “obtida uma máquina busca qualquer operador”, e sua fila é para obter um operador (externo ou interno). Assim, só há duas filas efetivas no departamento que devem estar incluídas na avaliação de prioridade.

Por isso, a fórmula de prioridade é “ $Priority = 1/(0.5+NQ(nova1)+NQ(nova11111))$ ”, pois só essas duas filas indicam de fato os lotes que estão esperando no departamento para serem processados. A primeira fila de lote ou lotes esperando uma máquina do departamento e a segunda fila de lote ou lotes já com uma máquina conseguida, mas esperando ou disputando um operador. A função **NQ ( )** retorna o número de entidades na fila (*number in queue*). Assim, a prioridade é dada para o departamento com maior fila, pois, quanto mais próximo de zero, maior a preferência. O valor 0,5 da equação visa a evitar, quando não há fila, a divisão por zero, e nesse caso, o valor de “Priority” é 2, o que significa,

no programa, média prioridade. O valor de “Priority” está atribuído no módulo **Assign** “Assign 65” imediatamente anterior ao módulo **Seize** “Seize 12”.

Como os lotes que passam pelo bloco “obtida a máquina obtém o operador interno” e “sistema livre” conseguem o operador livre, sem disputa nem fila, só interessa então analisar os dados dos lotes que passam pelo bloco “obtida a máquina busca qualquer operador”. É possível verificar este fato analisando a tabela A3.11 (corresponde a Tabela 2 da planilha “Obtida uma maquina.xls” dentro da guia “funcional eta 3 regra 1”, que, por sua vez, é uma tabela rearranjada de dados transferidos pelo módulo **ReadWrite**). O tempo de início da máquina é sempre igual ao tempo de início do operador nesses blocos, que correspondem aos itens com valor respectivamente 3 e 1 na coluna “opção busca op”.

Tabela A3.11 Solicitação de operadores no *job shop*

Depto 1												
Depto	tempo entra	opcao busca op	oper intern livres	inicio máq	Inicio oper	indice oper	regra	Fila oper ext	fila velha oper ext	mov ext	Expur	Delta tempo
1	4,5339	3	3	4,5339	4,5339	1	1	0	0	0	0	15,352
1	6,6072	3	2	6,6072	6,6072	2	1	0	0	0	0	7,1092
1	8,1402	3	1	8,1402	8,1402	3	1	0	0	0	0	17,149
1	10,377	2	0	10,377	10,377	4	1	0	0	1	0	5,0173
1	10,408	1	1	13,716	13,716	2	1	0	0	0	0	4,6968
1	19,781	2	0	19,781	19,781	5	1	0	0	1	0	3,2781
1	23,105	3	1	23,105	23,105	1	1	0	0	0	0	2,2652
1	24,399	3	1	24,399	24,399	2	1	0	0	0	0	8,6753
1	25,291	3	1	25,291	25,291	3	1	0	0	0	0	0,6048
1	25,938	3	2	25,938	25,938	1	1	0	0	0	0	4,9775
1	26,563	3	1	26,563	26,563	3	1	0	0	0	0	1,5793
1	28,318	3	1	28,318	28,318	3	1	0	0	0	0	0,9182
1	29,634	3	1	29,634	29,634	3	1	0	0	0	0	10,167
1	41,066	2	0	41,066	41,669	7	1	0	0	1	0	14,527
1	43,475	2	0	43,475	44,39	1	1	0	0	0	0	4,9361
1	44,983	2	0	44,983	45,689	6	1	0	0	1	0	8,631
1	48,146	2	0	48,146	50,769	4	1	0	0	1	0	4,8686
1	51,478	2	0	51,478	52,146	1	1	0	0	0	0	3,1529
1	53,537	2	0	53,537	53,614	6	1	0	0	1	0	6,7005
1	57,634	3	1	57,634	57,634	3	1	0	0	0	0	6,8029

Através da planilha “novo\_priority.xls”, em especial nas tabelas 4 da guia “funcional eta 3 regra 1”(vide figura A3.16), pode se verificar que, na atividade 41, por não haver máquina disponível no “depto 2”, formou-se fila no departamento para obtenção de máquina. O atributo “Priority” passa a ser 0,6667. Isso porque é quociente de 1 dividido por 1,5 (1 na fila mais 0,5 para evitar divisão por zero).

	A	B	C	D	E	F	G	H	I	J	K	L
245	Tabela 4											
246	Ativid	Depto	tempo entra	opcao busca op	oper intern livres	sieze oper	Inicio oper	oper	tempo proce	Priorid ade	fila Mâq	fila Oper
286	40	2	23,856	3	1	23,856	23,856	e	3,8261	0	0	0
287	41	2	23,975	2	0	28,967	28,967	r	1,7473	0,6667	1	0
288	42	10	24,395	3	2	24,395	24,395	t	2,3662	0	0	0
289	43	1	24,399	3	1	24,399	24,399	b	5,5453	0	0	0
290	44	1	25,291	3	1	25,291	25,291	c	0,5784	0	0	0

Figura A3.16 O atributo "Priority" no *job shop*

Pode-se verificar também o SPT- *Shortest Processing Time*. A figura A3.17 indica que as atividades 41, 50 e 52 chegaram ao departamento 2, respectivamente, nos tempos 23,975, 25,512 e 27,485 horas. As três precisam de uma máquina 2, mas, por não haver disponibilidade imediata, só conseguem respectivamente nos tempos 28,967, 27,907 e 31,594 h. O motivo pelo qual a atividade 50 foi atendida primeiro é que a ordem das entidades na fila da máquina 2 (sendo  $\eta=3$ , é a fila nova2\_b) é dada pelo menor valor do atributo "tempo de processamento", SPT, que é dado no caso pela expressão  $\exp(0.1 \cdot q)$ . E esses valores eram respectivamente 1,7473, 0,1369 e 8,1235 h. Por isso, a atividade 50 foi a primeira a ser atendida.

	A	B	C	D	E	F	G	H	I	J	K	L
245	Tabela 4											
246	Ativid	Depto	tempo entra	opcao busca op	oper intern livres	sieze oper	Inicio oper	oper	tempo proce	Pri a		
286	40	2	23,856	3	1	23,856	23,856	e	3,8261			
287	41	2	23,975	2	0	28,967	28,967	r	1,7473	0,6667		
288	42	10	24,395	3	2	24,395	24,395	t	2,3662			
289	43	1	24,399	3	1	24,399	24,399	b	5,5453			
290	44	1	25,291	3	1	25,291	25,291	c	0,5784			
291	45	10	25,855	3	1	25,855	25,855	u	2,5952			
292	46	1	25,938	3	2	25,938	25,938	a	2,6044			
293	47	3	26,208	3	1	26,208	26,208	g	14,442			
294	48	7	26,23	2	0	26,23	26,23	s	3,5475			
295	49	3	26,367	2	0	26,367	27,687	m	5,5822			
296	50	2	26,512	1	1	27,907	27,907	e	0,1369			
297	51	1	26,563	3	1	28,563	26,563	c	1,0376			
298	52	2	27,485	1	1	31,594	31,594	d	8,1235			

Figura A3.17 O critério SPT no *job shop*

### A3.6 Descrição Detalhada do bloco “Entrada” do trecho Departamento dentro do Modelo Celular Virtual

Este item descreve os procedimentos realizados no modelo celular virtual para verificar o bloco “Entrada” que compõem a lógica dos Departamentos - conforme apresenta a figura 5.28 e o subitem 5.2.2. Este modelo acabou sendo bastante averiguado quando se fez a “verificação do novo virtual”, cujos dados e conclusões foram salvas na planilha Excel de mesmo nome. Conforme foi abordado no subitem 6.1.3.1, o objetivo dessa verificação consiste em: reproduzir a entrada dos lotes no departamento; permitir o controle visual da quantidade de lotes entrados; atribuir valores para alguns atributos e variáveis; realizar um teste múltiplo para decidir para qual bloco cada lote deve ser encaminhado. O teste deve constatar uma das quatro situações seguintes: (a) se há uma máquina livre e ajustada para a família do lote testado; (b) se há uma máquina livre não ajustada para a família do lote testado; (c) se há uma máquina ocupada, ajustada para a família do lote testado e; (d) se só há máquinas ocupadas e não ajustadas para a família do lote testado.

Para as situações (a) e (c), os lotes devem ser encaminhados para o bloco “Operando com máquina já ajustada”; para as duas outras situações, os lotes devem ser encaminhados para o bloco “Operando com máquina não ajustada”. O atributo `opc_10` armazena a situação de cada lote, no caso do Departamento 10. Na situação (a), os valores de `opc_10` serão 1, 2 ou 3, caso a entidade seja encaminhada respectivamente para a máquina livre e ajustada “Depto 10\_1”, “Depto 10\_2” ou “Depto 10\_3”. O valor desse atributo será 4, se ocorrer o caso (b). Já no caso (c), os valores de `opc_10` serão 5, 6 ou 7, caso a entidade seja encaminhada respectivamente para a máquina ajustada e ocupada “Depto 10\_1”, “Depto 10\_2” ou “Depto 10\_3”. Finalmente, o valor desse atributo será 8, se ocorrer o caso (d).

Pôde-se verificar antes que todos os elementos estavam de acordo com o planejado. Com relação ao módulo **Station**, constatou-se que os lotes estavam utilizando as 10 entradas. A seguir, constatou-se que os lotes que entram estão sendo contados no módulo **Process** e exibidos nos 10 elementos gráficos **Variable**. O módulo **Assign** subsequente, de fato, atribui os valores previstos para as variáveis e os atributos, e estes foram empregados para a verificação realizada na planilha “verificacao novo virtual.xls”.

Finalmente, pôde-se testar e verificar os módulos **Decide** e os **Assign** subsequentes, presentes nos dez departamentos. Os resultados encontram-se nas dez guias da planilha citada, uma por departamento. A seguir, descrevem-se algumas verificações mais significativas referentes ao departamento 10, que valem para todos os demais departamentos igualmente verificados.





processar o lote da atividade 2, que era da mesma família, ou seja, família 2. Assim, o lote foi para o bloco “Operando com máquina já configurada”, como comprova a opção 3 que corresponde à situação (a). Para facilitar a verificação, o fator de *setup* usado foi zero, com o objetivo de comprovar que este fator foi empregado. Pode-se confirmar que também está correto este aspecto do modelo, pois o valor na coluna “*setup*” aparece nulo. Pode-se ainda verificar que o lote 4 utiliza opção a 8 corretamente, pois todas as máquinas estavam ocupadas e nenhuma operando a família desse lote, que é a família 3. Corresponde ao caso (d). Por último, é possível verificar também o caso (c) que ocorre, por exemplo, com o lote 5 que, ao chegar no departamento 10, encontra também todas as máquinas ocupadas, mas a máquina “*Depto 10\_2*” está operando um lote da mesma família que este lote; assim, o lote foi para o bloco “Operando com máquina já configurada” com *setup* reduzido, como comprova a opção 3 e o valor nulo na coluna de “*setup*” na figura.

Entretanto, observou-se, por ocasião da redação deste capítulo, e tendo todos os resultados já obtidos e processados, que o lote 4 deu o **seize** antes do lote 5. É um detalhe secundário no modelo, pois, afinal, manteve os tempos de processamento, de *setup* e de movimentação inalterados. Porém, esse fato introduz uma pequena falta de ordem no seqüenciamento. Para evitar esta desordem, teria sido melhor que as máquinas 2, 3 e 4 (só o departamento 1 tem esta última máquina), no bloco “Operando com máquina já configurada” tivessem como prioridade do **Seize** o valor “*High(1)*” em vez de “*Medium(1)*”. Desse modo, levariam sempre vantagem com relação ao **Seize** dado no bloco “Operando com máquina não configurada”. Ainda assim, a perturbação é pequena, pois nas opções 1 e 5 está envolvida a máquina 1, que já está com a prioridade “*High(1)*”; nas opções 2 e 3, a máquina está livre, portanto não faz diferença a prioridade; já nas opções 6 ou 7, só ocorre inversão na seqüência das peças, ainda que nunca nos valores de tempo, quando, simultaneamente, dois lotes concorrem pela máquina 2 ou 3 respectivamente, e ainda assim, quando o lote que dá o **seize** pelo bloco “Operando com máquina não configurada”, tiver o tempo de processamento menor do que o outro lote. Este caso, que é muito particular e esporádico, não afetou significativamente os resultados. Foram refeitas aleatoriamente 10% das simulações com as células virtuais com essas prioridades e o resultado foi 1% superior dentro do escopo, ou seja, com lotes de 30 a 60 peças. Sendo assim, as conclusões favoráveis sobre o emprego das células virtuais que se possam tirar com os experimentos dessa tese foram 1% mais conservadoras do que poderiam ser com o emprego mais amplo da prioridade “*High(1)*”.

### A3.7 Descrição Detalhada da Verificação de Características Específicas do Modelo Celular Tradicional

O objetivo desse item é verificar se a modelagem da configuração celular tradicional, em particular do trecho da lógica de uma célula conforme a figura 5.21 e o subitem 5.3.4, não apresentam erros lógicos. Trata-se, então, de verificar a lógica da célula e o bloco “Operando sempre ocupado”.

#### A3.7.1 Descrição detalhada da verificação da lógica de uma Célula tradicional

Conforme abordado no subitem 6.1.4.1, cada célula é composta de 6 a 7 blocos (conjuntos de módulos) que representam e simulam as operações das 6 ou 7 máquinas de cada célula. A finalidade de cada bloco é representar e simular a solicitação, posse, utilização e liberação de cada máquina e operador habilitado por parte dos lotes (entidades), que pertençam a uma das famílias de peças que são processadas nesta máquina e célula. E representar a seguir o encaminhamento desses lotes para a próxima máquina da mesma célula, conforme as suas específicas seqüências de operações.

Assim, as operações que representam a simulação são: *i)* receber os lotes a serem processados por uma máquina específica alocada em uma célula específica; *ii)* gerar e armazenar um valor para o tempo de processamento específico para cada lote que passe por essa máquina específica; *iii)* contabilizar e exibir durante a simulação a quantidade de lotes que entram neste bloco; *iv)* solicitar, obter e utilizar a máquina específica desse bloco ou aguardar numa fila até que a máquina esteja disponível. *v)* gerar e armazenar o valor da prioridade do lote atual na obtenção de um operador habilitado, registrar e armazenar a informação de qual máquina está sendo utilizada pelo lote atual e qual das máquinas da célula foi utilizada; *vi)* registrar e armazenar o horário em que esta máquina e o operador começaram a ser ocupados, *vii)* buscar o operador que opera esta máquina como primeira habilidade (preferencialmente) ou outro livre em função do valor de “eta” (2 ou 3). *viii)* ordenar a fila dos lotes, esperando por um operador segundo o tempo mais curto de processamento; *ix)* definir se há tempo de movimentação do operador em função de o mesmo ter esta habilidade como sua primeira habilidade (melhor, se é de outra máquina dentro da mesma célula); *x)* reproduzir o tempo de processamento, incluindo movimentação do operador e tempo de *setup*, neste caso sempre reduzido; *xi)* liberar máquina e operador. *xii)* encaminhar o lote para a próxima operação.

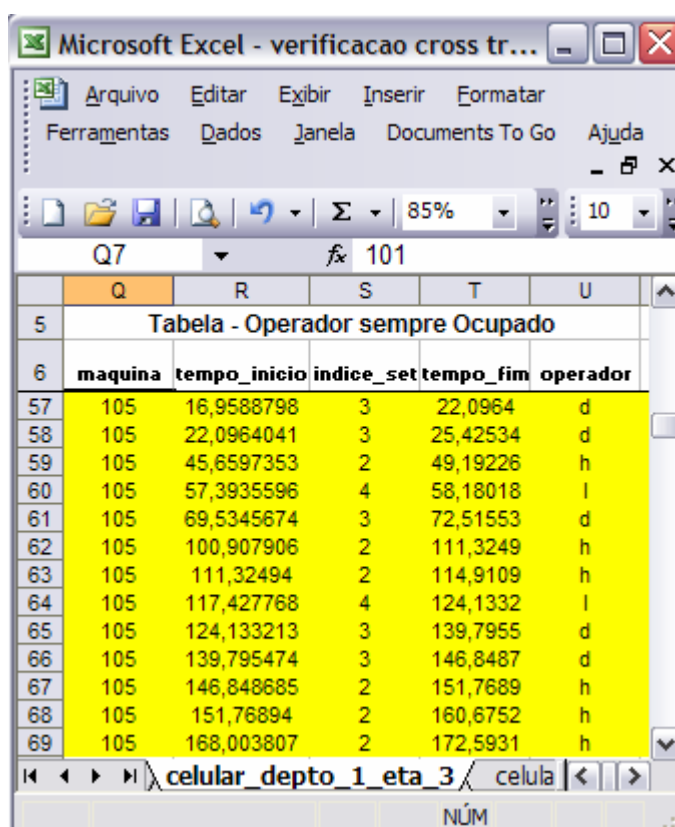
A parte da planilha “verificacao\_celular\_outras\_caracteristicas.xls” que aparece na figura A3.19 apresenta duas tabelas. A tabela mais à esquerda corresponde aos dados transferidos da simulação, já ordenados por ordem de solicitação de máquina por parte dos lotes. Cada linha da tabela corresponde à passagem de um lote por umas das máquinas da



célula não há operadores que possuam essa habilidade como primeira habilidade. De fato, tanto o **set** opr\_d1\_m8 (para “eta”=2) como o ex\_3\_opr\_d1\_m8 (para “eta”=3) tem como primeiro operador o “operador\_sempre\_ocupado”. De fato, as tabelas da figura A3.19 demonstram que o modelo está bem consistente e de acordo com o previsto.

### A3.7.2 Descrição detalhada da verificação da geração de operador sempre ocupado

Conforme abordado no subitem 6.1.4.2, os resultados gerados na verificação do *cross-training*, assim como do subitem anterior, permitem verificar que este artifício lógico está isento de erro lógico. Ou seja, o modelo deixa de tentar usar o primeiro operador do conjunto de operadores (que seria o primeiro operador segundo a lógica adotada no modelo), porque ele se encontra sempre ocupado. A finalidade é gerar tempo de movimentação interna quando uma máquina da célula não tem operador capacitado para operá-la como sua primeira habilidade.



The screenshot shows a Microsoft Excel window titled "Microsoft Excel - verificacao cross tr...". The active sheet is "celular\_depto\_1\_eta\_3". The selected cell is Q7, containing the formula "=101". The table "Tabela - Operador sempre Ocupado" is displayed with the following data:

	Q	R	S	T	U
5	Tabela - Operador sempre Ocupado				
6	maquina	tempo_inicio	indice_set	tempo_fim	operador
57	105	16,9588798	3	22,0964	d
58	105	22,0964041	3	25,42534	d
59	105	45,6597353	2	49,19226	h
60	105	57,3935596	4	58,18018	l
61	105	69,5345674	3	72,51553	d
62	105	100,907906	2	111,3249	h
63	105	111,32494	2	114,9109	h
64	105	117,427768	4	124,1332	l
65	105	124,133213	3	139,7955	d
66	105	139,795474	3	146,8487	d
67	105	146,848685	2	151,7689	h
68	105	151,76894	2	160,6752	h
69	105	168,003807	2	172,5931	h

Figura A3.20 A planilha com a tabela “Operador sempre Ocupado”

A figura A3.20 mostra a tabela “Operador sempre Ocupado” que aparece na guia “celular\_depto\_1\_eta\_3” da planilha “verificacao cross training.xls”. No trecho selecionado

da planilha, aparece a máquina 5 da Célula 1 (Depto 05\_1) do sistema celular. Esta máquina não tem operador “interno” dedicado, como podemos verificar pela tabela A3.12 que apresenta os operadores habilitados para cada máquina da célula 1 do sistema celular tradicional. E pelos resultados da tabela da figura 6.20, verifica-se que só se utilizam os operadores “h”, “d” e “l”, que correspondem aos índices 2, 3 e 4 do set de operadores dessa máquina “Depto 05\_1”, ou seja, “ex\_3\_opr\_d1\_m5”. O mesmo ocorre com todos os outros casos em que a máquina não dispõe de operador interno com essa capacidade como a sua primeira habilidade, ou seja, com as máquinas “Depto 01\_8”, “Depto 02\_2”, “Depto 02\_10”, “Depto03\_1”, “Depto 03\_1”, “Depto 03\_3”, “Depto 04\_4”, “Depto 04\_7”, “Depto 05\_6” e “Depto 05\_9”, o que se verifica também pelos resultados.

Tabela A3.12 Operadores habilitados para cada máquina da célula 1

d1																	
d1m1			d1m2			d1m4			d1m5			d1m6			d1m8		
eta	ind_opr	oper	eta	ind_opr	oper	eta	ind_opr	oper	eta	ind_opr	oper	eta	ind_opr	oper	eta	ind_opr	oper
2	1	a	2	1	d	2	1	h	2	1		2	1	l	2	1	
2	2		2	2	a	2	2	d	2	2	h	2	2		2	2	l
2	3		2	3		2	3		2	3		2	3		2	3	
3	1	a	3	1	d	3	1	h	3	1		3	1	l	3	1	
3	2		3	2	a	3	2	d	3	2	h	3	2	h	3	2	l
3	3		3	3		3	3	a	3	3	d	3	3		3	3	
3	4		3	4		3	4		3	4	l	3	4		3	4	

■ - indica que não há operador na célula com sua primeira habilidade direcionada a operar a máquina em questão

### A3.8 Descrição Detalhada do Bloco de Verificação do Tempo Necessário de *warm-up*

Este bloco é composto por 3 módulos:

Módulo **Create** “Relógio” – Uma entidade lógica é gerada a cada 100 horas (como um relógio fixo), com a primeira criação em 100 horas.

Módulo **Assign** “Produtividade” - a variável pecasporhora recebe o valor da expressão “(Processo Expedicao.NumberOut – saídaanterior) / 100”. E a variável “saídaanterior” recebe depois o valor “Processo Expedicao.NumberOut”, isto é, o total de lotes que saíram do sistema. Assim, a produtividade será medida e calculada pelo número de peças que foram produzidas nas últimas 100 horas, dividido por 100, ou seja, pelo número médio de lotes de peças por hora.

Módulo **Dispose** “Dispose 13” – retira do sistema as entidades lógicas criadas neste trecho da simulação.

### A3.9 Descrição Detalhada da Validação

Conforme abordado no item 6.2, o conceito de validação não se aplica aqui com total propriedade, uma vez que os modelos não estão reproduzindo um cenário real particular, mas conceitual, constituído de modelos teóricos adaptados ou reformulados a partir de pesquisas anteriores. Entretanto, admite-se uma validação num sentido mais amplo, comparando-se as características de desempenho dos modelos do novo padrão conceitual, a intensidade dos valores de resposta, o perfil gráfico dos resultados com os do padrão original. Avaliar a semelhança do padrão adotado na tese com o cenário conceitual apresentado por Suresh e Slomp (2005), é o que efetivamente se realiza a seguir, com a ajuda de gráficos.

Para facilitar a comparação e ampliar o escopo da avaliação, são apresentadas também as tabelas com os dados que deram origem a esses gráficos. Foi empregado o mesmo padrão utilizado por aqueles autores na tabela que denominam como “Mean flow ratio (COMP-LNQ-SPT)”. Nesta tabela, encontram-se os resultados da variável resposta tempo de fluxo, sob a forma de razão de fluxo médio. Esta razão é o quociente entre o tempo médio que os lotes gastam para serem totalmente processados e o tempo de 60,549 horas, que é também o tempo médio gasto, encontrado por aqueles autores, para que lotes de 50 peças no sistema *job shop* com “eta” igual a um, ao percorrem todos os processos. Além disso, corresponde também ao caso “COMP” nos critérios de prioridade LNQ e SPC, que são também empregados nesta tese, conforme abordado no capítulo anterior. São comparados os valores e gráficos daqueles autores com alguns dos resultados e gráficos desta tese.

Para efeito dessa comparação, optou-se por empregar os modelos da tese com demanda constante, com intervalos entre chegadas também constantes, variando-se, entretanto, os níveis de flexibilidade dos operadores, os níveis de redução de *setup*, e os tamanhos de lote. Procurou-se introduzir um mínimo de variações, uma vez que já havia diversas diferenças entre o padrão da tese e o desses autores, além de pontos obscuros, confusos e mesmo ausentes no padrão original, que tornaram a replicação inviável. Entre outros, conforme abordado anteriormente, diferenças entre os fatores e os níveis adotados, ausência de dados sobre o sequenciamento de produção que afeta diretamente a carga do sistema e o tempo de fluxo. E, ainda, falta de clareza no cálculo dos intervalos entre chegadas de lotes de peças, valores ausentes em tabelas, algumas especificações de filas pouco claras e descrições ambíguas de algumas das regras de prioridade.

Além da comparação com os dados utilizados referentes aos três modelos, *job shop* (JS), celular tradicional (CL) e o celular virtual (VCM), acrescentaram-se também os dados

do celular virtual com as prioridades de **Seize** de todas as máquinas dos blocos “Operando com máquina já ajustada” dos dez departamentos com valor alto, ou seja, “High(1)” (VCM\_High). O intuito de acrescentar esses últimos dados como se fossem de um quarto modelo, teve o propósito de reforçar, por um lado, o fato de que os resultados obtidos e utilizados na tese, dentro do escopo adotado (ou seja, para tamanhos de lotes de 30, 40, 50 e 60 peças conforme abordado no subitem 6.1.3.1), efetivamente não se distinguem substancialmente desses outros obtidos com o valor alto de prioridade. Pode-se, então, confirmar que a variação entre ambos é em torno apenas de 1 %. E por outro lado, o intuito foi também trazer mais dados para efeito de validação do modelo virtual, uma vez que, para valores de lote inferiores a 30 peças, as diferenças são muito significativas.

As Tabelas A3.13 e A3.14 apresentam os dados segundo o padrão “Mean flow ratio (COMP-LNQ-SPT)”, descrito há pouco, obtidos respectivamente por aqueles autores e pelos modelos dessa tese. As colunas referentes aos valores de tamanho de lote iguais a 30, 40, 50 e 60, que correspondem ao escopo da tese, estão destacadas com um fundo cinza (sombreadas).

Tabela A3.13 Razão de tempo médio de Fluxo obtidos por Suresh e Slomp (2005)

Sistemas	eta	10	15	20	25	30	35	40	50	60	80	100
JS	1				3,305	1,921	1,256	1,035	1	1,038	1,186	1,349
	2					2,492	0,888	0,815	0,831	0,885	1,018	1,175
	3					2,389	0,803	0,753	0,782	0,835	0,973	1,134
Redução Moderada do <i>setup</i> (40%)												
CL	1	(inviável)										
	2			4,208	1,415	0,995	0,944	0,951	1,024	1,112	1,365	1,592
	3			0,92	0,706	0,687	0,71	0,747	0,846	0,927	1,128	1,35
VCM	1			3,077	0,947	0,856	0,871	0,888	0,942	1,019	1,179	1,351
	2			4,04	0,971	0,771	0,761	0,761	0,807	0,868	1,008	1,16
	3				1,229	0,742	0,711	0,712	0,763	0,822	0,966	1,128
Redução Alta do <i>Setup</i> (75%)												
CL	1	(inviável)										
	2	0,725	0,403	0,424	0,481	0,515	0,582	0,656	0,785	0,883	1,159	1,396
	3	0,306	0,303	0,343	0,391	0,441	0,49	0,547	0,655	0,752	0,967	1,186
VCM	1	0,684	0,611	0,636	0,665	0,696	0,747	0,787	0,873	0,955	1,133	1,314
	2	4,723	1,14	0,741	0,684	0,674	0,698	0,718	0,779	0,849	0,994	1,161
	3		3,449	0,898	0,695	0,651	0,663	0,685	0,747	0,811	0,958	1,122



Tabela A3.14 Razão de tempo médio de Fluxo obtidos nesta Tese

Sistemas	eta	10	15	20	25	30	35	40	50	60	80	100
JS	1			2,59955	2,26924	1,97856	1,56353	1,30919	1,15592	1,19292	1,31266	1,47831
	2			2,78617	2,40797	1,78302	1,31761	0,97095	0,94535	0,99209	1,14255	1,29515
	3			2,78948	2,44265	2,1751	1,66791	0,91281	0,90604	0,95196	1,10539	1,24296
Redução Moderada do <i>setup</i> (40%)												
CL	1	(inviável)										
	2	3,0306	2,28245	1,51448	1,20646	1,24725	1,32124	1,28243	1,27962	1,29581	1,49449	1,75213
	3	2,99922	2,27749	1,43223	0,82842	0,76186	0,77739	0,80761	0,91381	1,01075	1,25204	1,47121
VCM	1		3,33119	2,40797	1,48062	1,07384	1,01009	0,99044	1,03255	1,12851	1,29234	1,46493
	2		4,35515	3,14786	1,69284	0,90241	0,80811	0,81587	0,86525	0,93792	1,09878	1,29449
	3		4,62105	3,6681	2,28245	0,91265	0,78647	0,78102	0,83552	0,89663	1,05683	1,2058
VCM_High	1		2,69369	1,99343	1,23437	1,04081	0,97277	0,96054	1,03074	1,12008	1,27962	1,46526
	2		3,09006	2,37163	1,33099	0,88556	0,81257	0,8119	0,86773	0,92768	1,08557	1,2755
	3		3,18915	2,86049	1,81721	0,87945	0,77656	0,76599	0,82941	0,89167	1,05865	1,2205
Redução Alta do <i>Setup</i> (75%)												
CL	1	(inviável)										
	2	0,63172	0,60761	0,54138	0,56351	0,59572	0,64791	0,71661	0,85584	0,9764	1,26278	1,48706
	3	0,4157	0,33791	0,38068	0,43353	0,48473	0,54088	0,59968	0,71876	0,82479	1,07929	1,29713
VCM	1	2,7102	1,28739	0,82099	0,76566	0,7617	0,77227	0,81257	0,89118	0,99457	1,17376	1,37145
	2	3,92409	2,15363	0,81934	0,66624	0,6517	0,66293	0,69514	0,76384	0,85237	1,0058	1,20151
	3	4,32047	3,03225	1,0745	0,67185	0,62511	0,67185	0,66558	0,74056	0,81686	0,98565	1,15609
VCM_High	1	1,24478	0,89564	0,74733	0,73048	0,74039	0,77937	0,80365	0,8887	0,99027	1,17508	1,39589
	2	1,87451	1,1802	0,74122	0,65963	0,64394	0,65914	0,68242	0,77012	0,84328	1,00745	1,19209
	3	2,13711	1,67187	0,86294	0,65005	0,62214	0,63453	0,66145	0,72949	0,81702	0,97937	1,14436

Por outro lado, a Tabela A3.15 apresenta a diferença percentual entre os dados obtidos no celular virtual (VCM) e aqueles obtidos no celular virtual com as prioridades de **Seize** de todas as máquinas dos blocos “Operando com máquina já ajustada” dos dez departamentos com valor alto, ou seja, “High(1)” (VCM\_High). O cálculo foi realizado através da expressão “(VCM - VCM\_High) / VCM\_High”. O valor médio obtido para os dados da região com fundo cinza, ou seja, da escopa da tese, é de 1,17%.

Tabela A3.15 Resultados obtidos nesta Tese

Sistemas	eta	10	15	20	25	30	35	40	50	60	80	100
Redução Moderada do <i>setup</i> (40%)												
(VCM - VCM_High) / VCM_High	1		24%	21%	20%	3%	4%	3%	0%	1%	1%	0%
	2		41%	33%	27%	2%	-1%	0%	0%	1%	1%	1%
	3		45%	28%	26%	4%	1%	2%	1%	1%	0%	-1%
Redução Alta do <i>Setup</i> (75%)												
(VCM - VCM_High) / VCM_High	1	118%	44%	10%	5%	3%	-1%	1%	0%	0%	0%	-2%
	2	109%	82%	11%	1%	1%	1%	2%	-1%	1%	0%	1%
	3	102%	81%	25%	3%	0%	6%	1%	2%	0%	1%	1%

Apresentam-se, a seguir, os gráficos que ajudam a avaliar a semelhança do padrão adotado na tese com o cenário conceitual apresentado por Suresh e Slomp (2005), com intuito de validação dos modelos desta tese. As curvas simples de cada gráfico correspondem aos resultados obtidos com esses modelos, a curva com pequenos pontos quadrados corresponde aos resultados obtidos por aqueles autores.

A Figura A3.21 apresenta os gráficos do *job shop* para os três níveis de flexibilidade dos operadores ( $\eta = 1, 2$  e  $3$ ). Percebe-se nitidamente que ambas as curvas são idênticas, diferenciando-se apenas por uma pequena defasagem, quase constante nos três gráficos, que significa que os modelos da tese apresentaram um tempo médio de fluxo maior do que o daqueles autores. Isso não é um demérito aos resultados da tese, uma vez que, conforme abordado anteriormente, não se dispõe de todos os dados daqueles autores, em especial da seqüência de produção. Bastaria reduzir o número de estágios de algumas peças dos modelos da tese, permanecendo perfeitamente dentro do padrão adotado, que o quadro se reverteria. O importante é a forma das curvas e o fato de a defasagem não ser grande.

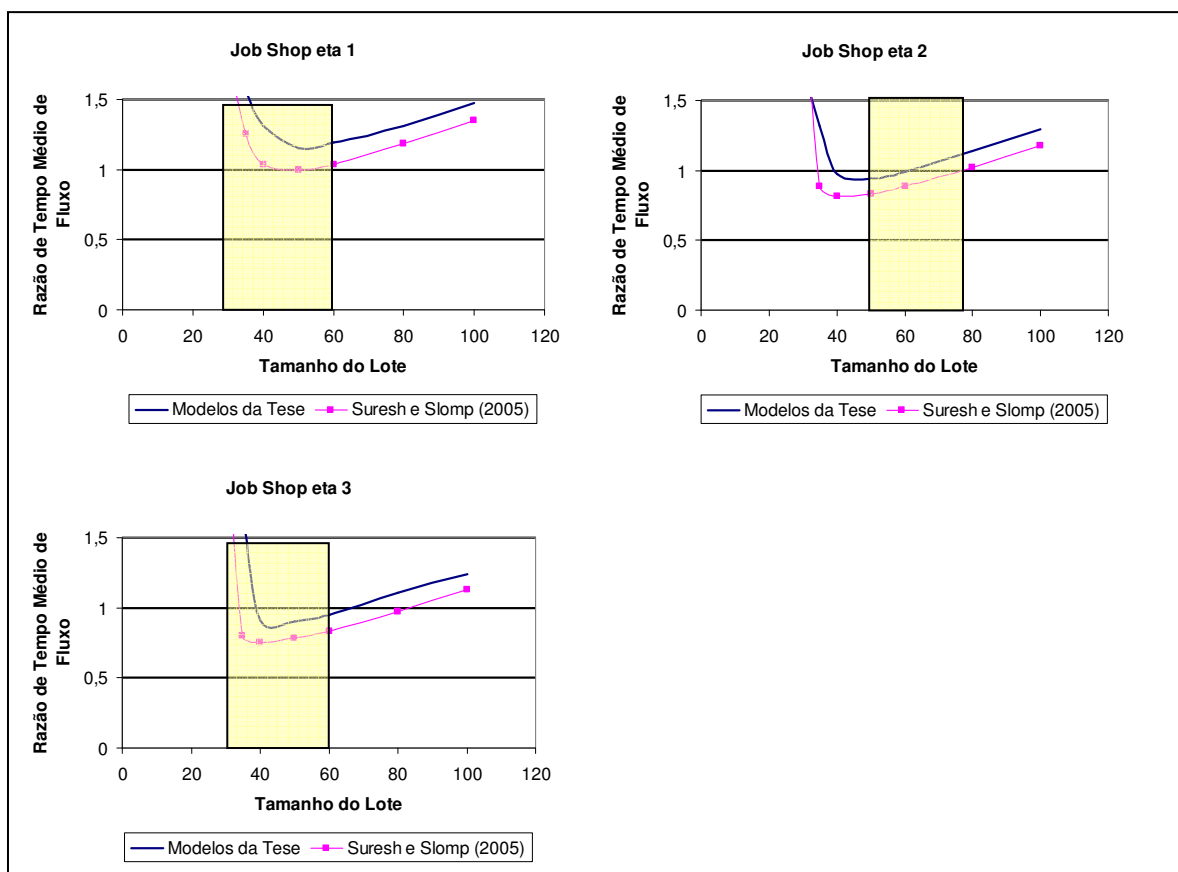


Figura A3.21 – Gráficos da configuração *job shop*

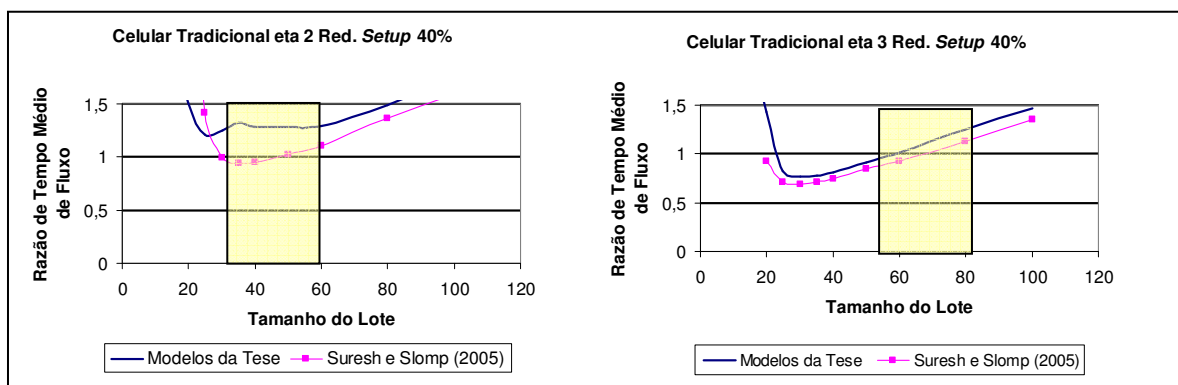


Figura A3.22 – Gráficos da configuração celular tradicional redução de *setup* de 40%

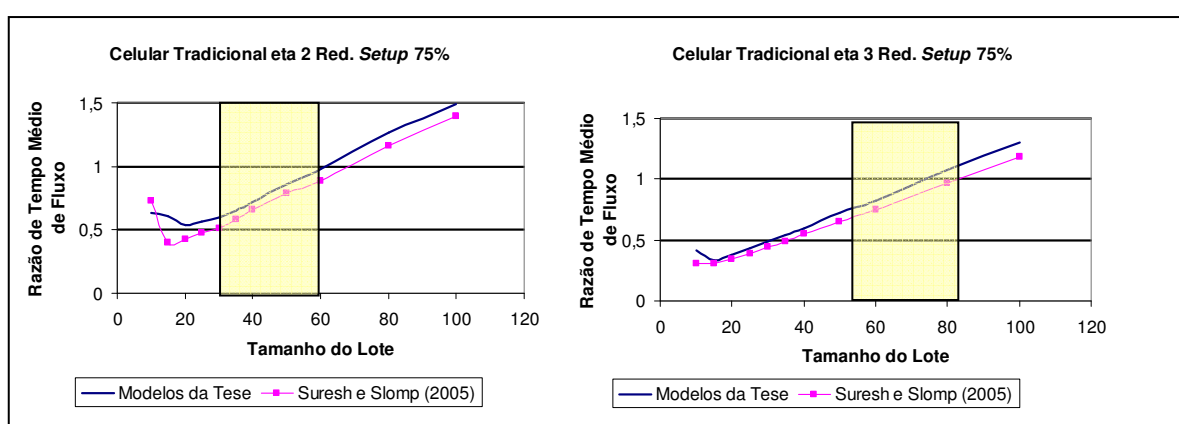


Figura A3.23 – Gráficos da configuração celular tradicional com redução de *setup* de 75%

As figuras A3.22 e A3.23 apresentam os gráficos do celular tradicional para os dois níveis de flexibilidade dos operadores que este modelo admite ( $\eta = 2$  e 3). A primeira figura contém os gráficos com os resultados obtidos com nível de redução de *setup* de 40%; a segunda, aqueles com nível de redução de 75%.

O modelo celular tradicional com “eta = 2” e redução de *setup* de 40% é o único caso em que os gráficos não coincidem bem. Foram refeitas as simulações para esses parâmetros e obtiveram-se os mesmos resultados. Mas, ao mudar o tamanho de lote para valores próximos, 31, 36 e 41 peças por lotes, o gráfico com os modelos da tese tendeu a assemelhar-se ao daqueles autores, ainda que com certa defasagem. Portanto, considera-se que as diferenças se devam a fatores aleatórios. Os outros três gráficos são idênticos aos daqueles autores, com uma defasagem muito pequena.

Já a figura A3.24 mostra os gráficos da configuração celular virtual com redução de *setup* de 40%. Os gráficos da esquerda da figura comparam aqueles autores com os resultados obtidos na tese para esses modelos. Por outro lado, os gráficos do lado direito representam os dados obtidos no celular virtual com as prioridades de **Seize** de todas as

máquinas dos blocos “Operando com máquina já ajustada” dos dez departamentos com valor alto, ou seja, “High(1)” (VCM\_High). Os gráficos da esquerda e da direita identificam-se quase completamente para tamanhos de lotes maiores ou iguais a trinta peças. Assim, também de forma gráfica, verifica-se que a diferença entre eles é imperceptível, podendo-se considerar ambos os resultados representativos do modelo virtual estudado. Por outro lado, pode-se atestar que esses resultados e os daqueles autores novamente são idênticos, com uma pequena defasagem, como nos casos anteriores, mas, neste caso, ainda menor

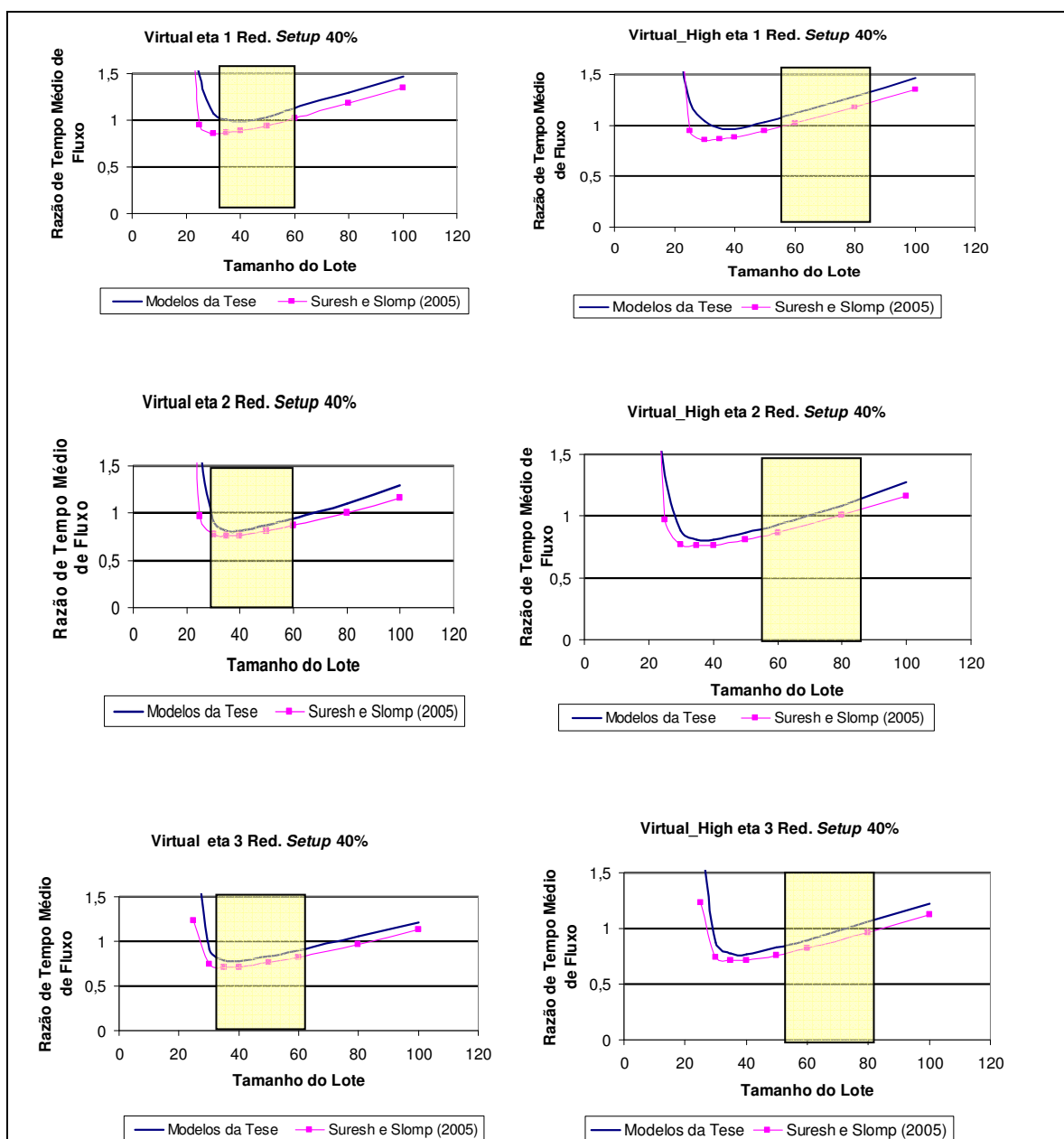


Figura A3.24 – Gráficos da configuração celular virtual com redução de *setup* de 40%

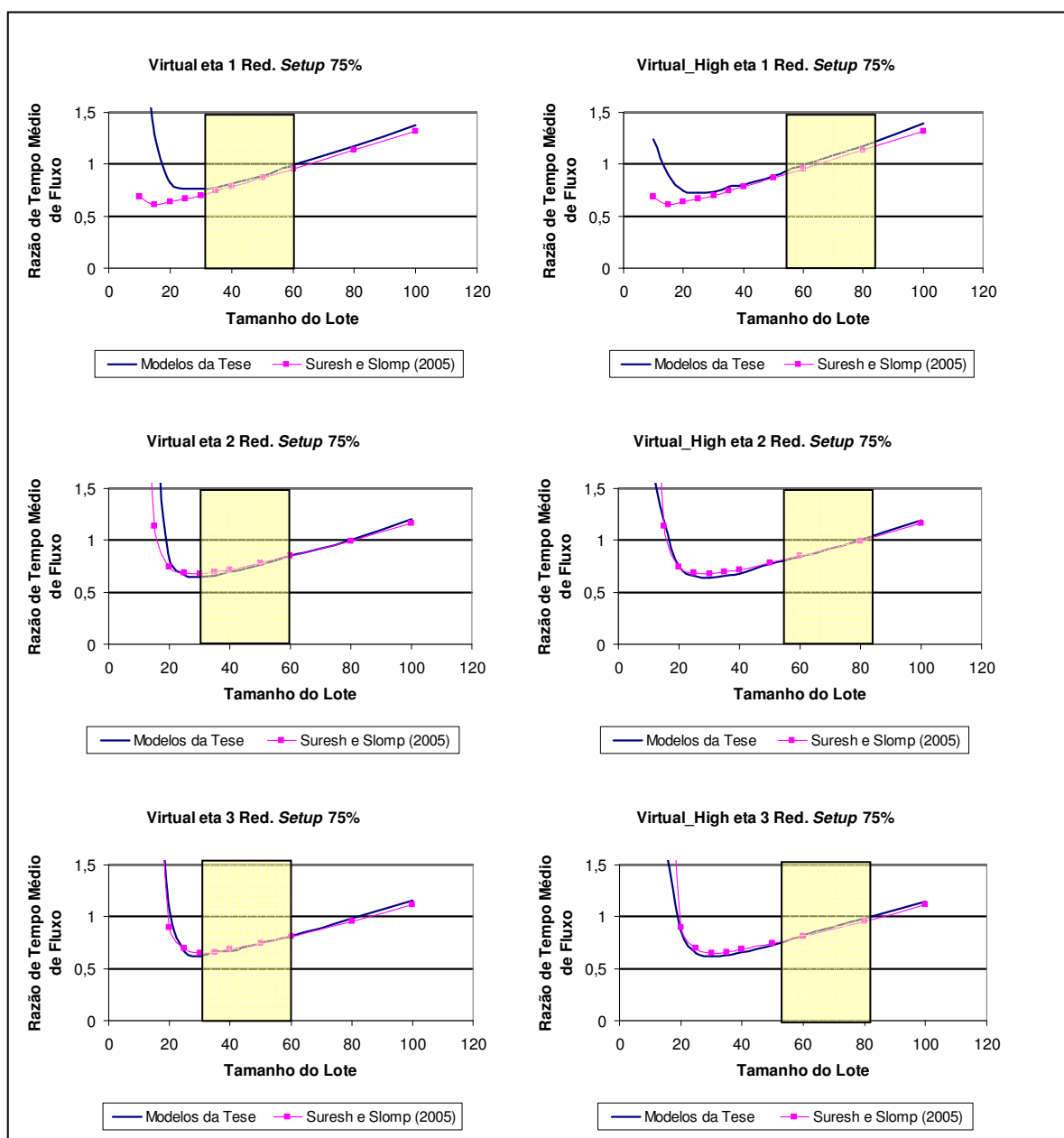


Figura A3.25 – Gráficos da configuração celular virtual com redução de *setup* de 75%

Por fim, a figura A3.25 mostra os gráficos da configuração celular virtual com redução de *setup* de 75%. Os gráficos da esquerda da figura comparam aqueles autores com os resultados obtidos na tese para esses modelos. Por outro lado, os gráficos do lado direito representam os dados obtidos no celular virtual com as prioridades de **Seize** de todas as máquinas dos blocos “Operando com máquina já ajustada” dos dez departamentos com valor alto, ou seja, “High(1)” (VCM\_High). Mais uma vez, pode-se observar que os gráficos da esquerda e da direita se identificam quase completamente, ao menos para tamanhos de lotes maiores ou iguais a trinta peças, que é o escopo da tese. Assim, também de forma

gráfica, verifica-se que a diferença entre eles é imperceptível, podendo-se considerar ambos os resultados representativos do modelo virtual estudado. Por outro lado, pode-se atestar que esses resultados e os daqueles autores novamente são idênticos, neste caso sem nenhuma defasagem. Uma vez que nos demais modelos há certa defasagem, em função do desconhecimento de dados daqueles autores, deveria haver também defasagem nesse modelo. Esta última constatação leva a crer que esta nova concepção de célula virtual tem um desempenho superior às demais configurações, inclusive à celular virtual daqueles autores. No capítulo 7, aprofunda-se esta questão ao fazer a apresentação e avaliação dos resultados obtidos.

## APÊNDICE 4

### RESULTADOS DOS EXPERIMENTOS: TABELAS E GRÁFICOS

Este apêndice tem por finalidade aprofundar e detalhar os resultados apresentados no capítulo 7 – Resultados dos Experimentos, agregando todas as tabelas e gráfico gerados pelos experimentos, permitindo assim uma análise mais meticulosa dos resultados e conclusões obtidos.

#### A4.1 Resultados envolvendo o tempo médio de fluxo

A seguir, apresentam-se os resultados obtidos com relação à diferença de desempenho no tempo médio de fluxo entre os três modelos avaliados para  $\eta = 1, 2$  e  $3$ . Para cada um dos fatores, são apresentados a análise de variância e os gráficos “Scatterplots” com regressão linear, quadrática ou cúbica, com as curvas passando pelos pontos médios dos pontos plotados.

##### A4.1.1 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o job shop (JS) para $\eta = 1$

A tabela A4.1 apresenta os valores da análise de variância da diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 1$ . A tabela apresenta ainda os graus de liberdade (DF), a diferença quadrática (SS), a média da diferença quadrática (MS), que é a razão entre SS e DF, F da distribuição de Snedecor e a probabilidade de erro do fator não influenciar significativamente na variável resposta. Para o nível de significância adotado 99,0%, são considerados significativos os fatores com  $P \leq 0,010$ .

Tabela A4.1 Análise de variância tempo médio de fluxo VCM-JS para  $\eta = 1$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1438	719	4,25	0,015
Inter Chegada	1	10638	10638	62,86	0,000
Redução Setup	1	31042	31042	183,44	0,000
Tam. Lote	3	256714	85571	505,67	0,000
Error	472	79873	169		
Total	479	379706			

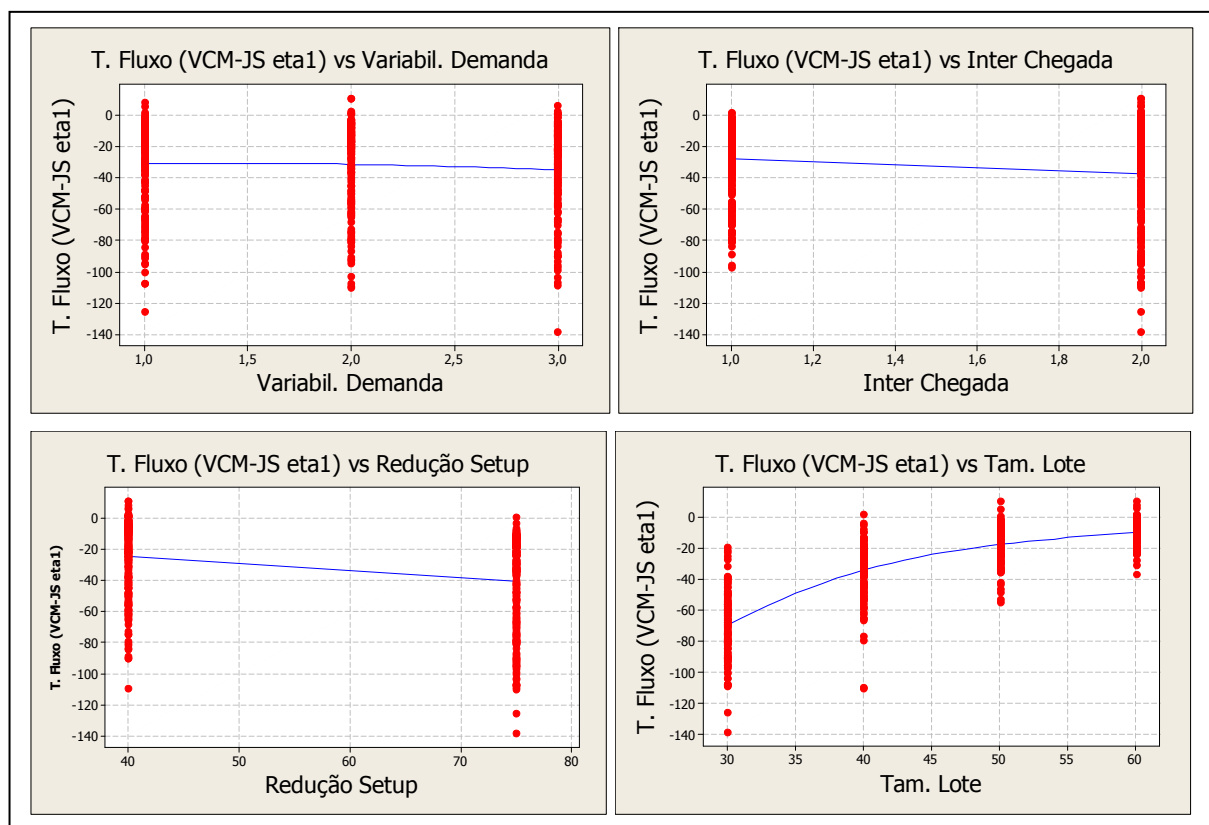


Figura A4.1 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para  $\eta = 1$

A figura A4.1 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de Fluxo entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento para  $\eta (\eta) = 1$ .

Os dados da análise de variância da tabela A4.1 indicam que os fatores que influenciam mais a diferença de desempenho no tempo de fluxo entre o celular virtual e o *job shop* são o tamanho do lote, seguido da redução de *setup* e logo a seguir do intervalo entre chegadas, nessa ordem. Indicam também que a variabilidade de demanda não influencia significativamente para o nível adotado de 99%, uma vez que seria significativa para um nível de significância 98,5 % ( $1 - P$ , onde o valor de  $P$  fornecido na tabela é 0,015).

Por outro lado, os gráficos da figura 7.1 permitem verificar que, no escopo analisado, o celular virtual tem um desempenho superior ao *job shop* em qualquer dos níveis dos quatro fatores. Os valores da diferença VCM-JS sendo negativos, indicam de fato que o tempo de fluxo do celular virtual é inferior ao do *job shop*. Por outro lado, pôde-se verificar que a vantagem das células virtuais aumenta na medida em que é menor o tamanho de lotes, maior a redução de *setup* (em outras palavras, quanto menor o *setup*) e menor a regularidade das chegadas (ou seja, quanto mais irregulares forem esses intervalos).



#### A4.1.2 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 2$

A tabela A4.2 apresenta os valores da análise de variância da diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 2$ .

Tabela A4.2 Análise de variância tempo médio de fluxo VCM-JS para  $\eta = 2$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1442	721	3,00	0,051
Inter Chegada	1	13667	13667	56,78	0,000
Redução Setup	1	15428	15428	64,09	0,000
Tam. Lote	3	344052	114684	476,42	0,000
Error	472	113621	241		
Total	479	488211			

A figura A4.2 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de Fluxo entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para  $\eta = 2$ .

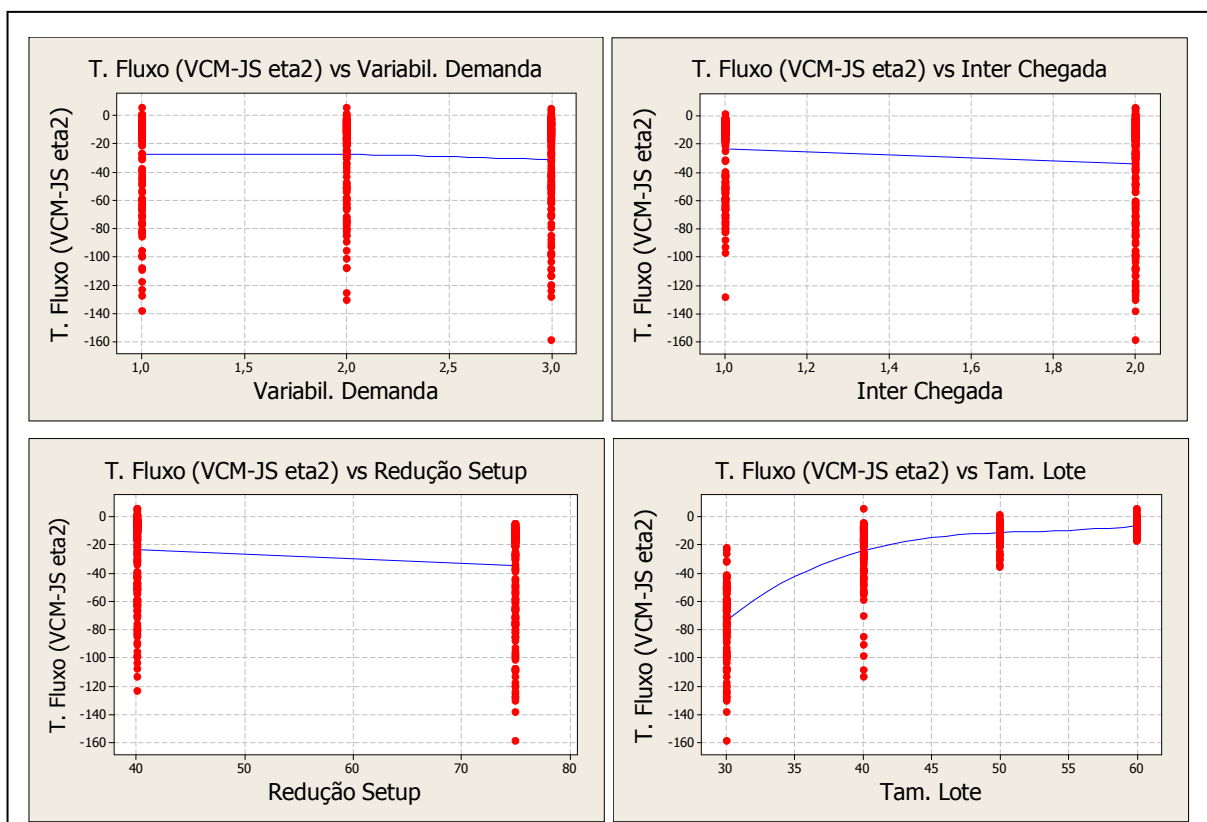


Figura A4.2 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para  $\eta = 2$

A tabela A4.2 e os gráficos da figura A4.2 são muito semelhantes aos obtidos no que se refere à diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 1, portanto, chega-se às mesmas conclusões anteriores.

#### A4.1.3 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 3

A tabela A4.3 apresenta os valores da análise de variância da diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 3.

Tabela A4.3 Análise de variância tempo médio de fluxo VCM-JS para eta ( $\eta$ ) = 3

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1982	991	2,48	0,085
Inter Chegada	1	22338	22338	55,93	0,000
Redução Setup	1	18721	18721	46,87	0,000
Tam. Lote	3	655996	218665	547,48	0,000
Error	472	188520	399		
Total	479	887557			

A figura A4.3 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de Fluxo entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para eta ( $\eta$ ) = 3.

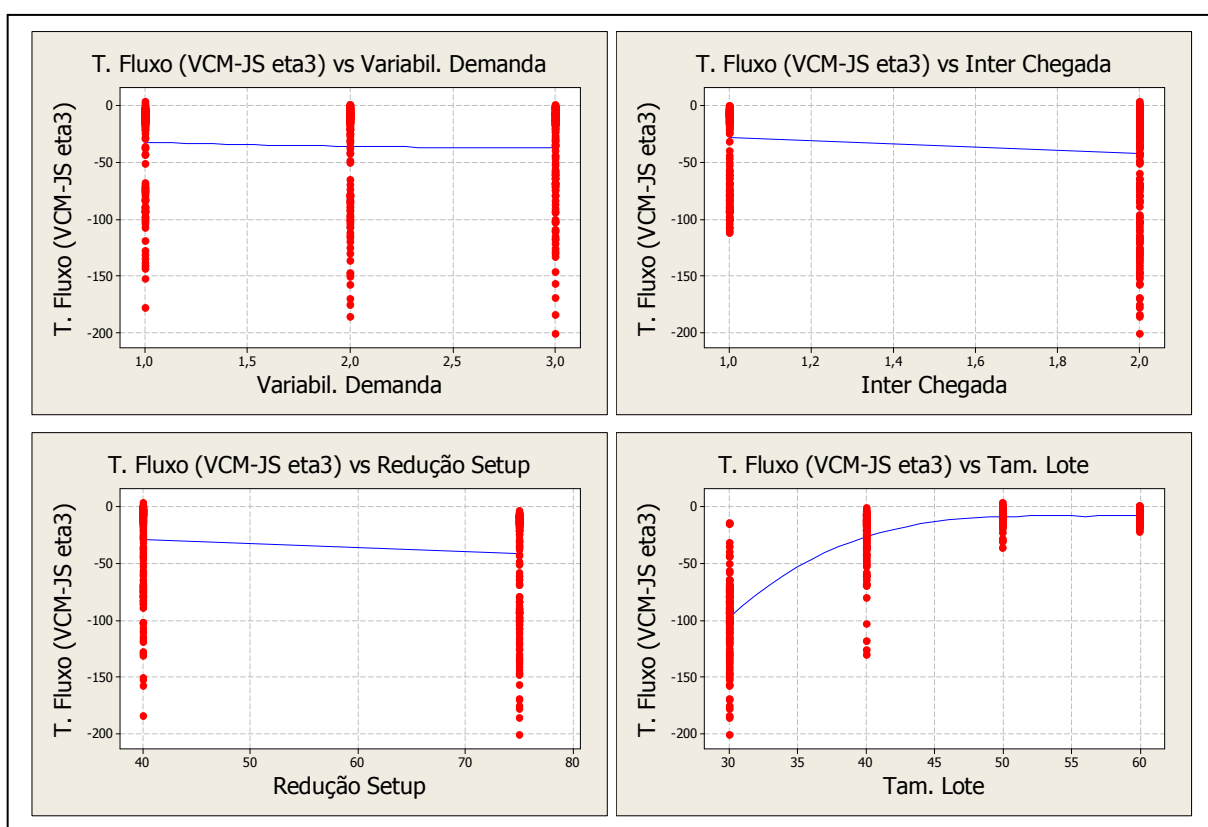


Figura A4.3 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para  $\eta = 3$

A tabela A4.3 e os gráficos da figura A4.3 são muito semelhantes aos obtidos em relação à diferença de tempo médio de fluxo entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 1$  e 2, portanto chega-se às mesmas conclusões anteriores.

Destacam-se, no gráfico da Figura A4.4 a seguir, as curvas de diferença de tempo médio de fluxo em função do tamanho de lote para  $\eta = 1, 2$  e 3. Percebe-se que descrevem uma variação de diferenças de tempo muito semelhantes. Entretanto, as pequenas variações entre elas indicam que a flexibilidade dos operadores produz ganhos um pouco diferenciados entre os dois sistemas. Mas, sem dúvida, prevalece a percepção de que, em geral, as três curvas descrevem realidades muito semelhantes, ou seja, que lotes menores geram vantagem competitiva das células virtuais sobre o *job shop*.

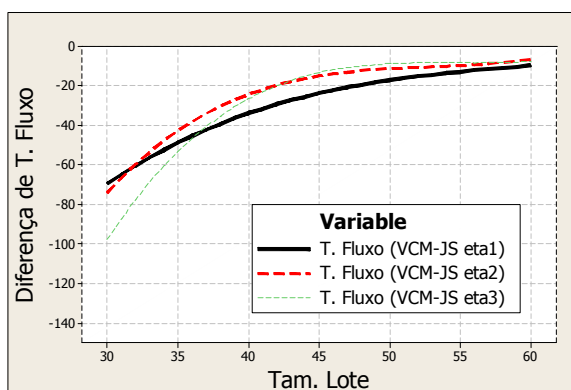


Figura A4.4 – “Scatterplots” com regressão do tempo médio de fluxo VCM-JS para  $\eta = 1, 2$  e 3

#### A4.1.4 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$

Conforme abordado anteriormente, dentro do padrão adotado o sistema celular tradicional não é viável para  $\eta = 1$ , limitando-se então a comparação entre esses dois sistemas para os outros níveis de flexibilidade de operador. A tabela A4.4 apresenta os valores da análise de variância da diferença de tempo médio de fluxo entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 2$ .

Tabela A4.4 Análise de variância tempo médio de fluxo VCM-CM para  $\eta = 2$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1650	825	10,23	0,000
Inter Chegada	1	23046	23046	285,59	0,000
Redução Setup	1	63627	63627	788,46	0,000
Tam. Lote	3	8477	2826	35,01	0,000
Error	472	38090	81		
Total	479	134890			

A figura A4.5 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de Fluxo entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 2$ .

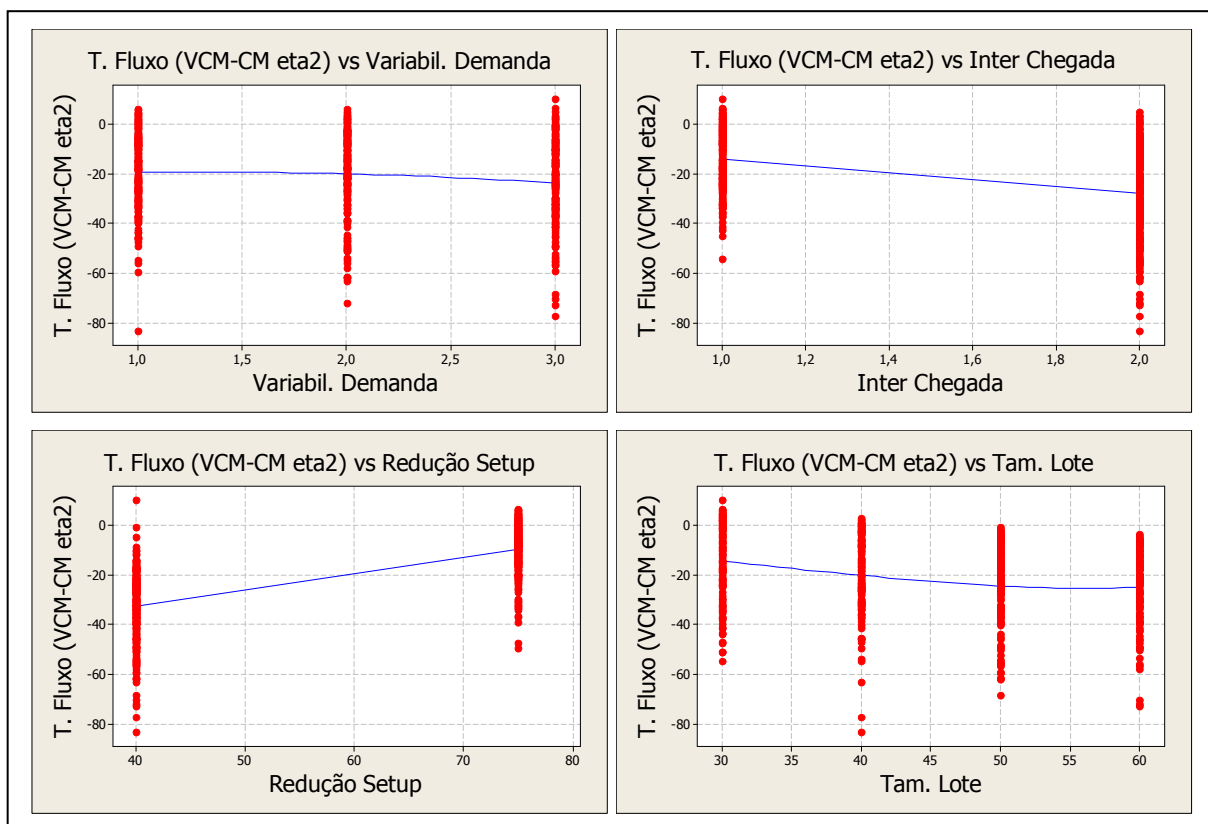


Figura A4.5 – “Scatterplots” com regressão do tempo médio de fluxo VCM-CM para  $\eta = 2$

Os dados da análise de variância da tabela A4.4 indicam que os quatro fatores influenciam significativamente na diferença de desempenho no tempo de fluxo entre o celular virtual e o celular tradicional. A ordem dessa influência é, em primeiro lugar, da redução de *setup*, seguido do intervalo entre chegadas, do tamanho do lote e, finalmente, da variabilidade de demanda. Por outro lado, os gráficos da figura A4.5 permitem verificar que, no escopo analisado, o celular virtual tem um desempenho superior ao celular tradicional em qualquer um dos níveis desses quatro fatores. Assim, o celular virtual tem sua vantagem aumentada quanto menor for a redução do *setup*; na medida em que for maior a variabilidade dos intervalos de chegadas (irregularidade na chegada de lotes); quanto maior for o tamanho dos lotes; e quanto maior for a variabilidade da demanda atual de peças (falta de previsibilidade). São esses os fatores que mostram a superioridade das células virtuais com relação às células tradicionais. Em outras palavras, o celular virtual é mais flexível que o tradicional e por isso menos sensível às limitações desse com relação à necessidade de uma demanda constante. Ou ainda, tem a vantagem da flexibilidade típica do *job shop*, a

sinergia do *pool* de máquinas. Mas, por outro lado, não consegue tirar tanta vantagem da redução de *setup* como o tradicional.

#### A4.1.5 Diferença de tempo médio de fluxo entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$

A tabela A4.5 apresenta os valores da análise de variância da diferença de tempo médio de fluxo entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 3$ .

Tabela A4.5 Análise de variância tempo médio de fluxo VCM-CM para  $\eta = 3$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	888,1	444,1	4,84	0,008
Inter Chegada	1	1516,9	1516,9	16,53	0,000
Redução Setup	1	7960,1	7960,1	86,73	0,000
Tam. Lote	3	25051,8	8350,6	90,98	0,000
Error	472	43321,2	91,8		
Total	479	78738,1			

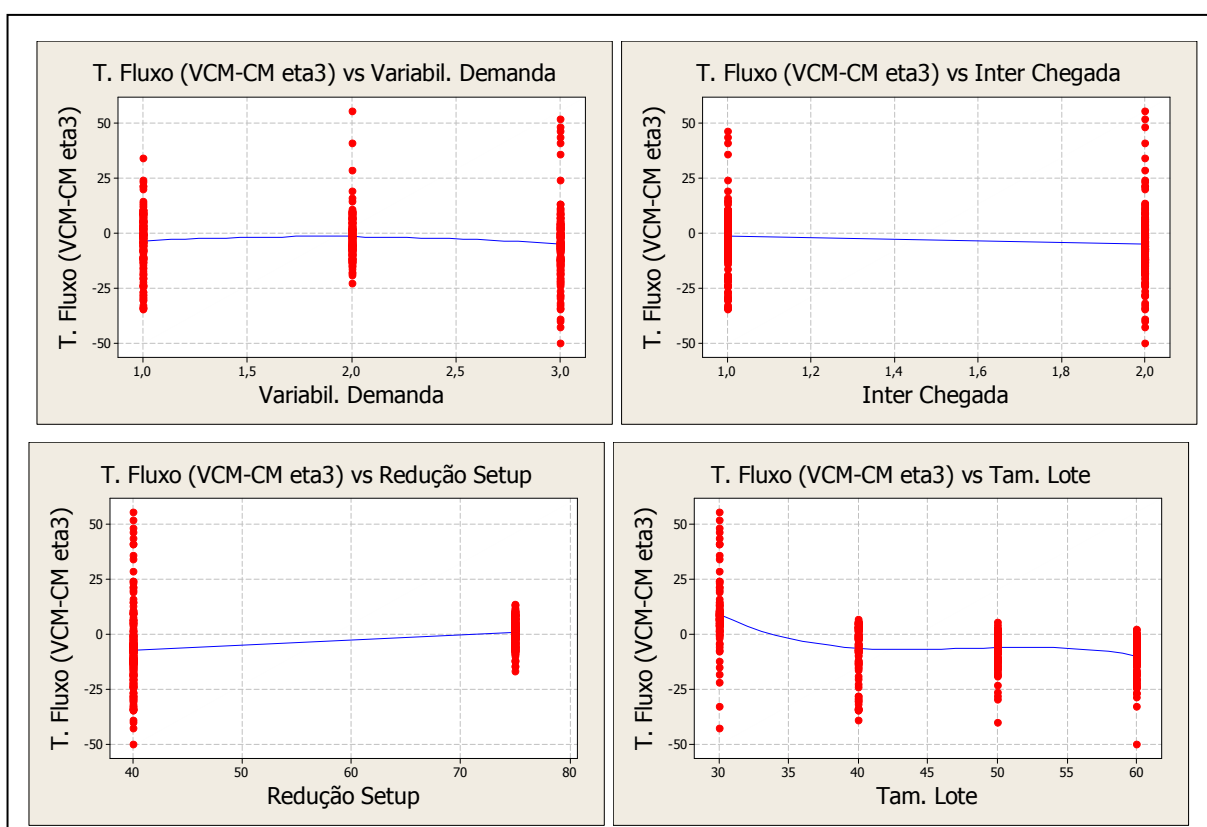


Figura A4.6 – “Scatterplots” com regressão do tempo médio de fluxo VCM-CM para  $\eta = 3$

A figura A4.6 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de Fluxo entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 3$ .

Pode-se observar, pela tabela A4.5 e pelos gráficos da figura A4.6, que o perfil da diferença de tempo médio de fluxo entre o celular virtual e o celular tradicional para  $\eta = 3$  é muito semelhante ao da mesma diferença para o caso de  $\eta = 2$ . Assim, aplicam-se aqui as afirmações anteriores, com algumas nuances. Por exemplo, a intensidade da diferença no fluxo de tempo entre os dois sistemas é bem menor, ou seja, a vantagem do celular virtual é menos acentuada. Percebe-se que a diferença média em torno de 20 horas com  $\eta = 2$  cai para cerca de 3 horas para  $\eta = 3$ . Por outro lado, a ordem de influência é, em primeiro lugar, do tamanho do lote, seguido da redução de setup, do intervalo entre chegadas e, finalmente, da variabilidade de demanda. Verifica-se também que, apesar de a vantagem do celular virtual permanecer dentro do escopo analisado, neste nível de flexibilidade, o celular tradicional chega a ser superior, ainda que com uma vantagem mínima, quando a redução do *setup* é muito alta e quando o tamanho de lote é muito pequeno. Assim, pode-se concluir que flexibilidade de operador gera baixa vantagem para o celular virtual.

## A4.2 Resultados envolvendo o tempo médio de filas

A seguir, apresentam-se os resultados obtidos com relação à diferença de desempenho no tempo médio de filas entre os três modelos avaliados, para  $\eta = 1, 2$  e  $3$ . Para cada um dos fatores é apresentada a análise de variância e os gráficos “Scatterplots” com regressão linear, quadrática ou cúbica, com as curvas passando pelos pontos médios dos pontos plotados.

### A4.2.1 Diferença de tempo médio de filas entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 1$

A tabela A4.6 apresenta os valores da análise de variância da diferença de tempo médio de filas entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 1$ .

Tabela A4.6 Análise de variância tempo médio de filas VCM-JS para  $\eta = 1$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	999	499	3,04	0,049
Inter Chegada	1	10771	10771	65,66	0,000
Redução Setup	1	21142	21142	128,89	0,000
Tam. Lote	3	264047	88016	536,57	0,000
Error	472	77424	164		
Total	479	374383			

A figura A4.7 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de filas entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para  $\eta = 1$ .

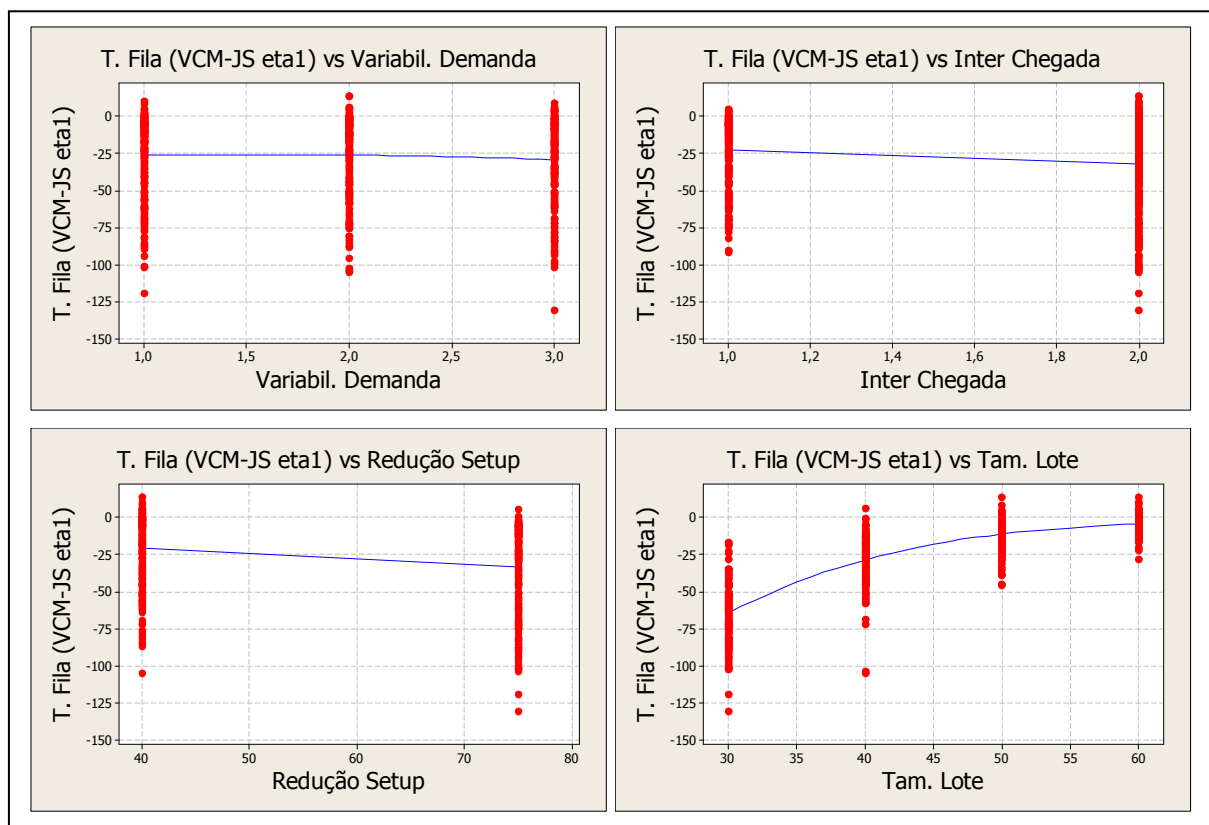


Figura A4.7 – “Scatterplots” com regressão do tempo médio de fila VCM-JS para  $\eta = 1$

Os resultados da diferença de tempo médio de filas entre o celular virtual e o *job shop* para  $\eta = 1$  apresentados na tabela A4.6 e nos gráficos da figura A4.7 são praticamente os mesmos da tabela A4.1 e gráficos da figura A4.1 referentes à diferença do tempo médio de fluxo. Logo, existe uma enorme correlação entre essas duas variáveis resposta. Assim, os comentários do subitem A4.1.1 valem também para o presente item.

O que explica que a diferença de tempo médio de fluxo seja equivalente à diferença do tempo média de fila? É que a maior parte, a quase totalidade, das diferenças de tempo médio de fluxo é causada pela diferença do tempo médio de fila entre os dois sistemas de configuração! De fato, o tempo médio de processo é o mesmo nos dois sistemas, as demais variações, ou seja, relativas ao *setup* e à movimentação, são relativamente pequenas.

#### A4.2.2 Diferença de tempo médio de filas entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 2$

A tabela A4.7 apresenta os valores da análise de variância da diferença de tempo médio de filas entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 2$ .

Tabela A4.7 Análise de variância tempo médio de filas VCM-JS para  $\eta = 2$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1368	684	2,85	0,059
Inter Chegada	1	12857	12857	53,60	0,000
Redução Setup	1	8051	8051	33,56	0,000
Tam. Lote	3	349205	116402	485,26	0,000
Error	472	113220	240		
Total	479	484701			

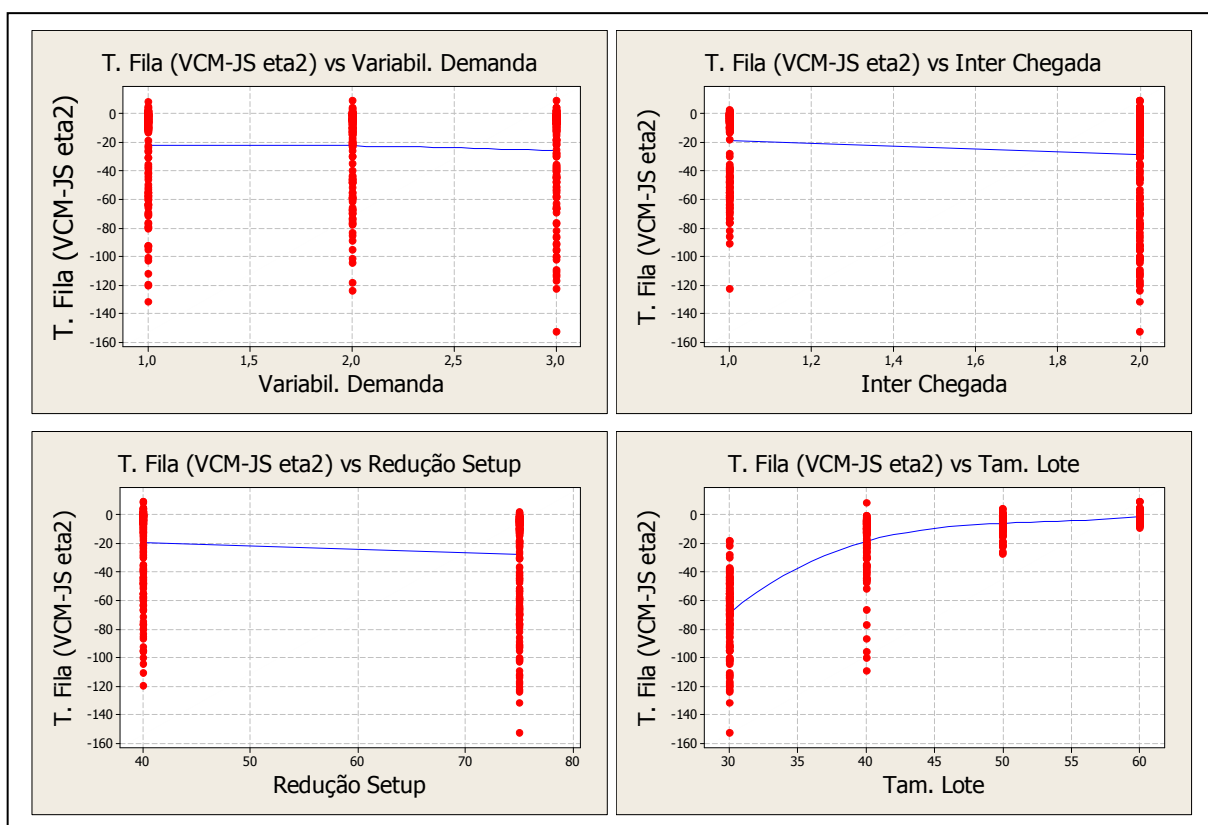


Figura A4.8 – “Scatterplots” com regressão do tempo médio de filas VCM-JS para  $\eta = 2$

A figura A4.8 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de filas entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para  $\eta = 2$ .

Da mesma forma que no subitem anterior, os resultados da tabela A4.7 e dos gráficos da figura A4.8 são equivalentes aos da tabela A4.2 e dos gráficos da figura A4.2. Assim, as conclusões são idênticas. A única diferença é na ordem de significância dos fatores. O mais



significativo é o tamanho do lote, seguido da intervalo entre chegadas e, logo a seguir, da redução de setup.

#### A4.2.3 Diferença de tempo médio de filas entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 3$

A tabela A4.8 apresenta os valores da análise de variância da diferença de tempo médio de filas entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 3$ .

Tabela A4.8 Análise de variância tempo médio de filas VCM-JS para  $\eta = 3$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1873	937	2,36	0,096
Inter Chegada	1	21432	21432	53,90	0,000
Redução Setup	1	10091	10091	25,38	0,000
Tam. Lote	3	670881	223627	562,43	0,000
Error	472	187672	398		
Total	479	891950			

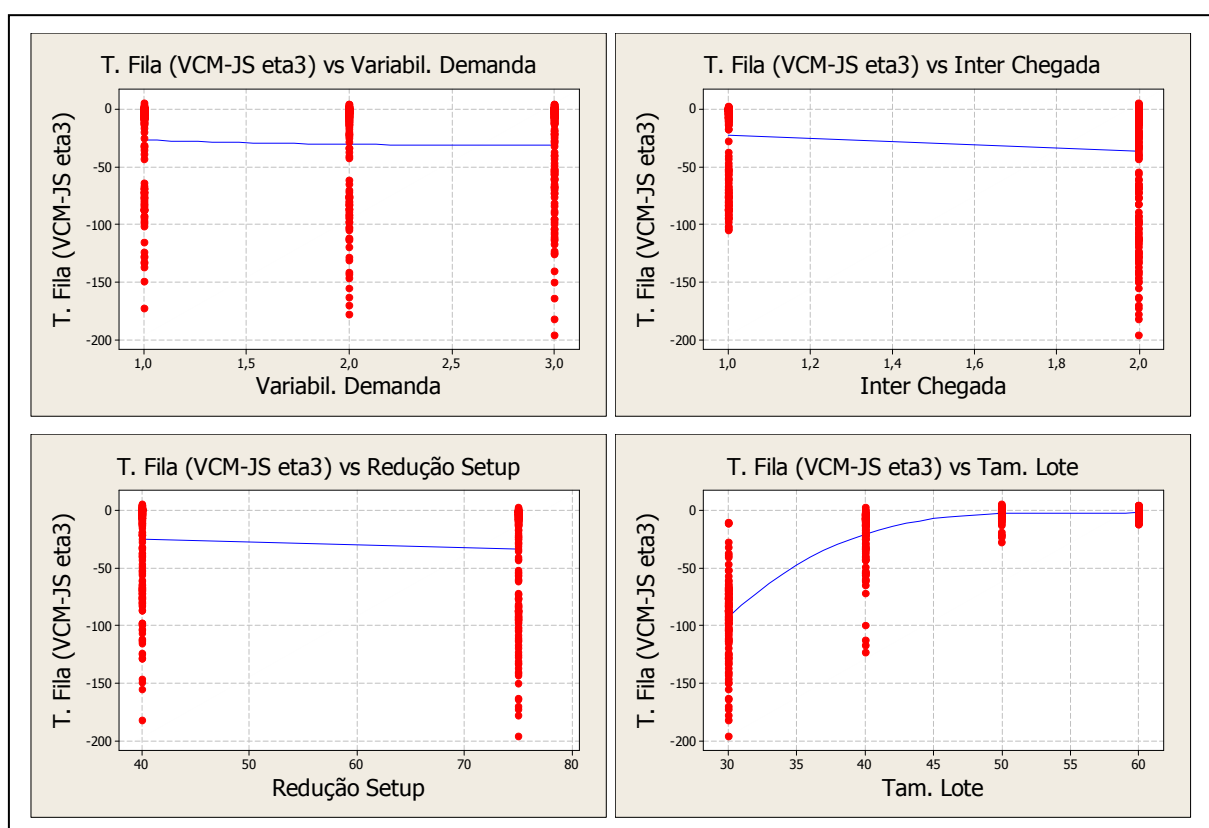


Figura A4.9 – “Scatterplots” com regressão do tempo médio de filas VCM-JS para  $\eta = 3$

A figura A4.9 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de filas entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para  $\eta = 3$ .

Igual aos subitens anteriores, os resultados da tabela A4.8 e dos gráficos da figura A4.9 são equivalentes aos da tabela 7.3 e dos gráficos da figura 7.3. Assim, as conclusões são as mesmas.

#### A4.2.4 Diferença de tempo médio de fila entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$

A tabela A4.9 apresenta os valores da análise de variância da diferença de tempo médio de fila entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 2$ .

Tabela A4.9 Análise de variância tempo médio de fila VCM-CM para  $\eta = 2$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	971	485	5,35	0,005
Inter Chegada	1	28136	28136	309,85	0,000
Redução Setup	1	44415	44415	489,12	0,000
Tam. Lote	3	11526	3842	42,31	0,000
Error	472	42861	91		
Total	479	127909			

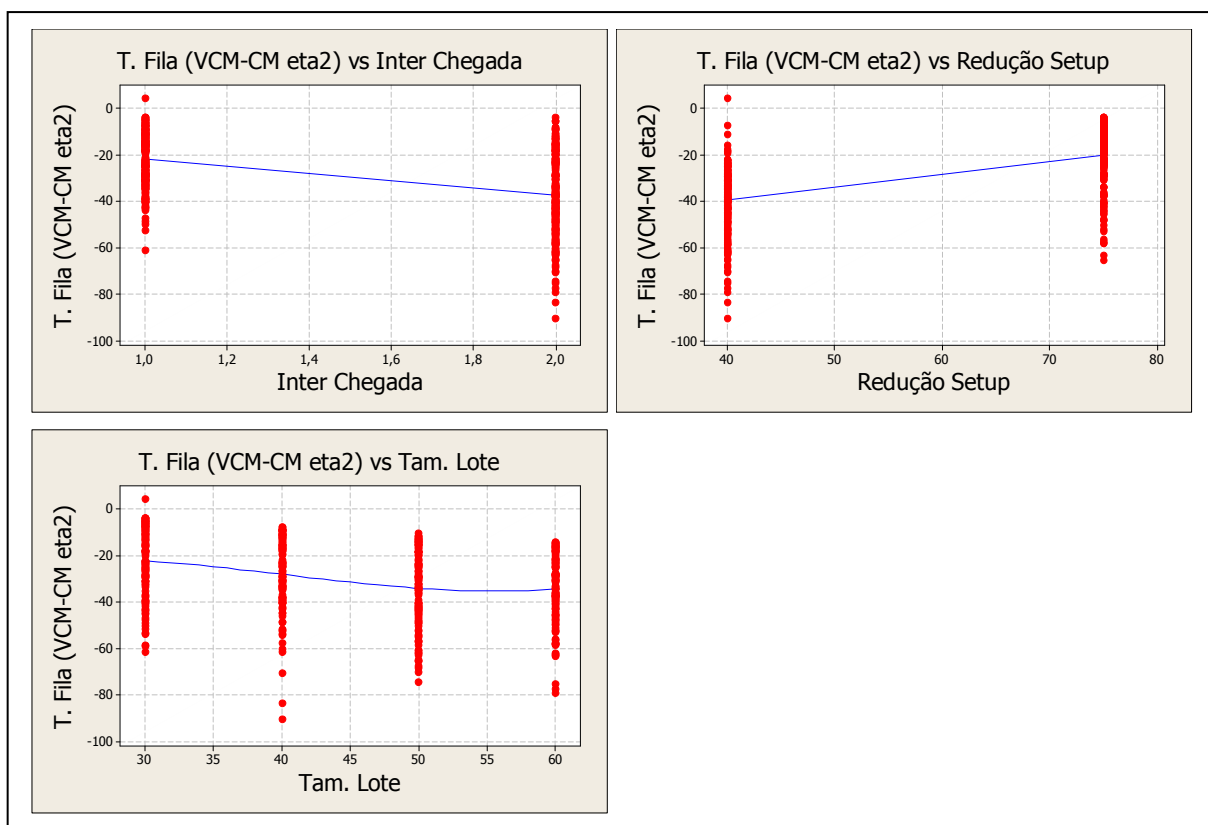


Figura A4.10 – “Scatterplots” com regressão do tempo médio de fila VCM-CM para  $\eta = 2$

A figura A4.10 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de fila entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 2$ .

Mais uma vez, igual aos subitens anteriores, os resultados da tabela A4.9 e dos gráficos da figura A4.10 são equivalentes aos da tabela A4.4 e dos gráficos da figura A4.5. Assim, as conclusões daqueles resultados valem para esses. Entretanto, os valores de tempo agora são maiores. No tempo médio de fluxo, a diferença de tempo era em torno de 20 h, agora é de 30 h. Por outro lado, há uma pequena diferença com relação ao fato da variabilidade de demanda, apesar de estatisticamente ser significativa a variação desse fator. Pelo gráfico, a vantagem do sistema celular virtual tende a ficar constante, tanto para uma maior como para uma menor variabilidade.

#### A4.2.5 Diferença de tempo médio de fila entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$

A tabela A4.10 apresenta os valores da análise de variância da diferença de tempo médio de fila entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 3$ .

Tabela A4.10 Análise de variância tempo médio de fila VCM-CM para  $\eta = 3$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	1052,3	526,1	7,55	0,001
Inter Chegada	1	4209,8	4209,8	60,41	0,000
Redução Setup	1	2895,7	2895,7	41,55	0,000
Tam. Lote	3	17259,4	5753,1	82,56	0,000
Error	472	32890,4	69,7		
Total	479	58307,5			

A figura A4.11 apresenta os gráficos “Scatterplots” com regressão para as diferenças de tempo médio de fila entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 3$ .

Também estes resultados são equivalentes aos obtidos com os dados referentes à diferença de tempo de fluxo apresentados na tabela A4.10 e nos gráficos da figura A4.11. Entretanto, aqui o celular virtual, apesar de ter uma vantagem bem menor que com  $\eta = 2$ , leva sempre vantagem sobre o celular tradicional e a diferença média de tempo de fila é em torno de 10 horas. Também há uma ordem diferente de significância dos fatores. O mais significativo é o tamanho do lote, seguido do intervalo entre chegadas e, logo a seguir, da redução de *setup* e da variabilidade de demanda. Por outro lado, o gráfico do fator variabilidade de demanda mostra a vantagem do sistema celular virtual, e tende a ficar com

diferença constante para variabilidade pequena (demanda constante), mas a diferença tende a crescer para uma maior variabilidade de demanda.

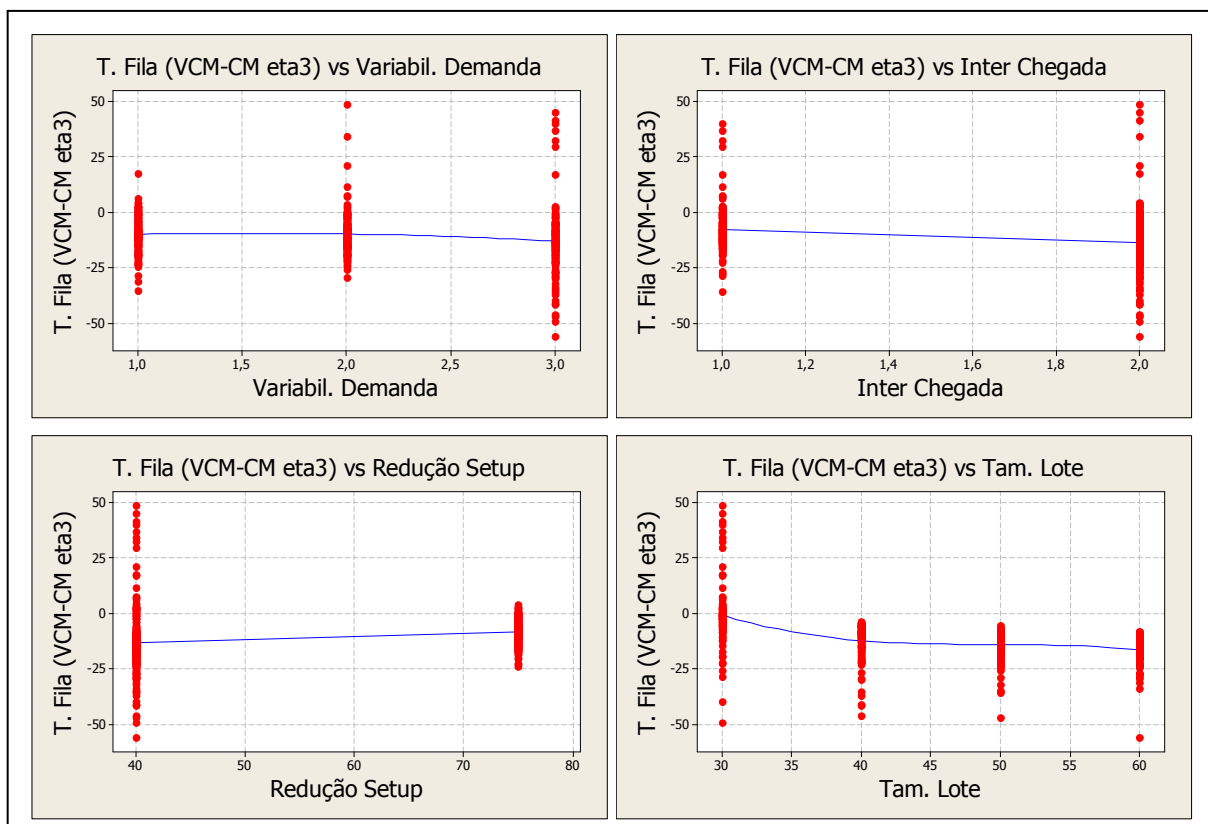


Figura A4.11 – “Scatterplots” com regressão do tempo médio de fila VCM-CM para  $\eta = 3$

### A4.3 Resultados envolvendo a ocupação dos operadores

A seguir, apresentam-se os resultados obtidos com relação à diferença de desempenho na ocupação dos operadores entre os três modelos avaliados, para  $\eta = 1, 2$  e  $3$ . Para cada um dos fatores é apresentada a análise de variância e os gráficos “Scatterplots” com regressão linear, quadrática ou cúbica, com as curvas passando pelos pontos médios dos pontos plotados.

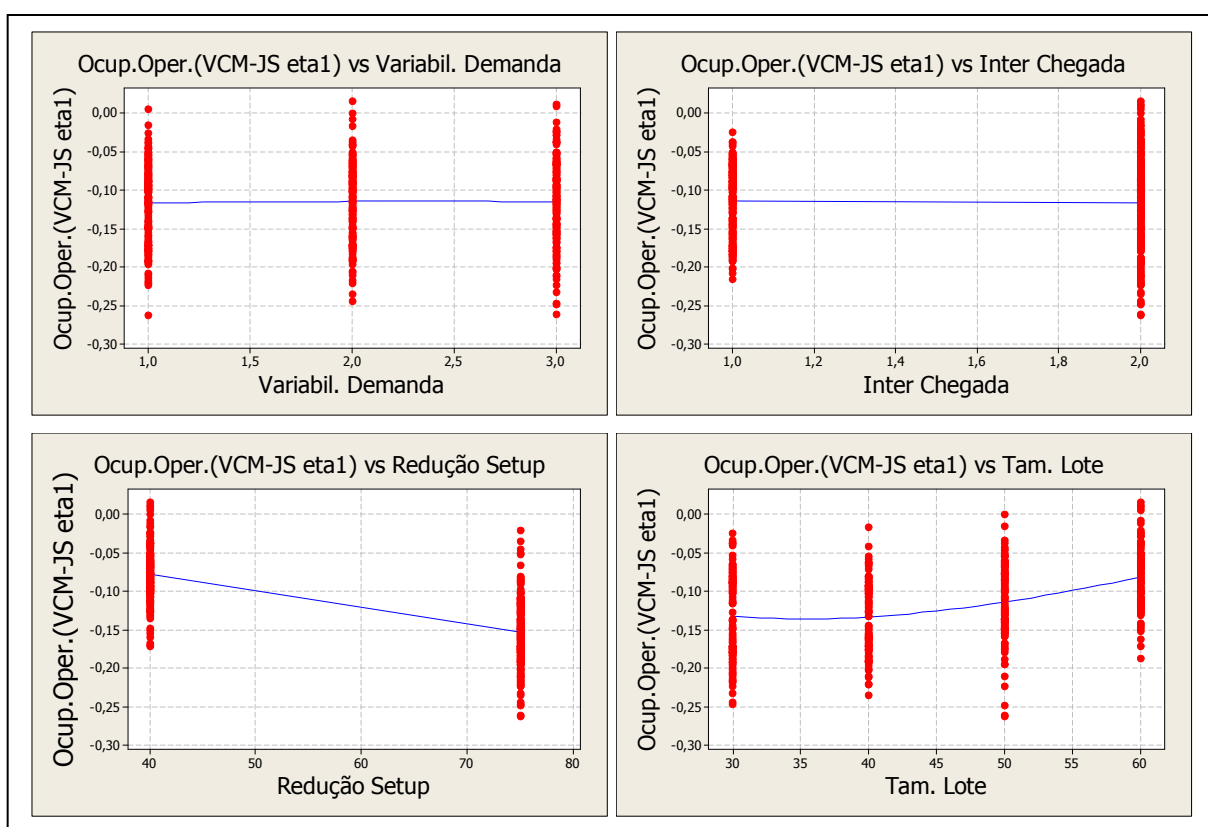
#### A4.3.1 Diferença de ocupação dos operadores entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 1$

A tabela A4.11 apresenta os valores da análise de variância da diferença de ocupação dos operadores entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 1$ .

Tabela A4.11 Análise de variância da ocupação dos operadores VCM-JS para eta ( $\eta$ ) = 1

Source	DF	SS	MS	F	P
Variabil. Demanda	2	0,00040	0,00020	0,23	0,795
Inter Chegada	1	0,00136	0,00136	1,58	0,209
Redução Setup	1	0,68071	0,68071	790,34	0,000
Tam. Lote	3	0,20866	0,06955	80,75	0,000
Error	472	0,40653	0,00086		
Total	479	1,29765			

A figura A4.12 apresenta os gráficos “Scatterplots” com regressão para as diferenças de ocupação dos operadores entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para eta ( $\eta$ ) = 1.

Figura A4.12 – “Scatterplots” com regressão da ocupação dos operadores VCM-JS para  $\eta = 1$ 

Os dados da análise de variância da tabela A4.11 indicam que os fatores que influenciam na diferença de ocupação dos operadores entre o celular virtual e o *job shop* para eta ( $\eta$ ) = 1 são apenas a redução de *setup* e o tamanho do lote, nessa ordem de significação. Por outro lado, os correspondentes gráficos da figura A4.12 permitem verificar que no escopo analisado o celular virtual tem uma menor ocupação dos operadores do que

o *job shop* em qualquer dos níveis dos dois fatores que são significativos. Os valores da diferença VCM-JS sendo negativos, indicam de fato que o celular virtual realiza suas tarefas com menor ocupação dos seus operadores do que o *job shop*. No item A4.4, verifica-se que a produtividade do celular virtual é superior à dos demais sistemas no escopo avaliado. Assim, esta baixa ocupação é eficaz, pois produz mais com menor emprego de recursos. Por isso, está se considerando aqui a menor ocupação dos operadores como desempenho superior. Por outro lado, pode-se verificar que a vantagem das células virtuais aumenta na medida em que é menor o tamanho de lotes e é maior a redução de *setup*.

#### A4.3.2 Diferença de ocupação dos operadores entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 2$

A tabela A4.12 apresenta os valores da análise de variância da diferença de ocupação dos operadores entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 2$ .

Tabela A4.12 Análise de variância da ocupação dos operadores VCM-JS para  $\eta = 2$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	0,01592	0,00796	4,06	0,018
Inter Chegada	1	0,00081	0,00081	0,41	0,520
Redução Setup	1	0,89757	0,89757	457,13	0,000
Tam. Lote	3	0,08132	0,02711	13,81	0,000
Error	472	0,92677	0,00196		
Total	479	1,92239			

A figura A4.13 apresenta os gráficos “Scatterplots” com regressão para as diferenças de ocupação dos operadores entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para  $\eta = 2$ .

Pelos dados da tabela A4.12 e dos gráficos da figura A4.13, pode-se comprovar que apenas o fator redução de *setup* e o tamanho do lote geram diferenças significativas na variável resposta diferença de ocupação entre esses dois sistemas. Dentro do escopo avaliado e para este nível de flexibilidade dos operadores, os resultados indicam um desempenho médio superior do celular virtual sobre o *job shop* em cerca de 10%. O fator tamanho do lote, apesar de significativo, indica que o desempenho maior do celular virtual é justamente no intervalo de tamanhos de lote do escopo da pesquisa e valores superiores ou inferiores aos limites desse intervalo tendem a reduzir a vantagem do celular virtual sobre o *job shop*. E, por último, verifica-se que quanto maior a redução do *setup*, maior a vantagem do celular virtual sobre o *job shop*.

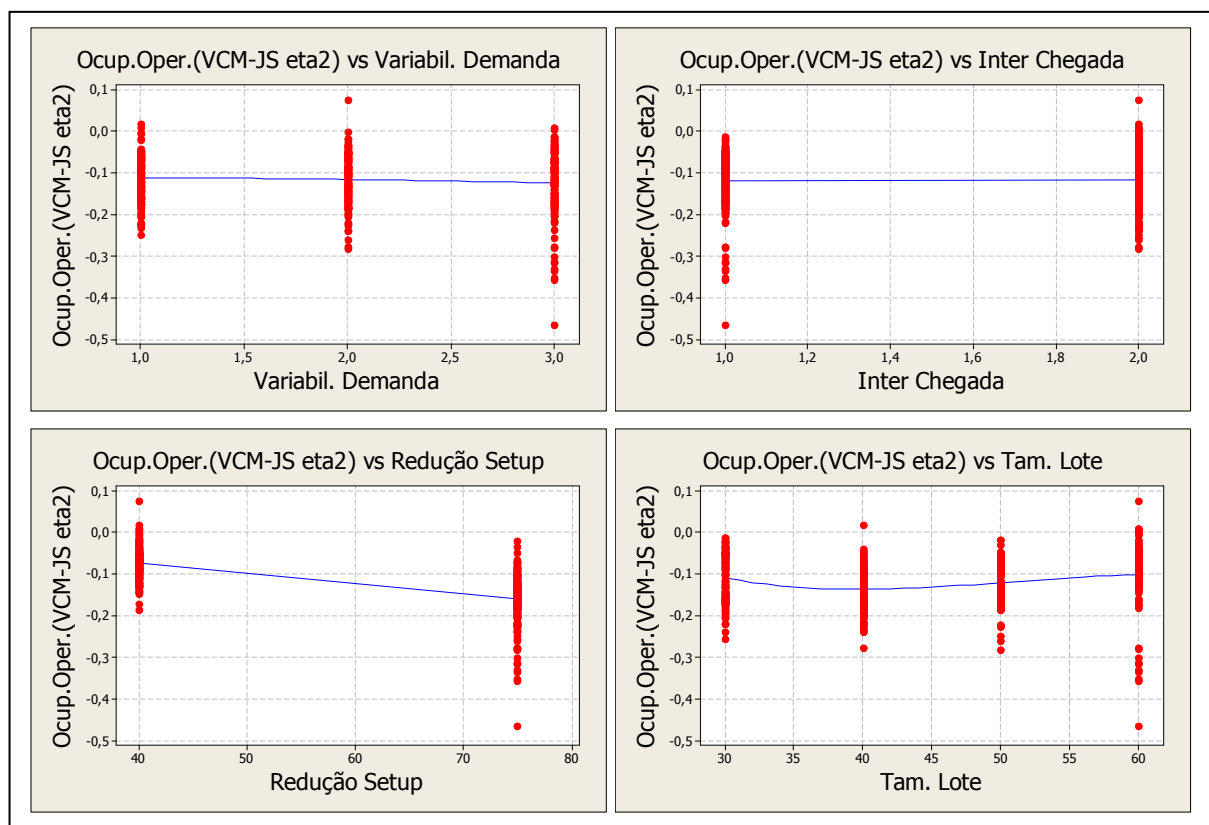


Figura A4.13 – “Scatterplots” com regressão da ocupação dos operadores VCM-JS para  $\eta = 2$

#### A4.3.3 Diferença de ocupação dos operadores entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 3

A tabela A4.13 apresenta os valores da análise de variância da diferença de ocupação dos operadores entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 3.

Tabela A4.13 Análise de variância da ocupação dos operadores VCM-JS para eta ( $\eta$ ) = 3

Source	DF	SS	MS	F	P
Variabil. Demanda	2	0,00050	0,00025	0,22	0,802
Inter Chegada	1	0,00293	0,00293	2,59	0,108
Redução Setup	1	0,78983	0,78983	698,16	0,000
Tam. Lote	3	0,20041	0,06680	59,05	0,000
Error	472	0,53398	0,00113		
Total	479	1,52765			

A figura A4.14 apresenta os gráficos “Scatterplots” com regressão para as diferenças de ocupação dos operadores entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para eta ( $\eta$ ) = 3. Através desses gráficos e tabela, conclui-se que os resultados são idênticos aos do subitem anterior, ou seja, para eta ( $\eta$ ) = 2, portanto chega-se às mesmas conclusões. O gráfico da figura A4.15 permite comprovar que são idênticos

os resultados da variável resposta para o fator redução de *setup* para  $\eta = 1, 2$  e  $3$ . Por outro lado, verifica-se que, para a flexibilidade nula dos operadores ( $\eta = 1$ ), a vantagem do celular virtual é menor para reduções de *setup* menores, mas, em compensação, maior para reduções maiores, em comparação com a flexibilidade dupla e tripla.

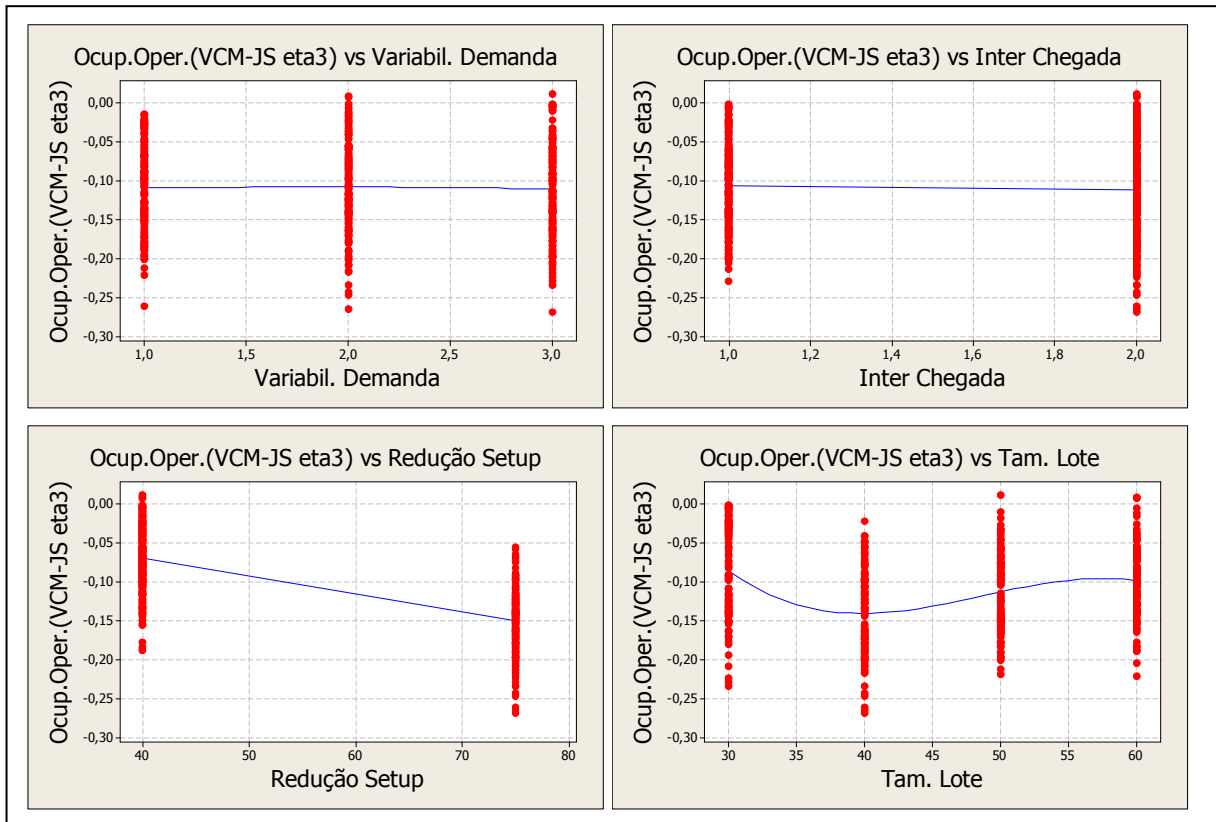


Figura A4.14 – “Scatterplots” com regressão da ocupação dos operadores VCM-JS para  $\eta = 3$

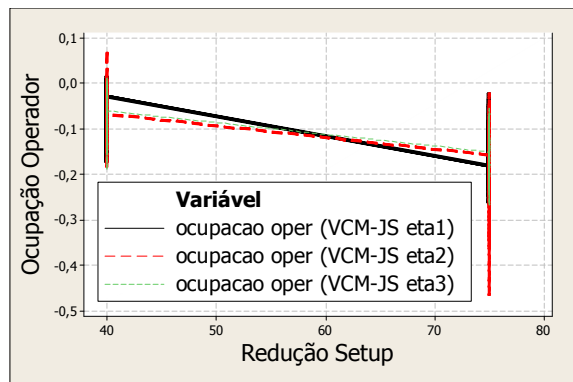


Figura A4.15 – Scatterplot com regressão da ocupação dos operadores (VCM-JS  $\eta = 1, 2$  e  $3$ )



#### A4.3.4 Diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 2$

A tabela A4.14 apresenta os valores da análise de variância da diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 2$ .

Tabela A4.14 Análise de variância da ocupação dos operadores VCM-CM para  $\eta = 2$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	0,01797	0,00898	5,25	0,006
Inter Chegada	1	0,00102	0,00102	0,60	0,441
Redução Setup	1	0,09728	0,09728	56,88	0,000
Tam. Lote	3	0,73350	0,24450	142,96	0,000
Error	472	0,80726	0,00171		
Total	479	1,65703			

A figura A4.16 apresenta os gráficos “Scatterplots” com regressão para as diferenças de ocupação dos operadores entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 2$ .

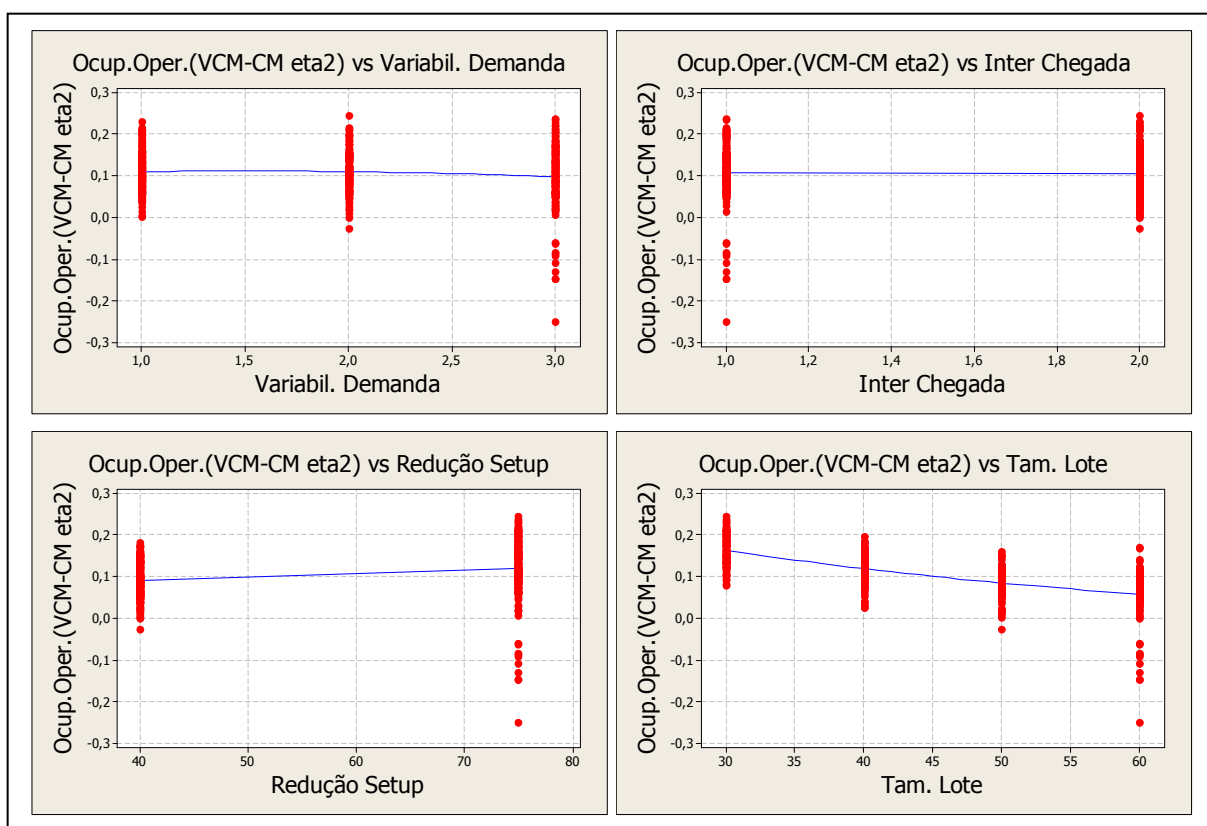


Figura A4.16 – “Scatterplots” com regressão da ocupação dos operadores VCM-CM para  $\eta = 2$

Pelos dados da tabela A4.14 e dos gráficos da figura A4.16, pode-se comprovar que os fatores tamanho do lote, redução de *setup* e variabilidade de demanda geram diferenças significativas na variável resposta diferença de ocupação dos operadores entre esses dois sistemas, nessa ordem. Dentro do escopo avaliado e para este nível de flexibilidade dos operadores, os resultados indicam um desempenho médio inferior do celular virtual sobre o celular tradicional em cerca de 10%, isto é, o celular tradicional ocupa menos os operadores. Entretanto, como a produtividade do celular virtual é um pouco maior, como se pode verificar no subitem A4.4.4, este fato comprova que o celular tradicional se beneficia mais da redução de *setup*, requerendo menos tempo de operador. Mas não consegue tirar proveito desse fato por ter problemas de sequenciamento (*schedule*) justamente por ter restrições de capacidade, o que comprova o subitem A4.2.4 com relação ao tempo de fila. O fator tamanho do lote é responsável pela queda do desempenho do celular tradicional, na medida em que cresce o tamanho de lote. Por outro lado, verifica-se que quanto maior a redução do *setup*, maior a vantagem do celular tradicional. Finalmente, o aumento da variabilidade de demanda gera vantagem do celular virtual sobre o celular tradicional, uma vez que a diferença de ocupação diminui, ainda que esta redução seja muito pequena.

#### A4.3.5 Diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$

A tabela A4.15 apresenta os valores da análise de variância da diferença de ocupação dos operadores entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 3$ . Por esses dados, pode-se comprovar que apenas o fator redução de *setup* e o fator tamanho do lote geram diferenças significativas na variável resposta diferença de ocupação dos operadores entre esses dois sistemas, nessa ordem.

Tabela A4.15 Análise de variância da ocupação dos operadores VCM-CM para  $\eta = 3$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	0,00076	0,00038	0,40	0,671
Inter Chegada	1	0,00254	0,00254	2,69	0,102
Redução Setup	1	0,24323	0,24323	256,85	0,000
Tam. Lote	3	0,64318	0,21439	226,40	0,000
Error	472	0,44697	0,00095		
Total	479	1,33668			

A figura A4.17 apresenta os gráficos “Scatterplots” com regressão para as diferenças de ocupação dos operadores entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 3$ . Pode-se comprovar que o desempenho do celular tradicional é ainda mais superior do que no subitem anterior, ou seja, para  $\eta =$

2, com um desempenho médio superior em cerca de 12,5%. Entretanto, algumas conclusões são idênticas. Assim, como a produtividade do celular virtual é idêntica ao do celular tradicional, como se pode verificar no subitem A4.4.4, este fato comprova mais uma vez que o celular tradicional tira mais partido da redução de *setup*, requerendo menos tempo de operador. Entretanto, não consegue tirar proveito desse fato por ter problemas de sequenciamento (*schedule*) justamente por ter restrições de capacidade, o que comprova o subitem A4.2.5 com relação ao tempo de fila. O fator tamanho do lote é responsável pela queda do desempenho do celular tradicional, na medida em que cresce o tamanho de lote. Por outro lado, verifica-se que quanto maior a redução do *setup*, maior fica a vantagem do celular tradicional.

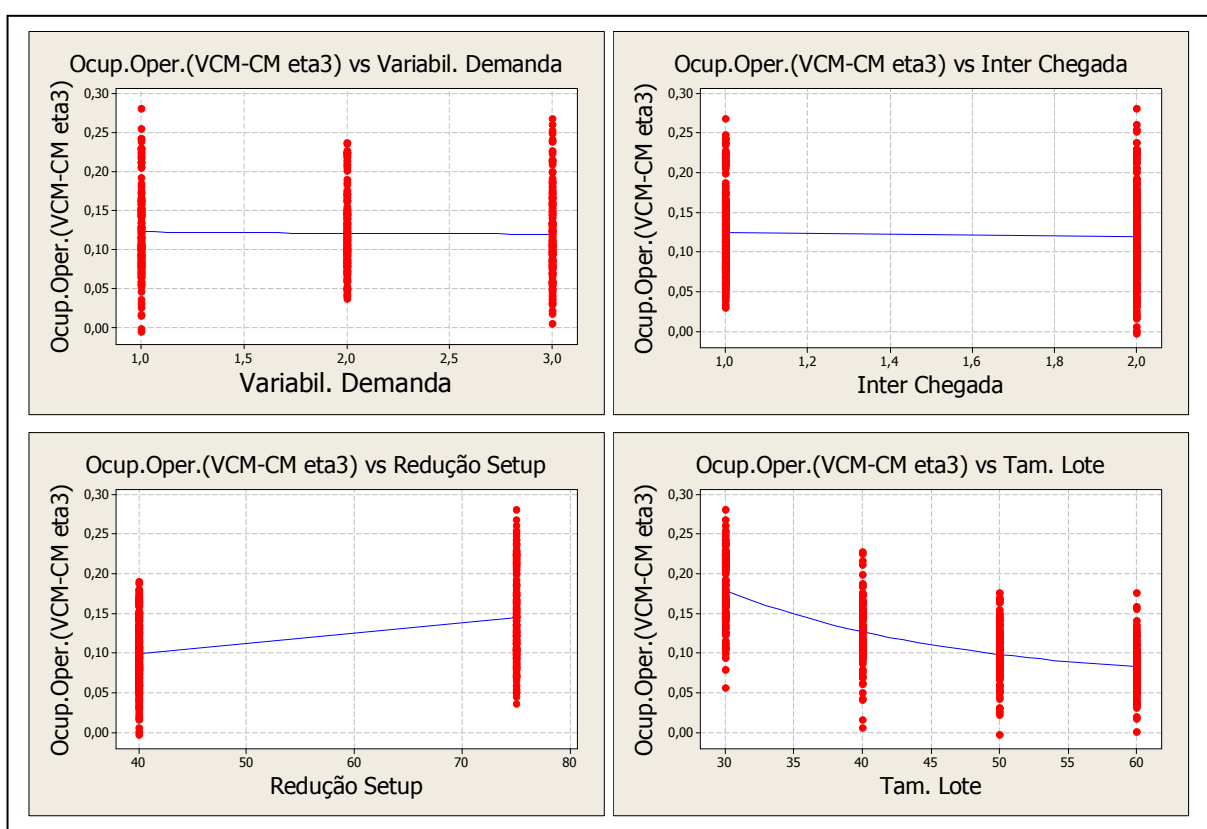


Figura A4.17–“Scatterplots” com regressão da ocupação dos operadores VCM-CM para  $\eta = 3$

#### A4.4 Resultados envolvendo a produtividade

A seguir, apresentam-se os resultados obtidos com relação à diferença de desempenho na produtividade entre os três modelos avaliados, para  $\eta$  ( $\eta$ ) = 1, 2 e 3. Para cada um dos fatores é apresentado a análise de variância e os gráficos “Scatterplots” com regressão linear, quadrática ou cúbica, com as curvas passando pelos pontos médios dos pontos plotados.

#### A4.4.1 Diferença de produtividade entre o celular virtual (VCM) e o *job shop* (JS) para $\eta = 1$

A tabela A4.16 apresenta os valores da análise de variância da diferença de produtividade entre o celular virtual (VCM) e o *job shop* (JS) para  $\eta = 1$ . Por esses dados, pode-se comprovar que apenas o fator tamanho do lote gera diferenças significativas na variável resposta diferença de produtividade entre esses dois sistemas.

Tabela A4.16 Análise de variância da diferença de produtividade VCM-JS para  $\eta = 1$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	14963632	7481816	3,60	0,028
Inter Chegada	1	7390403	7390403	3,55	0,060
Redução Setup	1	270750	270750	0,13	0,718
Tam. Lote	3	306903583	102301194	49,19	0,000
Error	472	981682268	2079835		
Total	479	1311210637			

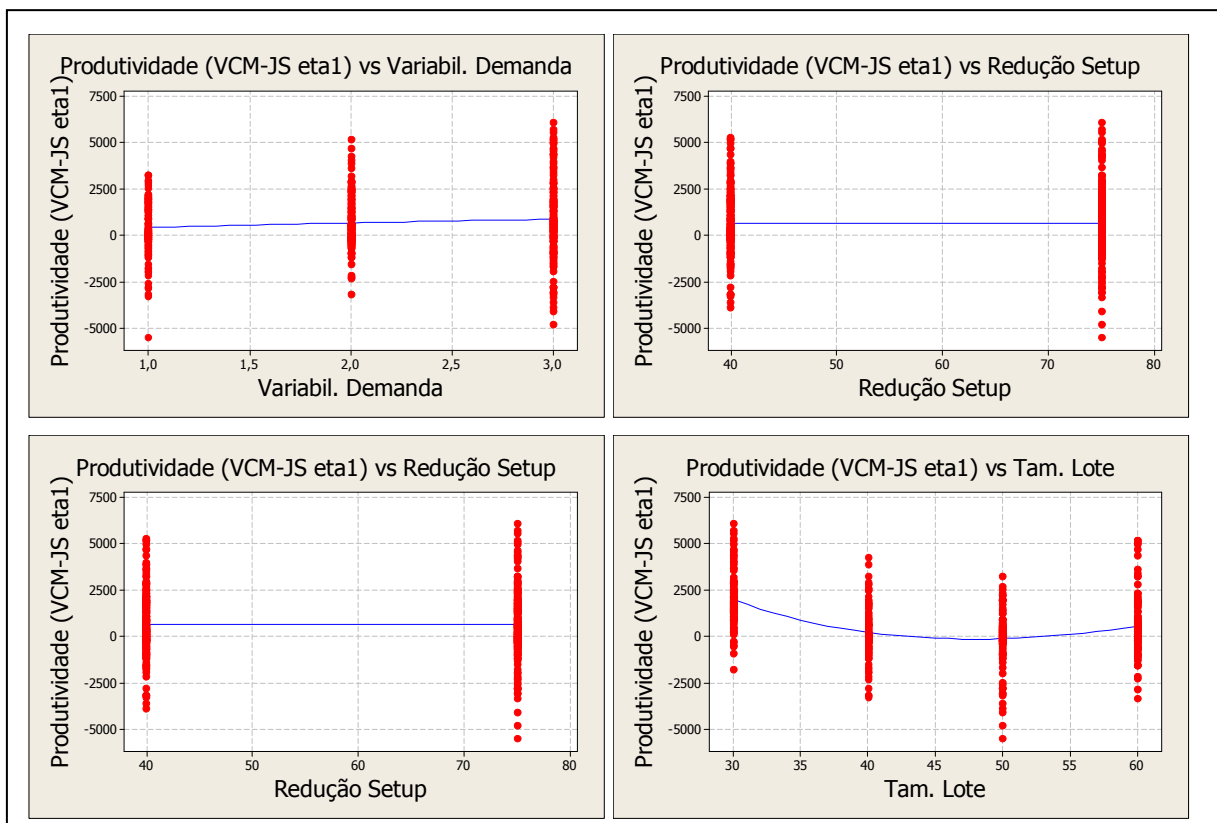


Figura A4.18 – “Scatterplots” com regressão da produtividade VCM-JS para  $\eta = 1$

A figura A4.18 apresenta os gráficos “Scatterplots” com regressão para as diferenças de produtividade entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para  $\eta = 1$ . Pode-se comprovar que a produtividade do celular virtual é superior em número de peças produzidas por hora, em cerca de 400 peças por hora. A análise do gráfico do único fator que é significativo permite concluir que este desempenho superior deve-se principalmente à diferença de produtividade, muito superior para o menor tamanho de lote dentro do escopo adotado, ou seja, lote de quarenta peças. De fato, para lotes menores o tempo de *setup* impacta bastante na diferença de tempo total de fluxo, conforme o subitem A4.1.1, na diferença de tempo de fila, conforme o subitem A4.2.1, e na produtividade.

#### A4.4.2 Diferença de produtividade entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 2

A tabela A4.17 apresenta os valores da análise de variância da diferença de produtividade entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 2. Igual que para eta ( $\eta$ ) = 1, pode-se comprovar que apenas o fator tamanho do lote gera diferenças significativas na variável resposta diferença de ocupação dos operadores entre esses dois sistemas.

Tabela A4.17 Análise de variância da produtividade VCM-JS para eta ( $\eta$ ) = 2

Source	DF	SS	MS	F	P
Variável Demanda	2	13169062	6584531	3,06	0,048
Inter Chegada	1	2901	2901	0,00	0,971
Redução Setup	1	2041021	2041021	0,95	0,330
Tam. Lote	3	1337009676	445669892	207,23	0,000
Error	472	1015069840	2150572		
Total	479	2367292499			

A figura A4.19 apresenta os gráficos “Scatterplots” com regressão para as diferenças de produtividade entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para eta ( $\eta$ ) = 2. Da mesma forma que para eta ( $\eta$ ) = 1, pode-se verificar que a produtividade do celular virtual é superior em número de peças produzidas por hora, nesse caso, em cerca de 1.000 peças por hora. Novamente, a análise do gráfico do único fator que é significativo permite concluir que este desempenho superior deve-se principalmente à produtividade muito superior para o menor tamanho de lote dentro do escopo adotado. A conclusão, novamente, é que para lotes menores o tempo de *setup* impacta bastante no tempo total de fluxo, conforme o item 7.22, nos tempos de fila, conforme o item 7.32, e na produtividade.

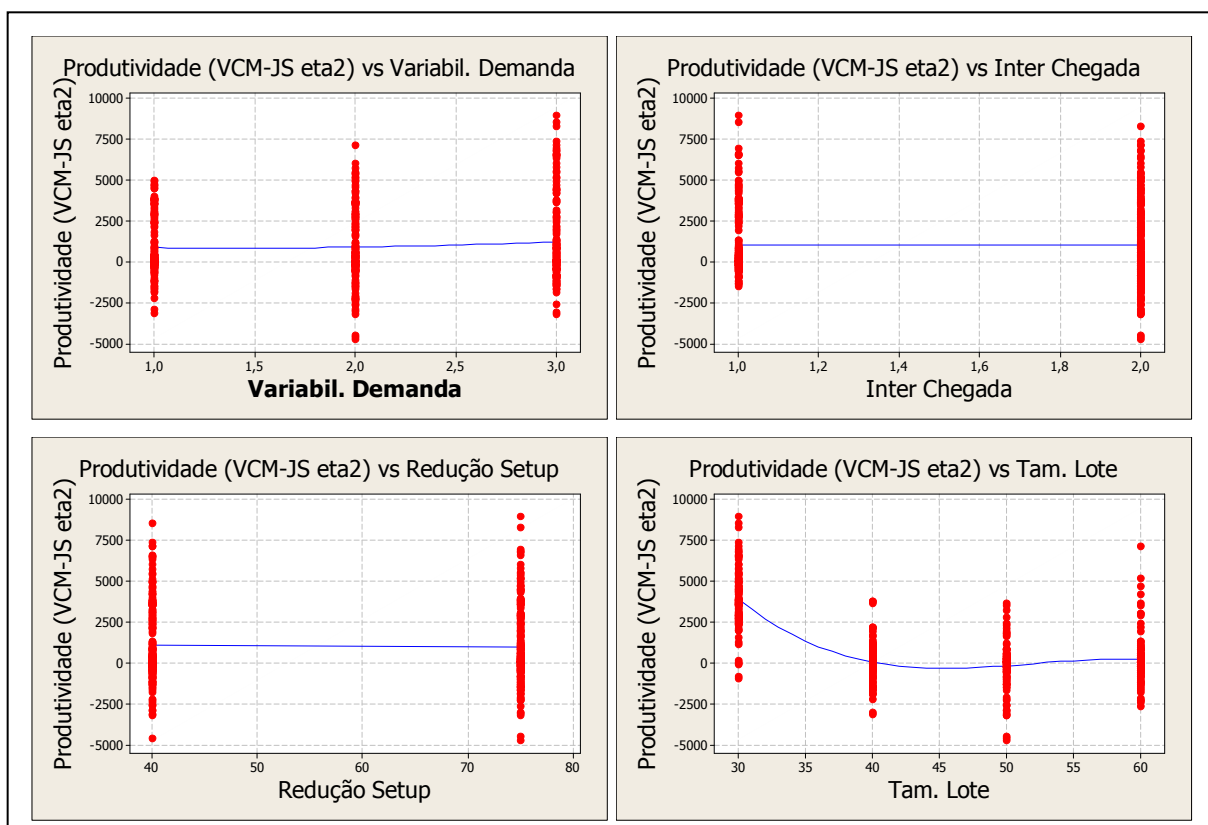


Figura A4.19 – “Scatterplots” com regressão da produtividade VCM-JS para eta ( $\eta$ ) = 2

#### A4.4.3 Diferença de produtividade entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 3

A tabela A4.18 apresenta os valores da análise de variância da diferença de produtividade entre o celular virtual (VCM) e o *job shop* (JS) para eta ( $\eta$ ) = 3. Igual que para eta ( $\eta$ ) = 1 e 2, pode-se comprovar que apenas o fator tamanho do lote gera diferenças significativas na variável resposta diferença de ocupação dos operadores entre esses dois sistemas.

Tabela A4.18 Análise de variância da produtividade VCM-JS para eta ( $\eta$ ) = 3

Source	DF	SS	MS	F	P
Variabil. Demanda	2	7198152	3599076	1,87	0,155
Inter Chegada	1	23520	23520	0,01	0,912
Redução Setup	1	444083	444083	0,23	0,631
Tam. Lote	3	1636707180	545569060	284,11	0,000
Error	472	906384452	1920306		
Total	479	2550757387			

A figura A4.20 apresenta os gráficos “Scatterplots” com regressão para as diferenças de produtividade entre a célula virtual e o *job shop* versus cada um dos quatro fatores do experimento, para eta ( $\eta$ ) = 3. Da mesma forma que para eta ( $\eta$ ) = 1 e 2, pode-se verificar que a produtividade do celular virtual é superior em número de peças produzidas por hora, nesse caso, também em cerca de 1.000 peças por hora. Novamente, a análise do gráfico do único fator que é significativo permite concluir que este desempenho superior se deve principalmente à produtividade muito superior no menor tamanho de lote dentro do escopo adotado. A conclusão, novamente, é que para lotes menores o tempo de *setup* impacta bastante no tempo total de fluxo (conforme o subitem A4.1.3), nos tempos de fila (conforme o subitem A4.2.3) e na produtividade.

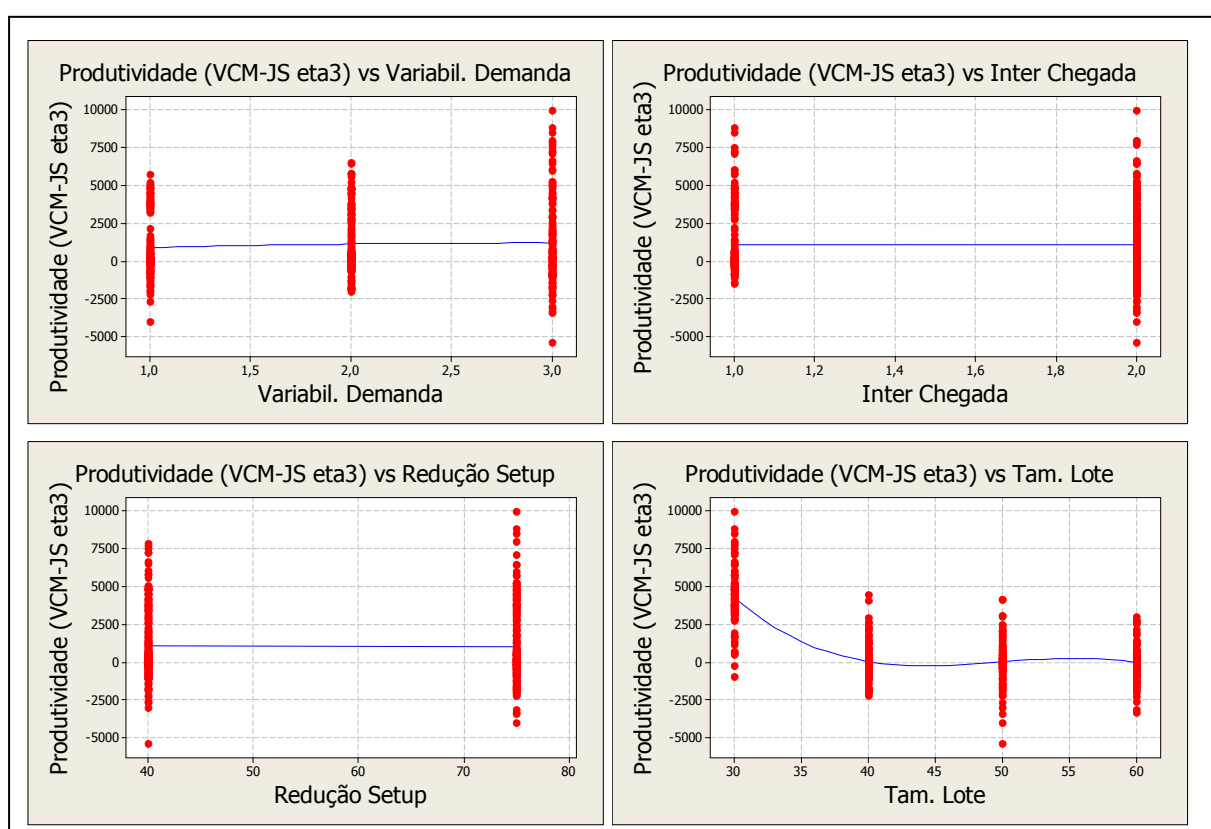


Figura A4.20 – “Scatterplots” com regressão da produtividade VCM-JS para eta ( $\eta$ ) = 3

#### A4.4.4 Diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para eta ( $\eta$ ) = 2

A tabela A4.19 apresenta os valores da análise de variância da diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para eta ( $\eta$ ) = 2. Pode-se comprovar que apenas os fatores redução de *setup* e tamanho do lote geram

diferenças significativas, nesta ordem de significância, na variável resposta diferença de produtividade entre esses dois sistemas.

Tabela A4.19 Análise de variância da produtividade VCM-CM para eta ( $\eta$ ) = 2

Source	DF	SS	MS	F	P
Variabil. Demanda	2	3009695	1504848	0,87	0,419
Inter Chegada	1	3207870	3207870	1,86	0,174
Redução Setup	1	96373763	96373763	55,75	0,000
Tam. Lote	3	49969987	16656662	9,64	0,000
Error	472	815974565	1728760		
Total	479	968535880			

A figura A4.21 apresenta os gráficos “Scatterplots” com regressão para as diferenças de produtividade entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para eta ( $\eta$ ) = 2. Pode-se verificar que a produtividade do celular virtual é ligeiramente superior em número de peças produzidas por hora, nesse caso cerca de 150 peças por hora. Apesar de a variação ser muito pequena, pode-se perceber pelos gráficos que quanto maiores forem os tamanhos de lote ou maior a redução de *setup*, menor ou nula é a vantagem do celular virtual sobre o tradicional.

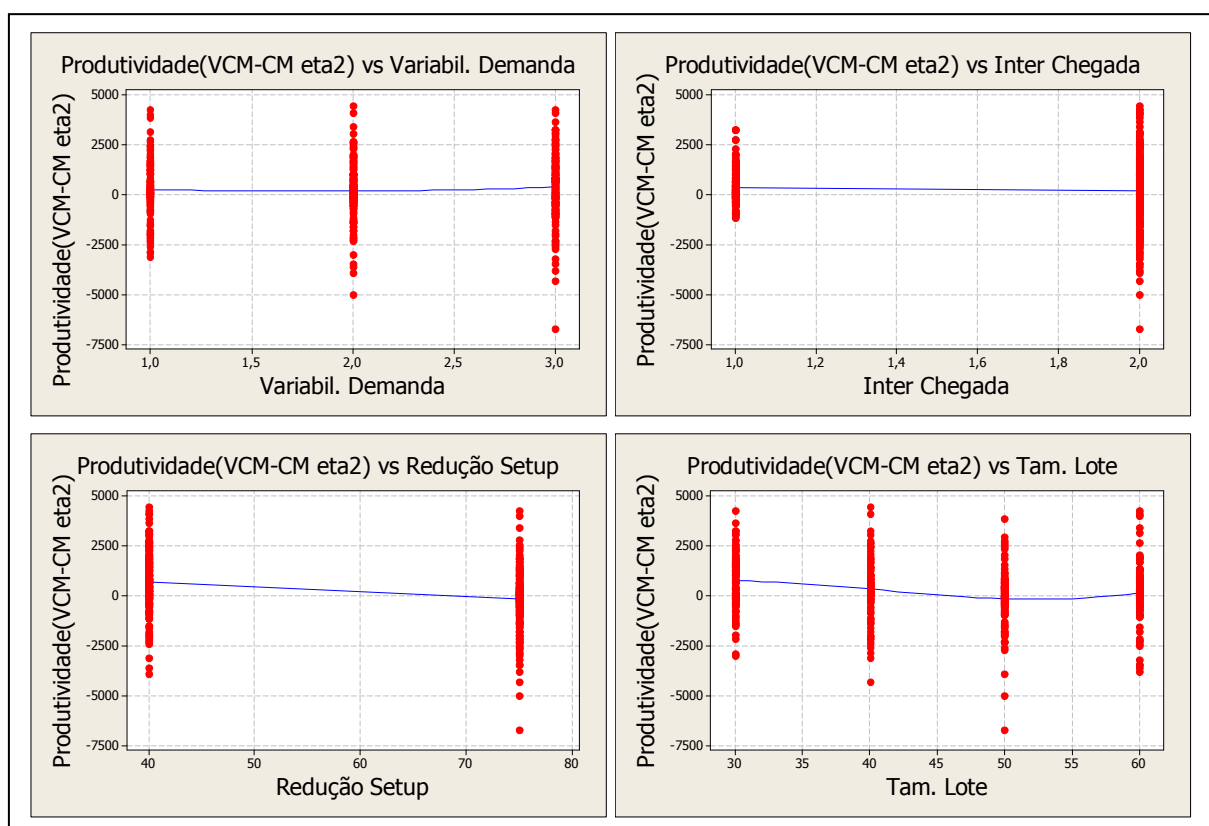


Figura A4.21 – “Scatterplots” com regressão da produtividade VCM-CM para eta ( $\eta$ ) = 2



#### A4.4.5 Diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para $\eta = 3$

A tabela A4.20 apresenta os valores da análise de variância da diferença de produtividade entre o celular virtual (VCM) e o celular tradicional (CM) para  $\eta = 3$ . Nenhum dos fatores afeta significativamente a variável resposta.

Tabela A4.20 Análise de variância da produtividade VCM-CM para  $\eta = 3$

Source	DF	SS	MS	F	P
Variabil. Demanda	2	50202	25101	0,02	0,983
Inter Chegada	1	29141	29141	0,02	0,888
Redução Setup	1	2093521	2093521	1,42	0,234
Tam. Lote	3	9891076	3297025	2,24	0,083
Error	472	694644920	1471705		
Total	479	706708859			

A figura A4.22 apresenta os gráficos “Scatterplots” com regressão para as diferenças de produtividade entre a célula virtual e o celular tradicional versus cada um dos quatro fatores do experimento, para  $\eta = 3$ . De fato, os gráficos indicam que não há variação significativa na variável resposta diferença de produtividade para os quatro fatores avaliados, dentro do escopo do estudo. Pode-se dizer que ambos apresentam o mesmo desempenho em todos os fatores e níveis examinados para este nível de flexibilidade de operador, ou seja, para  $\eta = 3$

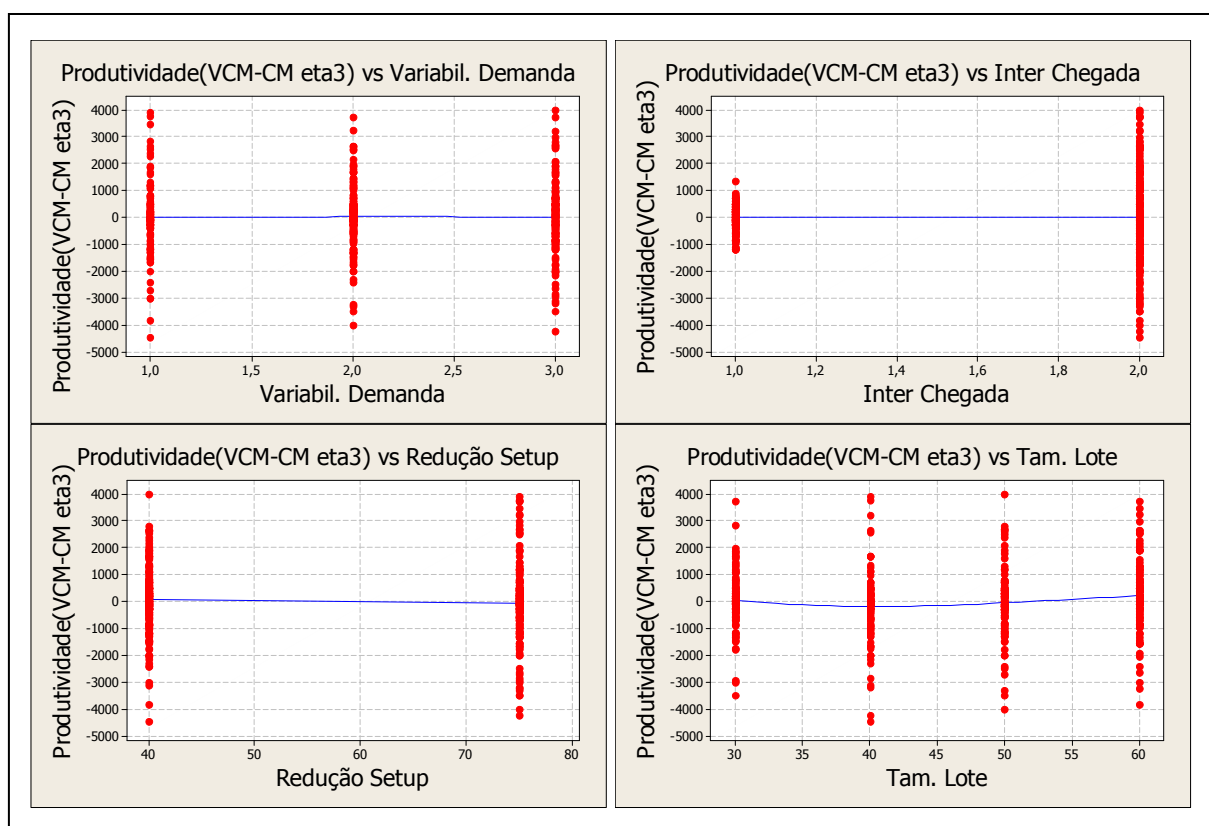


Figura A4.22 – “Scatterplots” com regressão da produtividade VCM-CM para  $\eta = 3$