

Rodrigo Rizzi Starr

**Contribuições para a detecção de vazamentos em
tubulações de gás natural:
Uma abordagem baseada em conhecimento**

Florianópolis

30 de julho de 2006

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

**CONTRIBUIÇÕES PARA A DETECÇÃO DE VAZAMENTOS EM
TUBULAÇÕES DE GÁS NATURAL: UMA ABORDAGEM BASEADA
EM CONHECIMENTO**

Dissertação submetida à

UNIVERSIDADE FEDERAL DE SANTA CATARINA

para obtenção do grau de

MESTRE EM ENGENHARIA MECÂNICA

RODRIGO RIZZI STARR

Florianópolis, julho de 2006.

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

**CONTRIBUIÇÕES PARA A DETECÇÃO DE VAZAMENTOS EM TUBULAÇÕES DE
GÁS NATURAL: UMA ABORDAGEM BASEADA EM CONHECIMENTO**

RODRIGO RIZZI STARR

Esta dissertação foi julgada adequada para obtenção do título de
MESTRE EM ENGENHARIA

ESPECIALIDADE ENGENHARIA MECÂNICA
sendo aprovada em sua forma final.

Prof. Jonny Carlos da Silva, Dr. Eng. – Orientador

Prof. Fernando Cabral, Ph.D. – Coordenador do Curso

BANCA EXAMINADORA

Prof. Acires Dias, Dr. Eng. – Presidente

Prof. Guilherme Bittencourt, Dr. Rer. Nat.

Prof. Nelson Back, Ph. D.

E gozo, num momento sensitivo e competente,
A libertação de todas as especulações
E a consciência de que a metafísica é uma
consequência de estar mal disposto.

Fernando Pessoa

À minha mãe, que teria gostado de ver isto pronto

Agradecimentos

Agradeço a todos os que me aguentaram por esses longos anos. Em especial, gostaria de agradecer à minha família, por ter me apoiado quando decidi fazer um mestrado. Também quero agradecer aos membros da equipe do projeto SEGRED, pois tornaram suportáveis os percalços que o projeto enfrentou. Ainda relacionado ao projeto SEGRED, gostaria de agradecer a compreensão e o interesse de todos os controladores do gasoduto que participaram das entrevistas de aquisição de conhecimento.

Também gostaria de agradecer aos muitos amigos que me ajudaram lendo este trabalho e dando opiniões. Estes foram: meu pai, Greice Bauer, Luis Fernando Calil, Yuji Sakurada, Cyro Albuquerque e Gustavo Alves. Muitos dos acertos deste trabalho são culpa deles. Os erros, obviamente, são culpa única e exclusivamente minha. Quero também agradecer ao Gustavo Alves pelas frutíferas discussões sobre programação, por ter me auxiliado a decifrar algumas das mensagens de erro mais crípticas do g++ e por ter me ensinado a fazer compilação cruzada.

Aos meus companheiros de república Luis Antonio e Yuji agradeço por terem relevado meu mau humor e minha recalcitrante recusa em lavar a louça durante os meses finais desse trabalho.

Agradeço também a Greice por ter me aturado durante esses dois anos, inclusive nos meus momentos de mau humor.

Finalmente, agradeço ao meu orientador pelo apoio e a CNPq por ter patrocinado parte deste trabalho.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
Lista de siglas	xii
Lista de símbolos e operadores	xiv
Resumo	xvi
Abstract	xvii
1 Introdução	1
1.1 Objetivo e justificativa do trabalho	2
1.2 Gasoduto Bolívia-Brasil	2
1.3 Projeto SEGRED	8
1.4 Organização do texto	10
2 Detecção de vazamentos	12
2.1 Detecção de vazamentos por métodos biológicos	14
2.2 Detecção de vazamentos por <i>hardware</i>	14
2.3 Detecção de vazamentos por <i>software</i>	16
2.3.1 Variações abruptas de fluxo ou pressão	16
2.3.2 Balanço de massa ou volume	17
2.3.3 Sistemas baseados em modelo dinâmico	18

2.3.4	Análise pontual de pressão	19
2.4	Descrição de algumas implementações	20
2.4.1	Métodos baseados em identificação de sistemas e sistemas dinâmicos	20
2.4.2	Métodos baseados em análises estatísticas	24
2.4.3	Métodos baseados em técnicas de inteligência artificial ou heurísticas	27
2.5	Deficiências de sistemas de detecção de vazamentos por <i>software</i>	31
2.6	Sistema instalado na empresa-alvo	32
2.7	Conclusões do capítulo	36
3	Sistemas baseados em conhecimento	37
3.1	Engenharia do conhecimento	37
3.2	Representação de conhecimento e inferência	40
3.2.1	Ontologias, quadros e redes semânticas	41
3.2.2	Sistemas de produção	47
3.2.3	Lógica difusa	51
3.3	Ferramentas	57
3.4	Conclusões do capítulo	59
4	Descrição do protótipo	60
4.1	Requisitos do sistema	61
4.2	Aquisição de conhecimento	63
4.2.1	Análises que devem ser realizadas pelo sistema	67
4.3	Implementação da base de conhecimento	70
4.3.1	Arquitetura do sistema	70
4.3.2	Análise dos sinais do SCADA	82
4.3.3	Sistema de inferência difusa	85
4.4	Testes e avaliação de performance	89

4.4.1	Testes realizados sobre o subsistema de análise numérica	89
4.4.2	Testes realizados sobre o sistema como um todo	90
4.5	Arquitetura proposta para o módulo de explicação	92
4.6	Comentários sobre o desenvolvimento	95
5	Conclusão e recomendações para trabalhos futuros	98
5.1	Recomendações para trabalhos futuros	100
	Referências	102
	Apêndice A Exemplos de uso do sistema	107
	Apêndice B Vocabulário técnico sobre gasodutos	110

Lista de Figuras

1.1	Mapa do gasoduto Bolívia-Brasil	3
1.2	Estação de entrega	4
1.3	Estação de compressão	5
1.4	Válvula de bloqueio de trecho	6
1.5	Exemplo de tela do SCADA	7
1.6	Visão geral da arquitetura do SEGRED Rede	9
1.7	Diagrama das classes que representam o gasoduto.	9
2.1	Trecho de tubulação utilizado para simular um vazamento.	29
2.2	Esquema do funcionamento dos filtros PQ e QP.	33
3.1	Exemplo de uma ontologia simples com apenas um tipo de relação	43
3.2	Exemplo de uma ontologia mais complexa	44
3.3	Exemplo de rede semântica	45
3.4	Exemplo de um quadro e de um sistema de quadros	46
3.5	Arquitetura de um sistema especialista	49
4.1	Ontologia de situações operacionais possíveis para o sistema especialista.	68
4.2	Diagrama do processo de inferência da classificação heurística	70
4.3	Diagrama das classes utilizadas para representar o gasoduto	72
4.4	Classes para a representação dos estados dos filtros e conclusões do sistema	73
4.5	Diagrama de influência das análises no contexto de filtros	76
4.6	Diagrama de influência das análises no contexto de análises dos sensores	78
4.7	Diagrama de influência das análises no contexto de posição geográfica	79
4.8	Diagrama de influência das análises intermediárias	80

4.9	Diagrama de influência da parte final das análises	81
4.10	Exemplo de aproximação produzida pelo algoritmo	83
4.11	Classes da implementação das análises sobre os dados dos sensores.	84
4.12	Diagrama das classes utilizadas no sistema de inferência difusa.	88
4.13	Árvore de inferência	93
A.1	Tela inicial do sistema de detecção de vazamentos	107
A.2	Exemplo de análise do detector de vazamentos: situação normal	108
A.3	Exemplo de análise do detector de vazamentos: vazamento próximo a EE . . .	108
A.4	Exemplo de análise do detector de vazamentos: empacotamento	109

Lista de Tabelas

2.1	Características da linha simulada em (BILLMAN; ISERMANN, 1987).	22
2.2	Características da linha simulada em (BENKHEROUF; ALLIDINA, 1988).	23
2.3	Características do duto Niigata-Sendai (Japão).	24
2.4	Características da linha simulada em (TURNER; MUDFORD, 1988).	25
2.5	Características da linha de testes utilizada em (WANG et al., 1993).	27
2.6	Características da linha de referência utilizada em (BELSITO et al., 1998).	27
2.7	Desempenho do sistema de localização de vazamentos de (BELSITO et al., 1998).	28
4.1	Mapa de conhecimento para o sistema.	68
4.2	Valores dos conjuntos difusos utilizados na análise dos filtros.	75
4.3	Valores para os filtros do PMS em cada um dos casos testados	90
4.4	Resultados dos testes para as situações bem determinadas.	91
4.5	Resultados dos testes para as situações mal determinadas	91
4.6	Situação da implementação de cada um dos requisitos do sistema.	95

Lista de siglas

ANP	Agência Nacional do Petróleo, Gás Natural e Biocombustíveis
CENPES	Centro de pesquisas da Petrobrás
CLIPS	<i>C Language Expert System Shell</i> – shell para produção de sistemas especialistas
CLP	Controlador Lógico Programável
CSC	Central de supervisão e controle
ECOMP	Estação de compressão
EE	Estação de entrega
GLP	Gás Liquefeito de Petróleo
GNC	Gás Natural Comprimido
GNL	Gás Natural Liquefeito
GNV	Gás Natural Veicular
IA	Inteligência Artificial
LGN	Líquido de Gás Natural
LSS	<i>Leak Sensitivity Study</i> – estudo de sensibilidade para detecção de vazamentos
MCC	Manutenção centrada em confiabilidade
NASA	<i>National Aeronautics and Space Administration</i> – agência de administração aeronáutica e espacial americana
PIG	<i>Pipeline Inspection Gauge</i> – dispositivo de inspeção de dutos
PMS	<i>Pipeline Management System</i> – sistema de detecção de vazamentos e gerenciamento de gasodutos fornecido pela <i>Energy Solutions International</i>

PQ	Modelo onde as condições de contorno a montante são de pressão e as condições de contorno a jusante são de vazão
QP	Modelo onde as condições de contorno a montante são de vazão e as condições de contorno a jusante são de pressão
RNA	Rede neural artificial
SCADA	<i>Supervisory Control and Data Acquisition</i> – Sistema de aquisição de dados e controle supervísório
SDV	Sistema de Detecção de Vazamentos
SE	Sistema especialista
SEGRED	Sistema Especialista para Gerenciamento de Redes de Transporte de Gás Natural
SVD	<i>Singular Value Decomposition</i> – decomposição em valores singulares
TBG	Transportadora Brasileira Gasoduto Bolívia-Brasil S.A.
UF	<i>Unexpected Flow</i> – desbalanceamento de vazão
UFSC	Universidade Federal de Santa Catarina
UP	<i>Unexpected Pressure</i> – desbalanceamento de pressão
UPGN	Unidade de Processamento de Gás Natural
VES	Válvula de bloqueio de trecho (válvula de esfera)
VSAT	<i>Very Small Aperture Terminal</i> – sistema de comunicação via satélite

Lista de símbolos e operadores

q	Fluxo de massa em um trecho do duto
q_i	Fluxo de massa entrando em um trecho do duto
q_o	Fluxo de massa saindo de um trecho do duto
M_e	Massa contida em um trecho do duto
R	Valor do desbalanceamento de massa no duto
A	Parâmetro de sensibilidade de um detector de vazamento
$E\{g(x)\}$	Valor esperado de $g(x)$. É definida (KOOB, 2003), para o caso em que x é discreto, como:

$$E\{g(x)\} = \sum_{i=1}^N g(x_i)p(x_i)$$

onde $p(x_i)$ é a probabilidade de ocorrência do evento x_i . Para o caso contínuo a definição fica:

$$E\{g(x)\} = \int_{-\infty}^{\infty} g(x)p(x)dx$$

desde que $|E\{g(x)\}| < \infty$

\hat{x}	O valor estimado de x
σ	Desvio padrão
X	Universo de discurso
\underline{A}	Conjunto difuso
$\mu_{\underline{A}}$	Função de pertinência de um conjunto difuso
$\mu_{\underline{A}}(x)$	Valor da pertinência do elemento $x \in X$ ao conjunto $\mu_{\underline{A}}$
\underline{a}	Variável difusa. Representa a pertinência de um determinado elemento a um determinado conjunto difuso
\underline{h}	Função difusa

\underline{g}	Função de agregação
\underline{a}	Função de ativação
\underline{w}	Função peso
$\min(\cdot, \dots)$	Mínimo entre vários elementos, ou o menor elemento de um vetor
$\max(\cdot, \dots)$	Máximo entre vários elementos, ou o maior elemento de um vetor
$\sup_x(f(x))$	O maior valor da função $f(x)$, para x variando dentro de seu domínio
$\inf_x(f(x))$	O menor valor da função $f(x)$, para x variando dentro de seu domínio
$\vee(\cdot)$	Operador de disjunção (“ou” lógico)
$\wedge(\cdot)$	Operador de conjunção (“e” lógico)
$\neg(\cdot)$	Operador de negação
$G(\underline{a})$	Valor de certeza de uma variável difusa. É definido como:
	$G(\underline{a}) = 2 \left \underline{a} - \frac{1}{2} \right $
$e(\underline{A})$	Entropia de um conjunto difuso \underline{A} . É definida como:
	$e(\underline{A}) = \sum_{x \in X} \left(1 - 2 \left \mu_{\underline{A}}(x) - \frac{1}{2} \right \right)$
\mathbb{N}	Conjunto dos números naturais
\mathbb{Z}	Conjunto dos números inteiros
\mathbb{R}	Conjunto dos números reais
L_I	Cadeia de caracteres

Resumo

Este trabalho trata da criação do protótipo de um sistema especialista para refinar métodos de detecção de vazamentos já existentes. A presença de sistemas de detecção de vazamentos em dutos de transporte de óleo ou gás é importante pois, devido às grandes vazões e ao tipo de material transportado nesses dutos, é preciso detectar vazamentos rapidamente, de modo a evitar perdas humanas, financeiras e danos ambientais. Um dos métodos mais comuns para a detecção de vazamentos é a realização de análises sobre os valores de sensores já presentes no duto para outras funções. Este método de detecção de vazamentos costuma ser chamado de detecção de vazamentos por *software*. Apesar de ser amplamente utilizado na indústria, este método tem algumas deficiências, entre elas uma sensibilidade reduzida e em alguns casos uma taxa de alarmes falsos elevada. No entanto, com o contato diário com estes sistemas, os controladores do duto acabam desenvolvendo heurísticas para lidar com as deficiências do sistema. Além disso, os controladores costumam utilizar o resultado das análises do detector de vazamentos na avaliação de outras condições presentes no duto. Sendo assim, esse trabalho apresenta a implementação do protótipo de um sistema especialista capaz de realizar algumas das análises realizadas pelos controladores do duto. Esta implementação foi feita em parceria com uma empresa controladora de um gasoduto de grandes dimensões. O sistema foi desenvolvido utilizando a fusão de um sistema especialista tradicional com um sistema baseado em lógica difusa. O protótipo foi capaz de realizar satisfatoriamente algumas das análises necessárias, porém o processo de desenvolvimento se mostrou extremamente dependente da empresa onde o sistema será instalado. Foi identificado que o suporte gerencial desta empresa é fundamental para o sucesso de um sistema deste tipo, sobretudo na etapa de aquisição de conhecimento.

Palavras-chave: Sistemas especialistas. Lógica difusa. Detecção de vazamentos. Gás natural.

Abstract

This work deals with the implementation of an expert system prototype to refine existing leak detection methods. The presence of leak detection systems in oil or gas transportation pipelines is important because, due to large flow rates and kinds of fluid usually flowing through these pipelines, it is necessary to promptly detect leaks, so as to avoid human and financial losses and environmental damages. One of the most common method of leak detection is the analysis of information provided by sensors installed on the pipeline for other functions. This method of leak detection is usually called software-based leak detection. Despite being widely used in industry, this method has some drawbacks, among them a small sensitivity and, sometimes, a high rate of false alarms. On the other hand, in the daily contact with these systems, pipeline controllers end up developing heuristics to deal with the system deficiencies. Besides, pipeline controllers also use the results of the leak detector analysis while evaluating other operational conditions. Thus, this work presents the implementation of an expert system prototype able to carry out some of the analysis performed by pipeline controllers. This prototype was implemented in a partnership with a gas pipeline controller company. This system has been developed using the fusion of a traditional expert system with a fuzzy logic based system. The prototype was able to perform satisfactorily some of the required analysis, but the development process has shown itself very dependent on the company where the system will be installed. It has been identified that management support from this company is of fundamental importance to the success of the development process, specially during knowledge acquisition.

Keywords: Expert systems. Fuzzy logic. Leak detection. Natural gas.

1 Introdução

Enquanto gasodutos e oleodutos são um método eficiente e econômico de transportar fluidos por longas distâncias, os riscos associados a vazamentos potenciais são altos. Vazamentos em dutos carregando fluidos como óleo, gasolina ou gás natural podem causar graves acidentes, com risco de perdas humanas, danos ao meio ambiente e perdas materiais (BELSITO et al., 1998). Além disso, nos dutos de transporte de líquidos, existe também o problema do roubo de mercadoria (IOURTCHEENKO; LOSENKOV, 2002).

Como em dutos de transporte normalmente as vazões são muito grandes, mesmo o vazamento de uma porcentagem pequena do fluxo de massa do duto pode ter conseqüências catastróficas se não for mitigado a tempo. Além disso, este tipo de duto costuma passar por regiões pouco habitadas, onde um acidente, ainda que tenha uma chance menor de causar perdas humanas, pode ficar oculto por mais tempo, provocando maiores danos ambientais e financeiros.

Por estas razões, dutos deste tipo sempre contam com algum sistema de detecção de vazamentos (SDV). Pelas mesmas razões, as legislações sobre o setor nos EUA e na Alemanha exigem que este tipo de sistema sempre seja instalado. A norma alemã requer ainda que dois sistemas independentes estejam ativos durante a operação e pelo menos um esteja ativo caso o duto esteja inoperante, mas não esteja vazio (MATKO et al., 2003; EUA, 2005).

Estes sistemas de detecção de vazamentos operam sobre dados obtidos de sensores no duto e, utilizando diferentes técnicas, determinam se está ocorrendo um vazamento e, quando possível, onde ele está ocorrendo e qual a quantidade de produto vazado. Assim, devido aos dados recolhidos do duto ou às análises realizadas, é comum em implementações comerciais que esses sistemas de detecção apresentem para os controladores do duto mais informações além das relacionadas à detecção de vazamentos propriamente dita. Por isso estes sistemas normalmente integram outras funções, como rastreamento de PIGs (*Pipeline Inspection Gauge* – dispositivo para inspeção de dutos) e controle de estoque (quantidade de produto dentro do duto).

Diante do exposto acima, este trabalho visa estudar as vantagens que uma abordagem baseada em inteligência artificial (IA) pode fornecer ao ser aplicada sobre os métodos tradicionais

de detecção de vazamentos. Esse estudo foi feito no contexto do projeto SEGRED, que é discutido mais adiante (seção 1.3). Para este projeto foi implementado um protótipo junto com a empresa administradora do gasoduto Bolívia-Brasil, a TBG. Esta empresa será chamada, de agora em diante, de empresa-alvo.

1.1 Objetivo e justificativa do trabalho

O objetivo deste trabalho é refinar o resultado de algoritmos de detecção de vazamentos já existentes com o tipo de análise que atualmente só pode ser feita pelos controladores do duto. Assim, o que se propõe é melhorar um sistema de detecção de vazamentos já existente utilizando o conhecimento dos controladores.

Para avaliar as vantagens e desvantagens desta abordagem foi feito um protótipo de um sistema deste tipo. Para a implementação, foi utilizada uma integração entre sistemas baseados em regras e sistemas difusos. Os detalhes do processo de desenvolvimento estão descritos no capítulo 4 e permitem um estudo de caso do desenvolvimento de um sistema especialista.

A justificativa para uma abordagem desse tipo é que, como para a detecção de vazamentos é necessário realizar análises detalhadas do estado do gasoduto, várias outras situações de interesse para os controladores também podem ser detectadas. Além disso, algumas situações podem influenciar no resultado do detector, fazendo com que ele emita alarmes falsos, ou não seja capaz de executar sua função corretamente. Criar um sistema que seja capaz de identificar essas situações e informar aos controladores, ou, por outro lado, utilizar as análises do detector para identificar diferentes situações pode representar uma vantagem em termos operacionais.

1.2 Gasoduto Bolívia-Brasil

O sistema proposto neste trabalho foi implementado para o trecho brasileiro do gasoduto Bolívia-Brasil. Nesta seção serão descritos detalhes do gasoduto necessários à compreensão do sistema.

O trecho brasileiro do gasoduto Bolívia-Brasil estende-se desde a fronteira com a Bolívia próxima a Corumbá até a cidade de Canoas, próxima a Porto Alegre, no Rio Grande do Sul. O gasoduto possui um comprimento total de 3.150 km, sendo 2.593 km no lado brasileiro. A Figura 1.1 mostra um mapa com o gasoduto e todas as suas estações.



Figura 1.1: Mapa do gasoduto Bolívia-Brasil (cortesia da empresa-alvo).

Do ponto de vista deste trabalho, são importantes os seguintes componentes do gasoduto, além da tubulação propriamente dita:

- Estações de entrega (EES);
- Estações de compressão (ECOMPs);
- Válvulas de bloqueio de trecho (VESs);
- Central de supervisão e controle (CSC).

As estações de entrega são os pontos onde a custódia do gás é transferida da empresa-alvo para as distribuidoras regionais. Diz-se que ocorre uma transferência de custódia porque a empresa-alvo não é dona do gás transportado. A principal função dessas estações é diminuir a temperatura e pressão do gás até a condição de entrega, que é definida por contrato, e medir a quantidade de gás que foi entregue. Assim, nas EEs existem tanto válvulas para diminuir a pressão do gás da pressão em que ele é transportado (normalmente entre $60 \cdot 10^5$ a $100 \cdot 10^5$ Pa) até a pressão em que ele deve ser entregue para a distribuidora (em torno de $35 \cdot 10^5$ Pa), quanto aquecedores para compensar a queda de temperatura do gás devida à expansão e manter o gás ao redor da temperatura de entrega, que normalmente é cerca de 20 °C. Além disso, estas estações contam com sistemas para medir a quantidade de gás entregue, e sistemas de monitoração

e controle que, entre outras atividades, comunicam-se regularmente com o SCADA¹ (*Supervisory Control and Data Acquisition* – sistema de aquisição de dados e controle supervisão) da empresa-alvo, para transmitir informações pertinentes. Normalmente a frequência com que ocorre a comunicação é baixa, em torno de 3 vezes por dia. Estas estações contam com sistemas para que, caso ocorra alguma variação muito grande em algum parâmetro monitorado, o novo valor seja enviado imediatamente ao SCADA, independentemente de quando foi enviado o último pacote de dados. A Figura 1.2 mostra uma estação de entrega.

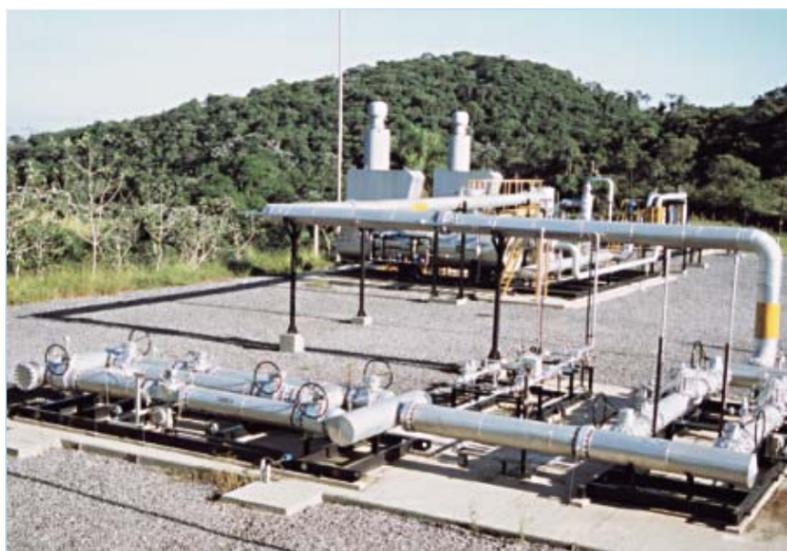


Figura 1.2: Estação de entrega (cortesia da empresa-alvo).

As estações de compressão, como a mostrada na Figura 1.3, são responsáveis por fornecer energia ao gás para compensar a perda de carga do duto. É através delas que se ajusta a quantidade de gás que é transportada pelo gasoduto, e sua correta operação e funcionamento é fundamental para que a demanda de gás dos clientes possa ser atendida. Com exceção de duas estações no estado de São Paulo, todas as outras estações do gasoduto são controladas remotamente a partir da central de supervisão e controle (CSC), na sede da empresa-alvo no Rio de Janeiro. Uma ECOMP é uma instalação complexa, que inclui de 2 a 4 compressores, centrífugos ou alternativos; turbinas ou motores alternativos para acionamento dos compressores; resfriadores para o gás natural (que se aquece devido ao trabalho de compressão); filtros; sistema de gás de utilidades (utilizado, por exemplo, para alimentar os motores dos compressores) e sistema de supervisão e controle. Devido à importância e complexidade dessas estações, a frequência de comunicação com a CSC é muito maior que no caso das EEs, sendo em torno de uma transferência a cada segundo ou a cada três segundos.

¹Neste trabalho, quando se opta por utilizar um termo ou sigla no seu original em inglês, é sempre em concordância com o modo como o termo é utilizado dentro da empresa-alvo.



Figura 1.3: Vista geral de uma estação de compressão (cortesia da empresa-alvo).

Como existem variações no consumo de cada EE, dependendo da hora do dia, do dia da semana, e mesmo da época do ano, as vazões nas ECOMPs devem ser sempre ajustadas de acordo com o consumo previsto. Por esta razão é comum aumentar ou diminuir a potência aplicada pelos compressores. Por isso é uma situação rara o gasoduto estar em regime permanente. A situação normal é estarem ocorrendo transientes de pressão e vazão devido a esses ajustes.

Além dessas duas estações, um outro componente do gasoduto de interesse para o trabalho é a válvula de bloqueio de trecho (VES). Estas válvulas se encontram distribuídas mais ou menos uniformemente ao longo do duto, com uma distância média de 30 km entre elas. Sua função é isolar um trecho do gasoduto caso ocorra o rompimento da tubulação. Estas válvulas devem interromper o duto caso uma das seguintes condições seja detectada: ou a pressão no gasoduto está abaixo de um determinado limite (que varia dependendo da localização da válvula), ou a pressão está diminuindo com uma velocidade maior que um determinado limite (que também varia de acordo com a localização da válvula). A Figura 1.4 mostra uma válvula de bloqueio.

O princípio de funcionamento é o seguinte: em caso de rompimento do duto, a pressão a montante e a jusante do ponto de rompimento cairá rapidamente, e as válvulas a montante e a jusante deste ponto se fecharão, efetivamente isolando o trecho e impedindo que vaze para a atmosfera uma quantidade maior de gás. Pode ocorrer também de a válvula se fechar acidentalmente, devido a transientes de pressão no gasoduto (SALAS; REID, 2002). Estes transientes podem ocorrer, por exemplo, devido à entrada em operação de um compressor em uma ECOMP.

Estas válvulas não possuem nenhum tipo de monitoramento remoto e são pneumáticas, acionadas pelo próprio gás do gasoduto. Desse modo, caso ocorra um fechamento de válvula, espúrio ou não, o único meio de se identificar que ocorreu o fechamento é através do efeito nos medidores de vazão e/ou pressão nas estações próximas à válvula. Mesmo assim, por esse método é difícil identificar com certeza a válvula que se fechou, pois é comum haver mais de uma válvula entre duas estações. Portanto, para se encontrar a válvula fechada, e para abrir



Figura 1.4: Válvula de bloqueio de trecho. Elas normalmente são enterradas (cortesia da empresa-alvo).

novamente a válvula caso tenha ocorrido fechamento espúrio, é preciso verificar várias delas. Este processo pode ser bastante lento, podendo demorar várias horas dependendo das condições climáticas, uma vez que algumas válvulas se encontram em locais de difícil acesso.

Finalmente, é preciso compreender o funcionamento da central de supervisão e controle (CSC) da empresa-alvo. É a partir desta central, instalada na cidade do Rio de Janeiro, que todo o gasoduto é monitorado e controlado. Todos os dados recolhidos de todos os sensores do gasoduto são enviados para o sistema SCADA, cujos bancos de dados se encontram na CSC. O SCADA, além de armazenar esses valores para construir um banco de dados histórico, permite a apresentação desses valores em telas sinópticas (um exemplo pode ser visto na Figura 1.5) e a criação de alarmes para quando algum valor ultrapassa uma faixa considerada segura. O sistema fornece também uma interface para atuar em sistemas que podem ser controlados remotamente, como as ECOMPs, por exemplo.

Quando os dados das estações em campo são armazenados pelo SCADA, junto com cada ponto o sistema armazena um campo com informações sobre a qualidade daquele dado, chamado de *quality bit*². Os CLPs (controladores lógicos programáveis) localizados em cada estação, ao fazerem a aquisição dos dados dos sensores para enviar ao SCADA, podem determinar que existe algum sensor defeituoso. Neste caso, ao invés de enviar um valor sem sentido, o CLP informa ao SCADA que o sensor está defeituoso. Isto fica indicado no SCADA no campo

²Apesar do nome, este campo é capaz de armazenar mais valores além de 0 e 1, diferente da interpretação usual da palavra *bit*.

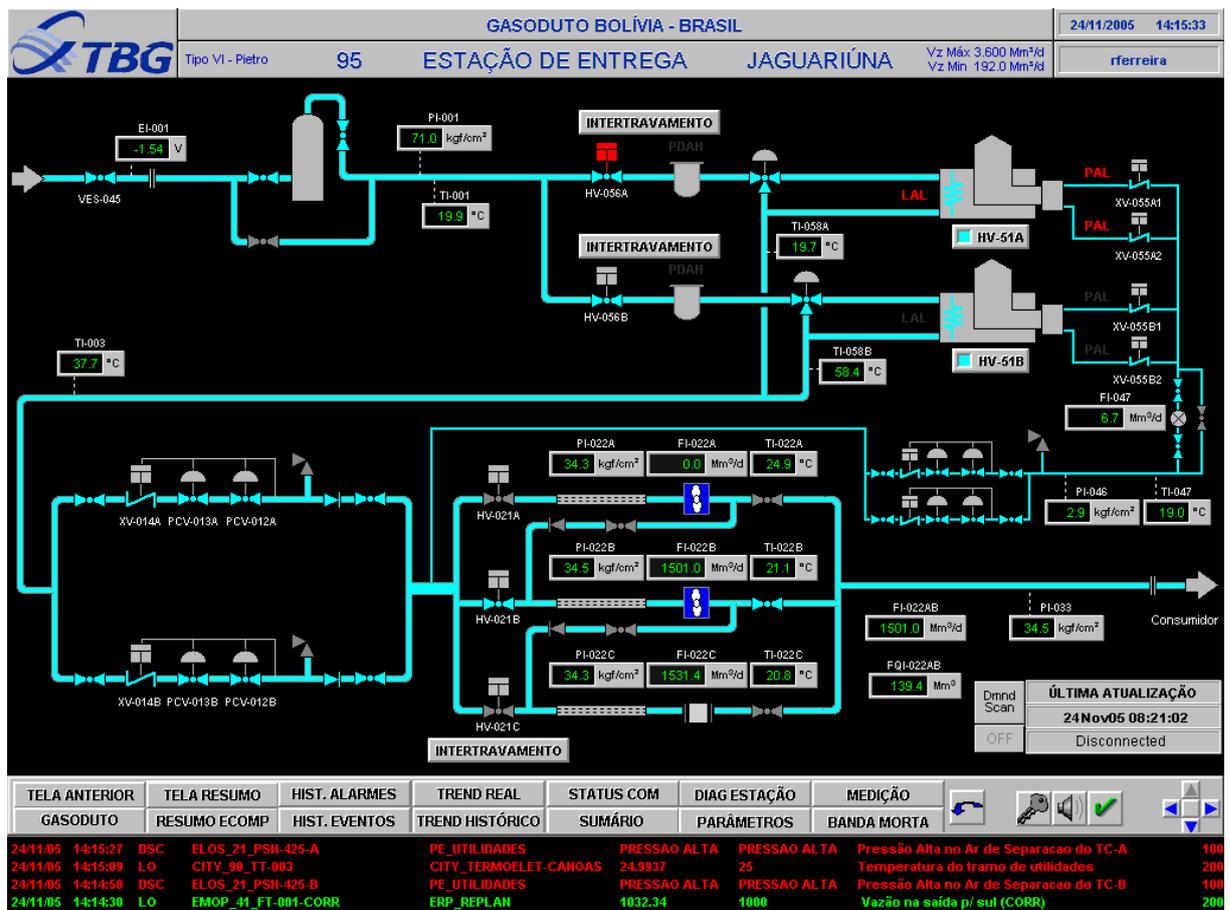


Figura 1.5: Exemplo de tela do SCADA. A tela mostra a estação de entrega de Jaguariúna. Nas 4 linhas inferiores é possível ver, em vermelho, os alarmes emitidos pelo SCADA (cortesia da empresa-alvo).

de qualidade daquela leitura. Além disso, existem casos em que a CSC pode ter interesse em desligar alguma leitura do SCADA. Isso pode ocorrer em estações que estão em manutenção, por exemplo, para evitar alarmes devido a valores fora da faixa. Para esses casos, o SCADA fornece a opção de substituir essa leitura por um valor fornecido pelo usuário. Diz-se que este sensor está “forçado”, e isto também é identificado no campo de qualidade. Essa é uma condição relativamente comum, e que deve ser levada em consideração por qualquer sistema de detecção de vazamentos, por exemplo.

Na central de controle estão, por turno, dois engenheiros, chamados controladores, que monitoram os valores do SCADA e controlam todo o gasoduto. É responsabilidade dessas pessoas identificar problemas no gasoduto, e acionar o pessoal necessário para resolvê-los. Também é responsabilidade delas ajustar as vazões no gasoduto para que todos os consumidores sejam atendidos. Essas tarefas são muito envolventes e complexas, o que costuma gerar estafa nos controladores.

1.3 Projeto SEGRED

O SEGRED é um sistema que vem sendo desenvolvido desde 2000, em uma parceria entre a UFSC, a empresa-alvo e a Petrobrás/CENPES. O objetivo inicial do sistema é ser uma ferramenta de suporte à identificação de falhas e situações potencialmente perigosas em estações de entrega. Com o prosseguimento do projeto, a empresa-alvo mostrou interesse em expandir o sistema para supervisionar todas as operações do gasoduto. Assim, dois sistemas foram desenvolvidos: o SEGRED Estações (CASTELANI et al., 2002; SILVA JR.; SILVA, 2002), cuja base de conhecimento foca as estações de entrega, e o SEGRED Rede (SILVA; PORCIÚNCULA, 2003; SILVA et al., 2004), que considera a operação do gasoduto como um todo, sem entrar em detalhes de cada estação ou equipamento. É como parte do escopo do SEGRED Rede que o presente trabalho foi desenvolvido. Por ser o foco do trabalho, sempre que se utilizar SEGRED, se estará tratando do SEGRED Rede.

Uma das características particulares do SEGRED é a integração entre o sistema especialista e um *software* de simulação dinâmica. Esta parece ser uma tendência da aplicação de sistemas especialistas na indústria de óleo e gás (JOHNSON, 1999; URAIKUL et al., 2000; WALLOOPPILLAI et al., 2000). Do ponto de vista de conhecimento, o sistema especialista contém conhecimento operacional sobre o gasoduto, enquanto o simulador contém conhecimento sobre a dinâmica do gasoduto.

Esta integração permite a análise de cenários operacionais, validação de medições do SCADA, e extrapolação automática do cenário operacional atual para produzir prognósticos. Isto emula um processo que é realizado em algumas situações pelos controladores do gasoduto, de estudar o seu comportamento em um simulador antes de realizar manobras mais complexas. No entanto, como esse processo de preparar a simulação, realizá-la e interpretar os resultados é moroso, isso é feito em relativamente poucas situações. Como o sistema especialista permite automatizar uma parte deste processo, pelo menos para os casos mais rotineiros, obtém-se uma melhora nas decisões tomadas pelos operadores (SAMPATH; YEE, 2000).

O programa tem a arquitetura apresentada na Figura 1.6. A base de conhecimento do sistema especialista é dividida em grupos de regras, cada um com funções bem definidas, e objetos que representam os componentes do gasoduto ou realizam funções auxiliares. As interações entre cada um desses agrupamentos é coordenada por um subsistema de controle, o qual, por sua vez, também é implementado como um conjunto de regras.

Existe um grupo de objetos para representar o conhecimento sobre a topologia e estrutura do gasoduto, fornecendo informações sobre o estado de cada componente, e o modo como estão

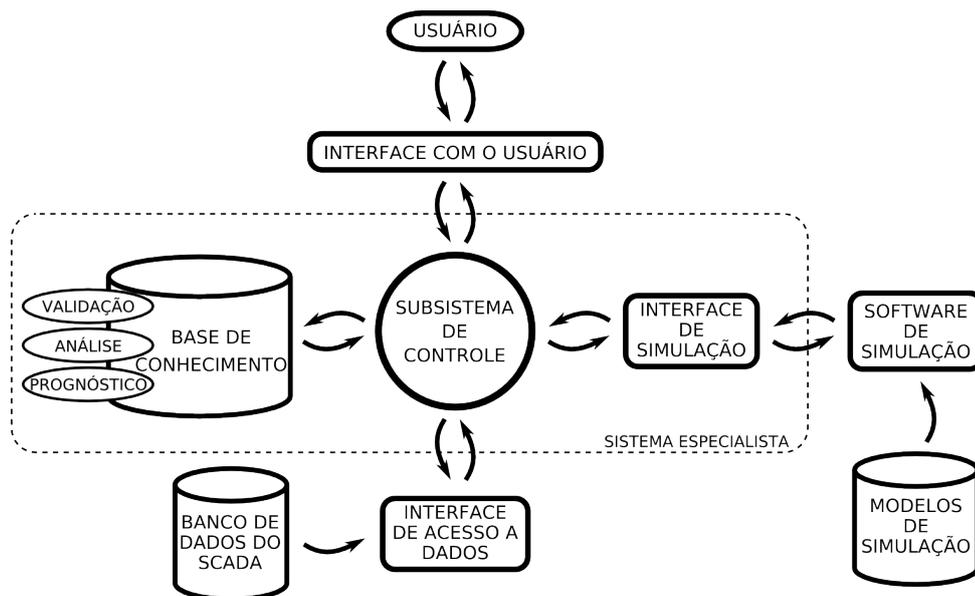


Figura 1.6: Visão geral da arquitetura do SEGRED Rede. As setas indicam o fluxo de informações.

organizados. Um diagrama dessas classes pode ser visto na Figura 1.7. Neste diagrama, VES significa válvula de bloqueio de trecho, EE estação de entrega, ECOMP estação de compressão, SRCP sistema de redução e controle de pressão, SM sistema de medição de vazão e SF sistema de filtragem e aquecimento (sendo estes três últimos subsistemas de uma estação de entrega). TRECHO é a classe que representa a organização geográfica dos componentes do gasoduto.

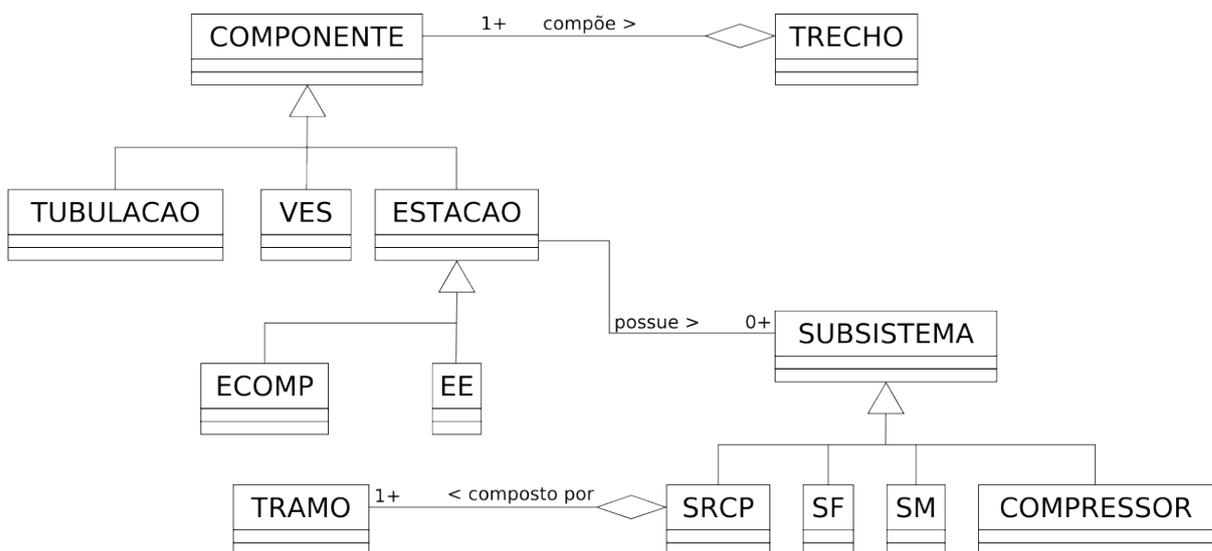


Figura 1.7: Diagrama das classes da base de conhecimento do SEGRED que representam o gasoduto.

O sistema realiza três funções principais: validação dos dados vindos do SCADA ou do operador; análise do estado atual do gasoduto, com foco na detecção de falhas; e prognóstico.

Validação Na fase de validação a consistência de todos os pontos lidos do SCADA ou fornecidos pelo operador é verificada. Caso alguma inconsistência seja detectada, um alarme é disparado e as causas possíveis da inconsistência são avaliadas. Se for desejado, os pontos problemáticos podem ser substituídos automaticamente, de forma que o sistema especialista possa levar a cabo as análises restantes.

Análise Nesta fase o estado de cada componente é comparado com os seus limites operacionais. Esta comparação busca determinar, por exemplo, se a pressão a montante de uma EE está muito baixa, ou se a temperatura do gás entregue está baixa. Estas informações são então associadas a uma série de causas possíveis, e a causa que melhor descreve a situação é informada ao controlador. Estes modos de falha foram obtidos a partir de relatórios de MCC (Manutenção Centrada em Confiabilidade) para cada estação.

Prognóstico Durante esta fase o simulador é usado para prever o estado do gasoduto algumas horas a frente (normalmente 6 ou 8 horas). O resultado desta simulação é alimentado no sistema especialista, que então executa análises semelhantes às executadas durante a fase de análise, sobre dados simulados. A partir destas análises são geradas para o operador mensagens relativas a cada operação. As mensagens indicam a situação, o componente, a hora em que foi identificada a situação e uma indicação da criticidade da situação.

A comunicação com o simulador é implementada através de uma interface de simulação, que traduz os dados entre o formato utilizado pelo simulador e pelo sistema especialista, e fornece um modo abstrato para acessar o simulador, diminuindo o acoplamento entre os programas.

Finalmente, a comunicação entre o sistema especialista e o SCADA é feita através de uma interface específica, que também busca diminuir o acoplamento entre a representação utilizada pelo sistema especialista e a representação utilizada na empresa-alvo.

1.4 Organização do texto

No capítulo 2 — Detecção de vazamentos — é apresentada uma revisão bibliográfica sobre técnicas de detecção de vazamentos. Uma descrição do mecanismo de detecção de vazamentos atualmente instalado na empresa-alvo e a justificativa da abordagem utilizada pelo presente trabalho também estão presentes neste capítulo.

Em seguida, no capítulo 3 — Sistemas baseados em conhecimento — são mostradas todas as técnicas utilizadas neste trabalho. Também são mostrados alguns aspectos relacionados ao

gerenciamento do projeto e desenvolvimento de sistemas especialistas.

No capítulo 4 — Descrição do protótipo — está descrita, em detalhes, a solução que foi implementada. Também encontra-se neste capítulo uma descrição do processo de desenvolvimento do sistema.

Finalmente, no capítulo 5 — Conclusão e recomendações para trabalhos futuros — alguns comentários são feitos sobre o desenvolvimento do sistema, vantagens e desvantagens da abordagem utilizada são discutidas, e algumas recomendações para pesquisa futura são propostas.

2 Detecção de vazamentos

Existem vários modos possíveis de se construir sistemas de detecção de vazamentos. Costuma-se classificar estes sistemas de acordo com esses modos. Existem duas classificações possíveis: a primeira os classifica em três categorias (ZHANG, 1997), levando em consideração o tipo de sensor utilizado para a detecção:

Métodos biológicos: Utilizam indicadores biológicos para a detecção.

Métodos por *hardware*: Requerem que novos sensores sejam instalados no duto.

Métodos por *software*: Utilizam apenas sensores instalados no duto para outras finalidades.

Apesar da nomenclatura, é importante notar que os métodos biológicos e por *hardware* também podem fazer uso de *softwares* para auxiliar no processamento de informações. O que diferencia estes dois métodos do método por *software* é o uso de outros sensores no duto além dos sensores já instalados para outras razões. De outro modo, os métodos de detecção de vazamentos por *software* têm esse nome porque dependem *exclusivamente* de um *software* para realizar a detecção.

A segunda classificação, recomendada pela API 1130 do *American Petroleum Institute*, divide os sistemas de detecção (SDVs) em duas categorias:

SDV *externally based*: Estes sistemas utilizam sensores externos ao duto, como por exemplo, sensores químicos para gases. Na classificação mostrada anteriormente todos os métodos de detecção biológicos e uma grande parte dos métodos por *hardware* se enquadram nesta categoria.

SDV *internally based*: Utilizam medidas internas do duto, como pressão e fluxo de massa, para identificar vazamentos. Todos os SDVs por *software* e alguns SDVs por *hardware* se enquadram nessa classificação.

Neste trabalho é usada a primeira classificação por se adequar melhor ao modo como o assunto é tratado. Também se faz necessário definir algumas métricas que indiquem o desempenho dos mesmos. Para tanto, apresenta-se o seguinte conjunto de métricas, que leva em consideração tanto critérios técnicos como financeiros (ENERGY, 2004; ZHANG, 1997):

Sensibilidade: Indica o menor vazamento detectável em condições ideais, ou seja, com o sistema SCADA funcionando corretamente e sem nenhum transiente ocorrendo no gasoduto. É expressa como uma vazão mássica.

Disponibilidade: Representa a porcentagem do tempo em que o sistema será capaz de funcionar com sensibilidade máxima. Devido a situações inesperadas, como transientes ou erros de instrumentação, o SDV pode precisar ter sua sensibilidade diminuída para evitar alarmes falsos. Além disso, existem SDVs que não funcionam de modo contínuo (*e.g.* os baseados em indicadores biológicos).

Credibilidade: Uma medida da probabilidade de um alarme produzido pelo sistema não ser um alarme falso, supondo que o SCADA esteja funcionando corretamente.

Precisão: Máxima distância entre a localização estimada e a localização real do vazamento. Se aplica apenas para os sistemas capazes de localizar o vazamento.

Custo de aquisição e comissionamento: Este valor inclui a compra de todos os equipamentos e *softwares* necessários para o funcionamento do sistema, além dos custos de instalação e configuração dos equipamentos, capacitação de pessoal, etc.

Custo de operação: Este valor inclui custos com a manutenção do sistema, gastos com pessoal e recursos para que o sistema permaneça em funcionamento.

Nas 3 seções seguintes serão descritas as principais características de cada um dos métodos de detecção de vazamentos: biológicos, por *hardware* e por *software*, com um maior detalhamento deste último. Em cada seção será dada uma pequena avaliação qualitativa do método com relação a algumas das métricas definidas acima, de acordo com a literatura pesquisada.

Em seguida a estas 3 seções serão apresentadas algumas implementações de detectores de vazamentos que foram encontradas na literatura. Depois se comentará sobre as deficiências dos métodos de detecção por *software* e se explicará o funcionamento do sistema de detecção instalado na empresa-alvo. O capítulo termina com alguns comentários conclusivos.

2.1 Detecção de vazamentos por métodos biológicos

Os métodos biológicos se baseiam na presença de um técnico humano especialmente treinado no local onde está ocorrendo o vazamento. Para realizar a detecção, o técnico pode levar em consideração ruídos, odores específicos do produto que o duto transporta¹, indicadores biológicos, como amarelamento da vegetação próxima ao vazamento (informação verbal)², ou indicadores físicos, como uma alteração do terreno (ZHANG, 1997). Também podem ser utilizados cachorros especialmente treinados para detectar sons ou odores característicos de um vazamento. Apesar de aparentemente simplório este é o método de detecção mais sensível existente até o momento, sendo que vazamentos muito pequenos só podem ser detectados através dele.

Claramente o uso desse método implica que a faixa onde se encontra o duto deva ser percorrida a pé em sua totalidade. Isso pode ser completamente inviável em alguns casos, como por exemplo, no oleoduto Trans-Alaska (que tem uma extensão de 1.242 km). Em outros casos as empresas agendam inspeções regulares da linha, como por exemplo o trecho brasileiro do gasoduto Bolívia-Brasil, que é inspecionado em sua totalidade a cada 6 meses (informação verbal)².

Em termos das métricas definidas anteriormente, o uso de métodos biológicos para a detecção de vazamentos apresenta ótima sensibilidade, precisão, credibilidade, baixo custo de instalação e um custo moderado de operação. Porém, devido a sua natureza intermitente, apresenta uma péssima disponibilidade. Entretanto, pode ser fundamental para a detecção de vazamentos pequenos demais para serem detectados por outros métodos.

2.2 Detecção de vazamentos por *hardware*

SDVs por *hardware* envolvem a instalação, ao longo do duto, de equipamento especializado para a detecção de vazamentos. Por esta razão, normalmente estes SDVs apresentam sensibilidade e credibilidade de razoável a ótima mas, em contrapartida, um custo de instalação alto e, em alguns casos, um custo de operação alto também. A disponibilidade pode variar de razoável a ótima, dependendo do método. Assim, esses sistemas normalmente são utilizados apenas em áreas de alto risco, como próximo a rios, reservas naturais ou regiões inacessíveis.

¹No caso do gás natural, este método não se aplica, pois ele é inodoro.

²Informações fornecidas pelo eng. Renato Aragonez, em palestra sobre inspeção de dutos ministrada em dezembro de 2004 no departamento de engenharia mecânica da Universidade Federal de Santa Catarina.

Existe uma enorme variedade de técnicas para a detecção (ZHANG, 1997; MATKO et al., 2003). Abaixo apenas algumas são apresentadas.

Emissão acústica: Uma série de transdutores acústicos é instalada ao longo do gasoduto e detectam o ruído produzido pelo gás vazando. Além do custo associado à instalação dos transdutores existe ainda o problema da transmissão remota dos dados (BRODETSKY; SAVIC, 1993).

Ondas de pressão: Transdutores de pressão muito sensíveis são instalados ao longo do gasoduto. Esses transdutores detectam a onda de pressão que percorre o duto quando da ocorrência do vazamento. O conhecimento da densidade do fluido permite estimar a posição do vazamento. Apresenta os mesmos problemas de custo e transmissão de dados que o método anterior.

Cabo sensor de fibra ótica: Um cabo de fibra ótica é instalado ao longo do duto. Ao ocorrer um vazamento, caso o gás entre em contato com o cabo, ocorrerá uma diminuição da temperatura que poderá ser detectada através de uma técnica chamada reflectometria ótica no domínio do tempo (MATKO et al., 2003).

Cabo sensor de vapores: Um tubo seletivamente permeável à substância que se deseja detectar é instalado ao longo do duto. Caso um vazamento ocorra, essa substância entrará em contato com o duto e será absorvida. Para se analisar o que foi absorvido pelo tubo uma bomba injeta ar em uma das extremidades a uma velocidade constante e de modo que o escoamento seja laminar. Na outra extremidade existe algum sensor capaz de detectar a substância de interesse. Cada aumento na concentração da substância de interesse é um “pico de vazamento”. Neste caso, a concentração da substância indica o tamanho do vazamento e é possível determinar a posição a partir da velocidade do escoamento no tubo.

Métodos visuais: Estes métodos se baseiam na diminuição ou aumento da temperatura no terreno ao redor do vazamento. Utilizando algum método de termografia, pode-se percorrer a faixa do duto e buscar um vazamento. Como estes equipamentos normalmente podem ser montados em automóveis ou helicópteros, torna-se possível inspecionar grandes distâncias em apenas um dia.

2.3 Detecção de vazamentos por *software*

SDVs por *software* são sistemas de detecção que utilizam dados de sensores já disponíveis no duto, normalmente nas estações de compressão ou de entrega, para buscar irregularidades que indiquem a presença de um possível vazamento. Estes sistemas se caracterizam como uma solução de compromisso entre custo e sensibilidade. A idéia é oferecer um sistema a um custo relativamente baixo, com alta disponibilidade e uma sensibilidade razoável.

Do mesmo modo que para a detecção de vazamentos por *hardware*, existem diversas técnicas possíveis de análise dos dados dos sensores. Estas técnicas podem ser divididas, de forma geral, em técnicas baseadas puramente em análise estatística e técnicas baseadas em princípios físicos, normalmente complementadas por alguma técnica estatística. Independente dos princípios utilizados no método, todos os métodos estudados apresentam uma estrutura do tipo extrator de características/classificador.

Nessa estrutura (FUKUNAGA, 1990), o primeiro processo na análise do sinal é um processo de extração de características. Neste, o sinal é analisado e transformado, normalmente, em uma n -upla de números. Esta n -upla é então passada para o classificador que, a partir destes valores e de alguns parâmetros internos, “decide” se está ou não ocorrendo um vazamento.

Uma característica comum a quase todos os SDVs por *software* é que sua sensibilidade é parametrizada. Desse modo é possível deixar o detector mais ou menos sensível, normalmente com um impacto direto sobre o número de alarmes falsos. Por conta disto, muitas implementações apresentam algum sistema auxiliar para diminuir a sensibilidade do detector (insensibilização) quando o duto passa por transientes fortes, que poderiam ser confundidos com vazamentos.

A seguir serão expostos os métodos básicos de detecção. Vale a pena notar que as categorias em que eles foram divididos as vezes se sobrepõem e que alguns métodos poderiam ser incluídos em mais de uma categoria. Como esta divisão é simplesmente didática, nestes casos decidiu-se incluí-los na categoria que pareceu mais pertinente ou que representava a característica principal do método.

2.3.1 Variações abruptas de fluxo ou pressão

Determina a ocorrência de um vazamento a partir de uma variação muito grande nos valores de vazão ou pressão. Ou seja, dispara sempre que o gradiente de pressão no sensor estiver abaixo de um determinado valor pré-estabelecido (isto é, quando a pressão estiver caindo rapidamente). Este mesmo princípio é utilizado nas válvulas de bloqueio de trecho, presentes a intervalos

regulares ao longo de um gasoduto para impedir que uma ruptura do duto provoque a liberação de todo o gás armazenado no mesmo. Este método, assim como as válvulas de bloqueio de trecho, ou gera muitos alarmes falsos ou tem uma sensibilidade muito baixa. Alguns SCADAs podem ser programados para gerar automaticamente alarmes desse tipo (SALAS; REID, 2002; WESTHOFF, 1999).

2.3.2 Balanço de massa ou volume

É um método baseado na aplicação da conservação da massa. O princípio de funcionamento é simples:

$$q_i(t) - q_o(t) = \frac{dM_e(t)}{dt} \quad (2.1)$$

onde $q_i(t)$ é o fluxo de massa na entrada do trecho, $q_o(t)$ é o fluxo de massa na saída do trecho, e $M_e(t)$ é a massa presente no trecho.

O grande problema deste método está em se determinar o valor do empacotamento (também chamado de inventário da linha ou *line pack*), o termo $M_e(t)$. Existe uma série de métodos para determinar esse valor, com mais ou menos precisão. Os mais utilizados são modelos de regime permanente e modelos dinâmicos. Em geral, quando se usa um modelo dinâmico para determinar o empacotamento, esse método é classificado em outra categoria, devido à complexidade do modelo dinâmico.

Modelos de regime permanente são utilizados principalmente para dutos de transporte de líquidos. A desvantagem desses modelos é que, por não levarem em consideração a dinâmica do sistema, apresentam divergências muito grandes quando o sistema está passando por um transiente. Nestes casos, a sensibilidade do detector deve ser diminuída para evitar alarmes falsos. Em (PIETSCH; PETHERICK, 1994) se apresenta um boa discussão da influência dos erros de estimativa do empacotamento na sensibilidade do detector de vazamentos.

A tomada de decisão (classificação) é feita a partir do desbalanceamento de linha, definido por:

$$R(t) = q_i(t) - q_o(t) - \frac{dM_e(t)}{dt} \quad (2.2)$$

Caso $R(t) > A$, onde A é a sensibilidade, se considera que existe um vazamento.

Uma linha de desenvolvimento comum é utilizar algum método de teste de hipótese sobre o valor de $R(t)$, de modo a obter uma menor taxa de alarmes falsos. Uma das desvantagens (que o próximo método a ser apresentado busca mitigar) deste método é que ele não leva em consideração uma série de fenômenos dinâmicos no duto, como a compressibilidade do fluido,

elasticidade das paredes do duto e trocas térmicas com o meio. Isso é uma enorme desvantagem, sobretudo em gasodutos, pois nestes o efeito de compressibilidade não é desprezível.

2.3.3 Sistemas baseados em modelo dinâmico

Muitos métodos de detecção são baseados em modelos dinâmicos do duto. Este modelo pode ser realizado de várias formas. A mais comum é utilizando um *software* simulador, normalmente específico para a tarefa. Um outro modo, utilizado em alguns métodos de detecção (ver seção 2.4.1), é a resolução analítica das equações dinâmicas que representam o duto e a implementação da solução em algum *software* específico. Independentemente de como é feita a modelagem, o modelo resultante pode ser utilizado de várias formas.

A mais óbvia delas é utilizá-lo para refinar o método apresentado na seção anterior. Ao se calcular o empacotamento com maior precisão é possível conseguir uma sensibilidade maior para aquele método e uma diminuição no número de alarmes falsos.

No entanto, existem outras aplicações possíveis, como analisar diretamente a diferença entre os resultados da simulação e os dados obtidos do SCADA. Sob esse ponto de vista o modelo funciona como um filtro, bastante elaborado, de transientes.

Outros métodos de detecção utilizando modelos dinâmicos podem ser encontrados na literatura acadêmica, mas não foram encontradas referências de aplicações práticas desses sistemas. Entre esses sistemas vale destacar o descrito por Loparo et al. (1991), que utiliza uma abordagem de múltiplos modelos, alguns representando o duto com vazamento, outros sem vazamento. A detecção consiste em identificar o modelo cuja resposta melhor corresponda à resposta medida no duto. Em um outro caso, se usa uma rede neural artificial junto com um modelo dinâmico para detectar e localizar vazamentos (BELSITO et al., 1998).

Apesar do aumento de sensibilidade que o modelo dinâmico oferece, esta técnica traz também algumas desvantagens. A primeira é que, entre todas as técnicas de detecção de vazamentos por *software*, ela é a mais custosa devido ao custo de criação e manutenção do modelo e do *software* de simulação. Além disso, alguns parâmetros do modelo, como a rugosidade do duto e os valores de calibração dos sensores, são difíceis de se obter e variam com o tempo. De acordo com alguns autores, só se obtém modelos realmente úteis para a detecção de vazamentos usando modelos adaptativos (NICHOLAS et al., 1992).

Um outro ponto a ser considerado é o nível de detalhamento do modelo. É importante recordar que nem sempre um nível maior de detalhe no modelo significará um aumento na concordância entre os dados simulados e os dados reais (GOODREAU; BACHMAN, 2000). Esse

nível ótimo de detalhamento é sempre difícil de ser obtido e depende da situação específica. Por exemplo, é comum se considerar as paredes do duto como estando a uma temperatura constante, mas alguns autores levam em consideração a troca de calor entre a parede do duto e o solo ao redor (TURNER; MUDFORD, 1988; CONDACSE; GUIDRY, 2002) e afirmam que isso influenciou positivamente os resultados. Também é preciso levar em consideração as incertezas nos dados de entrada do modelo e como essas incertezas se propagam até os resultados finais. Além disso, os custos de criação e manutenção de um modelo mais complexo devem ser comparados com as vantagens obtidas com esse aumento de complexidade.

Uma alternativa para se diluir os custos de desenvolvimento e/ou manutenção do modelo e de compra do *software* é utilizar o mesmo simulador e/ou modelo para outras atividades, como a previsão de demanda, estudo de cenários operacionais, etc. Esta é a abordagem de alguns pacotes de simulação como, por exemplo, o sistema utilizado pela empresa-alvo.

2.3.4 Análise pontual de pressão

O método de análise pontual de pressão é um dos mais simples (e também dos menos sensíveis) métodos de detecção de vazamentos por *software*. Ele pode ser considerado um refinamento, utilizando técnicas estatísticas, do método de detecção de variações abruptas de pressão (NICHOLAS et al., 1992).

O princípio de funcionamento do método é que a pressão no duto irá declinar em função de um vazamento. A partir disto pode-se realizar um teste de hipóteses para determinar se a pressão está declinando de maneira significativa. A grande desvantagem deste método está exatamente em que este teste pressupõe que a pressão em um determinado ponto no duto é um processo estocástico estacionário. Esta aproximação só é razoável quando o duto está operando em regime permanente. Desse modo, durante operação transiente este método não pode ser utilizado, sob pena de uma avalanche de alarmes falsos.

O método consiste em analisar os dados mais recentes adquiridos por um determinado sensor. Estes dados são divididos em duas partições, que podem ou não se sobrepor parcialmente, uma indo do ponto mais antigo existente na série até um ponto intermediário e a outra indo desse ponto intermediário, ou de um ponto próximo a esse, até o ponto mais recente. Então a média e a variância dessas duas partições devem ser calculadas. Deve-se suspeitar de um vazamento quando a média da partição que trata dos pontos mais recentes for significativamente (do ponto de vista estatístico) menor do que a média da partição mais antiga. Pode-se estabelecer como parâmetro de sensibilidade a confiança do teste estatístico. Quanto maior a confiança desejada no resultado, menor a sensibilidade do método.

Um dos modos utilizados para melhorar esse método é acoplá-lo a um modelo dinâmico e fazer a análise pontual de pressão sobre o resíduo entre os valores medidos e os valores previstos pelo simulador.

Devido a suas limitações, esse método funciona melhor quando aplicado a dutos pequenos que carregam fluidos incompressíveis por distâncias pequenas e operam em regime permanente. Nestes casos, o sistema é extremamente vantajoso, pois apresenta baixo custo e baixa complexidade.

2.4 Descrição de algumas implementações

Nesta seção será apresentada uma revisão bibliográfica sobre métodos de detecção de vazamentos por *software*. Dada a natureza acadêmica da maioria dos artigos estudados, salvo em algumas exceções, nenhum deles foi implementado na prática ou utilizado durante longos períodos por alguma companhia. Sendo assim, deve-se utilizar com cautela os resultados apresentados. No entanto, sempre que esses resultados estiverem disponíveis, eles serão apresentados.

As descrições a seguir são mais indicativas da realidade dos SDVs baseados em *software* que as descrições apresentadas na seção 2.3, pois demonstram a complexidade e o nível de detalhe que um método deve levar em consideração para que possa ser usado na prática, e também mostram como os vários princípios apresentados na seção 2.3 podem ser combinados e melhorados.

Para facilitar a leitura, os métodos foram agrupados pelas características mais marcantes de cada um. Mas vale reforçar que esse agrupamento é simplesmente didático, sem se arvorar a ser definitivo nem perfeito.

2.4.1 Métodos baseados em identificação de sistemas e sistemas dinâmicos

O princípio de funcionamento desses métodos consiste em, a partir das equações diferenciais que descrevem o sistema, produzir filtros ou observadores de estado que identifiquem um vazamento e, se possível, indiquem a posição do mesmo. Sendo assim, pode-se classificar estes métodos como baseados em modelo dinâmico.

No sistema desenvolvido por Billman & Isermann (1987) se utiliza um método de balanço de massa, utilizando um modelo dinâmico para calcular o empacotamento. Estes autores propõe duas inovações: um método para determinar o coeficiente de atrito do gás com a tubulação e

um método para decidir se está ocorrendo um vazamento baseado em correlação cruzada.

Partindo das equações dinâmicas do fluido, e supondo que o fluxo é turbulento e isotérmico, os autores discretizam essas equações utilizando o método das diferenças centrais. A partir desse modelo e utilizando-se o método dos mínimos quadrados é construído um observador de estados não-linear. Este observador busca determinar o valor dos coeficientes de atrito da tubulação e esses coeficientes são então utilizados para atualizar o modelo dinâmico.

Para determinar a presença de um vazamento é realizada a comparação entre o fluxo de massa na entrada e na saída da linha. Assim, sendo $q_i(k)$ a vazão medida no ponto i do duto e $\widehat{q}_i(k)$ a vazão prevista pelo modelo para esse ponto, pode-se gerar dois resíduos $x(k)$ e $y(k)$, onde $x(k) = q_0(k) - \widehat{q}_0(k)$ representa o resíduo entre a vazão medida e a vazão simulada na entrada e $y(k) = q_n(k) - \widehat{q}_n(k)$ representa a mesma medida na a saída do duto.

Para realizar a comparação entre os dois sinais, os autores utilizam o seguinte teste de correlação cruzada:

$$\phi_{xy}(\tau) = E\{x(k)y(k+\tau)\} \quad (2.3)$$

Nesta equação, τ representa um atraso qualquer. Como os dados são obtidos como séries temporais, $\tau \in \mathbb{Z}$. A aplicação deste filtro resulta, teoricamente, em

$$\phi_{xy}(\tau) = \begin{cases} 0 & \text{sem vazamento} \\ -f(q_L, x_L) & \text{com vazamento} \end{cases} \quad (2.4)$$

onde f é uma função qualquer de q_L , o fluxo do vazamento e x_L , a posição do vazamento.

Para calcular a função de correlação cruzada pode-se utilizar um filtro recursivo de primeira ordem:

$$\widehat{\phi}_{xy}(\tau, k) = \kappa \widehat{\phi}_{xy}(\tau, k-1) + (1-\kappa)x(k)y(k+\tau), \quad (2.5)$$

onde κ é uma constante entre 0 e 1.

Para reduzir os efeitos do ruído, o alarme de vazamento é tomado como o somatório sobre vários atrasos $\tau \in [-M; M]$, $M \in \mathbb{N}$. De acordo com os autores esta soma reage sensivelmente a pequenos vazamentos.

$$\phi_{xy\Sigma}(k) = \sum_{\tau=-M}^M \widehat{\phi}_{xy}(\tau, k) \quad (2.6)$$

Após a detecção do vazamento, para estimar a posição são utilizadas duas medidas de autocorrelação:

$$\phi_{xx\Sigma}(k) = \sum_{\tau=-M}^M \widehat{\phi}_{xx}(\tau, k) \quad (2.7)$$

$$\phi_{yy\Sigma}(k) = \sum_{\tau=-M}^M \widehat{\phi}_{yy}(\tau, k). \quad (2.8)$$

A posição do vazamento pode então ser estimada por:

$$\widehat{x}_L = \frac{L_R}{1 - \frac{\phi_{xx\Sigma}(k)}{\phi_{yy\Sigma}(k)}}, \quad (2.9)$$

e a quantidade de massa perdida no vazamento é

$$\widehat{q}_L(k) = E\{x(k) - y(k)\}. \quad (2.10)$$

Para testar esse sistema foi utilizada a simulação de uma linha de gás com as características mostradas na Tabela 2.1.

Tabela 2.1: Características da linha simulada em (BILLMAN; ISERMANN, 1987).

Comprimento	150.000 m
Diâmetro	0,26 m
Fluido	Não especificado
Estações de monitoração de pressão	2
Vazão nominal	Não especificado
Pressão nominal	Não especificada
Período de amostragem	3 min

Para esse problema as definições de $x(k)$ e $y(k)$ foram modificadas para:

$$\widetilde{x}(k) = x(k) - E\{y(k)\} \quad (2.11)$$

$$\widetilde{y}(k) = y(k) - E\{x(k)\} \quad (2.12)$$

Este sistema foi capaz de detectar um vazamento de 2,5% do fluxo de massa entre 50 e 100 amostragens, contando a partir do início do vazamento, com o duto em operação transiente.

Em uma outra abordagem, Benkherouf & Allidina (1988) propõe utilizar um modelo dinâmico do duto prevendo uma série de pontos de vazamentos e utilizar um filtro para identificar a vazão em cada um desses pontos. A partir deste modelo os autores mostram ser possível determinar a posição do vazamento real (caso este exista) e a vazão naquele ponto. Em uma situação

normal, onde não existe nenhum ponto de vazamento, o resultado da identificação é vazão nula para todos os pontos.

Para criar o modelo, os autores partem das equações diferenciais parciais que descrevem o comportamento dentro do duto supondo a existência de uma série de vazamentos, homogeneamente espaçados ao longo do duto. A partir destas equações, realizando uma discretização utilizando o método das características, chegam a um conjunto de equações diferenciais ordinárias. Estas equações são então linearizadas ao redor do ponto de operação atual e um filtro de Kalman é construído a partir desse modelo para estimar a vazão em cada um dos pontos de vazamento.

Os autores demonstram que uma média ponderada da distância entre cada ponto de vazamento e a quantidade de massa que o estimador determinou para aquele ponto mostra a localização do vazamento real, e que a soma da massa total vazando estimada pelo modelo indica a quantidade de massa que está vazando. Os autores não apresentam qualquer esquema de como deve ser tomada a decisão de vazamento/não-vazamento.

Os resultados dos testes numéricos realizados indicam um desempenho razoável. A simulação foi feita utilizando-se um simulador dinâmico e acrescentando ao sinal obtido um ruído branco³ de média 0 e variância conhecida. As características da linha simulada estão na Tabela 2.2.

Tabela 2.2: Características da linha simulada em (BENKHEROUF; ALLIDINA, 1988).

Comprimento	90 km
Diâmetro	0,785 m
Fluido	Um gás não especificado
Estações de monitoração de pressão	3
Vazão nominal	2.000 kg/s
Pressão nominal	10 MPa
Período de amostragem	Não informado

O sistema foi capaz de detectar um vazamento de 2% do fluxo de massa em regime permanente, 80 min após a instalação do vazamento. Como os autores não propõe nenhum método para a tomada de decisão sobre vazamento, esse tempo pode provavelmente ser melhorado.

Um sistema de detecção de vazamentos baseado em modelo dinâmico mais tradicional é apresentado em (FUKUSHIMA et al., 2000). Este sistema está em operação para detectar vazamentos em uma linha de transporte de GLP entre Niigata e Sendai (Japão). As características do duto estão mostradas na Tabela 2.3. O modelo utilizado neste sistema foi obtido pela discretiza-

³Ruído branco é um ruído em que cada frequência aparece com a mesma intensidade.

ção de um conjunto de equações diferenciais parciais, utilizando o método das características. Em cada passo de detecção, ao se decidir que não existem vazamentos, é utilizado um sistema de sintonia dos parâmetros do simulador para reconciliar os sinais simulados com os sinais medidos.

Tabela 2.3: Características do duto Niigata-Sendai (Japão).

Comprimento	250 km
Diâmetro	Não indicado
Fluido	GLP
Estações de monitoração de pressão	30
Vazão nominal	$4,5 \times 10^6 \text{ m}^3/\text{s}$
Pressão nominal	Não indicada
Período de amostragem	30 s

O princípio utilizado para a detecção de vazamentos é a conservação da massa nas fronteiras de cada elemento da malha de discretização. O sistema é capaz de detectar vazamentos da ordem de 1,1% do fluxo de massa, em cerca de 16 amostragens. O sistema apresenta um erro máximo de 20% e um erro mínimo de 4% na localização dos vazamentos e o menor vazamento cujo tamanho é possível estimar é de 1,8%.

2.4.2 Métodos baseados em análises estatísticas

Aqui serão mostrados alguns métodos cujo principal diferencial é o uso de análises estatísticas, ou sobre os sinais obtidos direto do SCADA, ou sobre os resultados obtidos de algum outro método.

Turner & Mudford (1988) propõe um método bastante interessante, baseado em modelos dinâmicos (ver seção 2.3.3), que faz um uso um pouco inusitado do simulador, para obter o empacotamento, e em seguida utiliza análises estatísticas para descontar o efeito de ruídos nos sensores e decidir se existe ou não um vazamento.

Para a simulação o modelo da linha é dividido em vários segmentos, cada um correspondendo a uma estação de medição de pressão e temperatura. Durante a simulação os valores de pressão e temperatura são utilizados como condições de contorno para esses segmentos, mas não se obriga que a massa fluindo entre dois segmentos se conserve.

Desse modo, na fronteira de cada segmento haverá um desbalanceamento R entre a massa calculada saindo do segmento anterior e a massa calculada entrando no segmento atual. Os autores argumentam que, devido a esse relaxamento nas condições de contorno de cada segmento, é possível utilizar dados de todos os sensores disponíveis, sem ser necessário ignorar os dados

de nenhum sensor. E que o uso da maior quantidade possível de dados experimentais permite que o resultado do cálculo do simulador para o empacotamento no segmento seja o mais preciso possível.

O simulador retorna o desbalanceamento de massa em cada segmento, seguindo um procedimento semelhante ao já detalhado na seção 2.3.3. Se considera que está ocorrendo um vazamento quando R for maior que um determinado limiar A . No entanto, os autores argumentam que, como este desbalanceamento é influenciado fortemente por erros de medição e erros advindos da simulação, deve ser utilizada a média de R sobre um determinado número de amostragens N e A deve ser uma função do erro padrão deste sinal.

O artigo ainda trata da localização do vazamento e da determinação da quantidade de produto vazada, mas não convém mostrar estes resultados por estarem fora do escopo deste trabalho.

Os resultados dos testes realizados pelos autores indicam um bom desempenho, sendo capaz de detectar um vazamento de 1% do fluxo de massa em aproximadamente 100 amostragens (ou seja, 150 min).

Os dados utilizados na simulação são de uma seção do gasoduto que vai de Dampier até Perth (Austrália). Para a detecção de vazamentos, ao sinal obtido na simulação original foi acrescentado um ruído branco de média 0 e variância de 3,5 vezes o desvio padrão do sinal. As características da linha simulada se encontram na Tabela 2.4.

Tabela 2.4: Características da linha simulada em (TURNER; MUDFORD, 1988).

Comprimento	134 km
Diâmetro	0,643 m
Fluido	Um gás não especificado
Estações de monitoração de pressão e temperatura	6
Estações de medição de vazão	2, no início e no fim do duto
Vazão nominal	100 kg/s
Pressão nominal	Não indicada
Período de amostragem	90 s

Subseqüentemente, Turner (1988) demonstrou um melhoramento no processo estatístico proposto anteriormente. Especificamente, para melhorar a utilização dos dados disponíveis, o autor mostra um algoritmo para associar a cada medição um determinado peso e demonstra qual é a melhor distribuição de pesos. O uso dessa técnica pode melhorar (ou seja, diminuir) em cerca de 3 vezes a taxa de alarmes falsos obtida em experimentos anteriores.

Em uma abordagem bastante distinta desta, Loparo et al. (1991) apresenta um método para detecção de vazamentos em trocadores de calor que, entretanto, é genérico o suficiente a ponto de poder ser aplicado a dutos.

A abordagem consiste em construir um modelo dinâmico para cada um dos possíveis modos de operação do sistema, incluindo operação normal e vários tipos de falha. O princípio do método é supor que o processo de falha é um processo de salto randômico entre um número finito de estados (cada estado representado por um modelo). Assim, é construído um filtro cuja saída é a probabilidade condicional de que o processo esteja operando em cada um dos estados possíveis. A detecção é realizada por uma função de limiar simples, que seleciona como modelo que melhor representa a planta o modelo com uma probabilidade (definida pelo filtro) acima de um determinado patamar.

Apesar do método proposto ser teoricamente interessante, os resultados obtidos pelos autores em um experimento foram muito pobres, provavelmente devido a ruídos nos sensores e imprecisões nos modelos.

Uma outra abordagem bastante interessante é a proposta por Wang et al. (1993). O princípio é comparar duas séries temporais, uma reconhecidamente sem vazamento e outra que se deseja testar, e verificar se elas correspondem ao mesmo processo estocástico. Esta comparação será feita utilizando uma medida que se chama informação de Kullback, ou informação mútua.

O método consiste em selecionar uma série temporal x_k^0 em um momento em que, sabidamente, não existem vazamentos, e uma série x_k^1 que será testada. Às duas séries é associado um modelo auto-regressivo (no teste os autores utilizaram ordem 5 para esse modelo). Depois é aplicado sobre esses modelos e sobre algumas características estatísticas do sinal um teste baseado na informação de Kullback⁴. O resultado desse teste é depois passado por um limiar para classificar a série como indicadora de um vazamento ou não.

O artigo parte de duas suposições fortes: a queda de pressão ao longo do duto é linear e os sinais de pressão podem ser perfeitamente modelados por um modelo de regressão linear mais um componente de ruído gaussiano com média 0 e desvio padrão σ .

O artigo mostra todas as deduções e testa o sistema em um duto de água de 120 m de comprimento, como mostra a Tabela 2.5.

O sistema foi capaz de detectar um vazamento de 0,5% após 100 amostragens e um vazamento de 1% após cerca de 50 amostragens.

⁴Também chamada, por abuso de linguagem, de distância de Kullback-Leibler, apesar de, tecnicamente, não ser uma distância.

Tabela 2.5: Características da linha de testes utilizada em (WANG et al., 1993).

Comprimento	120 m
Diâmetro	Não indicado
Fluido	Água
Estações de monitoração de pressão	4
Vazão nominal	Não indicada
Pressão nominal	Não indicada
Período de amostragem	20 ms

2.4.3 Métodos baseados em técnicas de inteligência artificial ou heurísticas

Um outro grupo de técnicas existentes são aquelas baseadas em inteligência artificial. Estas técnicas podem ser tanto fruto de heurísticas como de teorias de aprendizado automatizado. Estas serão apresentadas primeiro, enquanto aquelas, por serem de maior interesse para esse trabalho, serão apresentadas por último.

Um exemplo do uso de aprendizagem automatizada pode ser encontrado em (BELSITO et al., 1998). Neste artigo os autores treinam uma rede neural que, recebendo como entrada sinais dos sensores do gasoduto e de um simulador dinâmico, produz como saída o tamanho do vazamento no gasoduto. As saídas possíveis da rede são: 0%, 1%, 2%, 3%, 5% e 10% do fluxo de massa.

Os autores se propõe a detectar vazamentos de até 1% com 100 s de atraso. A linha de referência é uma linha de transporte de amônia líquüefeita. Os testes foram feitos sobretudo com dados obtidos de um simulador do duto de referência.

As características da linha de referência estão mostradas na Tabela 2.6

Tabela 2.6: Características da linha de referência utilizada em (BELSITO et al., 1998).

Comprimento	74,668 km
Diâmetro	0,203 m
Fluido	Amônia líquüida
Estações de monitoração de pressão	13
Vazão nominal	10,4 kg/s
Pressão nominal	2,22 MPa (saída); 2,82 MPa (entrada)
Período de amostragem	15 s

O artigo inicia com um estudo das características dos sensores, como erro e deriva. Depois é gerado um banco de dados de sinais de treinamento, teste e validação, a partir do simulador, sem estarem corrompidos com ruído. Os vazamentos simulados (e utilizados para treinar a rede neural) são de 0%, 1%, 2%, 3%, 5% e 10%. Uma comparação dos espectros dos sinais dos

simulador (corrompido com ruído gerado por computador) e do duto de referência é feita para validar o ruído incluso no sinal do simulador.

A rede neural utilizada foi uma *perceptron* de 3 camadas, com função de ativação sigmoidal. Inicialmente a rede é treinada utilizando-se sinais sem ruído, por apresentar uma convergência mais rápida. Como entrada da rede foram utilizados os sinais dos dois medidores de fluxo, colocados no início e no final da linha, e dos 13 medidores de pressão. Os sinais foram normalizados para média 0 e desvio padrão 1. A saída da rede é um valor normalizado entre 0,1 a 0,9, indicando o tamanho do vazamento (de 0% a 10%).

Com a introdução de sinais com ruído foi necessário utilizar um filtro de média móvel (com 9 pontos) sobre os sinais de entrada na rede e acrescentar informação sobre a deriva dos medidores de fluxo: isso foi introduzido como uma diferença entre o valor dos fluxômetros de entrada e saída em $t - \tau$. Por isso, a rede foi expandida para aceitar mais uma entrada, ficando com 16 entradas no total.

Durante o treinamento da rede com sinais apresentando pequenos transientes se fez necessário incluir a saída do modelo dinâmico como mais uma entrada da rede neural.

Tabela 2.7: Desempenho do sistema de localização de vazamentos de (BELSITO et al., 1998).

Tamanho do vazamento (% da entrada)	1	2	5	10
Tempo de detecção (pontos)	8	6	4	3

A rede mostrou-se sensível ao número de estações de medição de pressão. Também se mostrou bastante sensível ao caso de sensores de vazão defeituosos.

O uso de RNAs para a detecção de vazamentos parece ser uma abordagem interessante, porém neste artigo os autores desprezaram uma série de conhecimentos já presentes sobre detecção de vazamentos, ao invés de associar uma RNA a outros métodos para tentar refinar a resposta.

Já em Konvalinka et al. (1992) está descrito um sistema baseado em aprendizado que foi testado em um gasoduto real. O artigo descreve a solução utilizada para a detecção de vazamentos no “DP Gas”, na antiga Iugoslávia. A rede possui cerca de 1300 km de extensão e funciona com uma pressão entre 25 e 42 bar. Tem 250 clientes (ou seja, pontos de saída) e mais de um fornecedor (o artigo não especifica quantos). O período de amostragem do SCADA é de 30 s. Os modelos matemáticos utilizados para simular a rede não incluem efeitos dinâmicos. Quando a rede está operando neste regime o modelo consegue calcular a vazão mássica e a pressão nos pontos de medição com um erro máximo de 1%.

Os autores levam em consideração que, a princípio, se poderia criar um SE para determinar a presença de vazamentos. No entanto, o conhecimento necessário para a implementação do SE não estava disponível quando da realização do projeto.

O método adotado pelos autores para resolver o problema foi utilizar algum método de aprendizado. Foi escolhido o aprendizado por indução em uma árvore de decisão (RUSSELL; NORVIG, 2004, cap. 18), por permitir uma melhor visualização do conhecimento gerado e por fornecer explicações razoáveis do processo de raciocínio.

A aplicação deste método implica três coisas: criar um corpo de exemplos que estejam classificados corretamente, fornecer um corpo de características úteis extraídas a partir dos sinais temporais do simulador e da planta (extração de característica) e gerar árvores de decisões a partir destes dados.

Para gerar os dados de vazamentos foi utilizado um meio bastante engenhoso. Se procurava no banco de dados histórico da rede de gás uma série em que, dentro de um certo período de tempo (t_1, t_2), todos os fluxos e pressões em um determinado trecho da rede estivessem aproximadamente constantes, com exceção de um ponto, que será chamado de ponto j (veja Figura 2.1). Quando esses sinais eram encontrados, o simulador era alimentado com todos os dados reais, a não ser com os valores do ponto j . Neste ponto, a vazão e a pressão eram mantidas fixas, com um valor consistente com uma vazão menor do que a que estava sendo efetivamente medida. Deste modo, simulava-se um vazamento de gás em alguma região do tramo que ligava ao ponto de medição j .

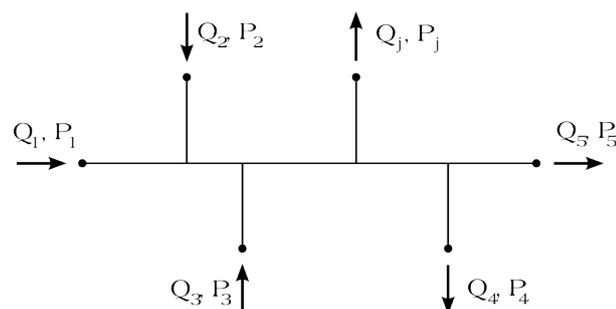


Figura 2.1: Trecho de tubulação utilizado em (KONVALINKA et al., 1992) para simular um vazamento. Na figura, as setas indicam a direção do gás e os pontos marcados com Q_i, P_i são locais onde se tem leituras de pressão e temperatura.

Dependendo do tamanho do vazamento simulado, eles foram classificados em três tipos: pequenos ($< 5\%$), médios (de 5% até 20%) e grandes ($> 20\%$). Os tipos de vazamento também foram classificados em três tipos: vazamentos de crescimento lento, vazamentos de crescimento rápido e vazamentos constantes. Foram recolhidos 11.000 exemplos através desse método, que

foram separados em 70% para treinamento e 30% para teste.

Com relação à extração de características, foi utilizado um conjunto de 16 transformações diferentes sobre os sinais obtidos a partir do simulador e do SCADA. Estas transformações foram escolhidas baseadas na intuição dos projetistas. Durante o treinamento, observou-se que algumas das análises eram irrelevantes.

O treinamento foi realizado utilizando-se o *software* “Assistant Professional”. Foi utilizado um processo puramente experimental de desenvolver árvores de decisão e depois testá-las com relação a taxa de acerto. Os resultados obtidos foram os seguintes:

1. O uso de uma única árvore de decisão para todo o gasoduto apresentou uma taxa de acerto muito baixa. O melhor resultado foi na classificação de vazamento/não-vazamento onde se obteve uma taxa de acerto de 85%;
2. O desenvolvimento de uma árvore de decisão para cada estação de medição produziu os melhores resultados, onde se obteve uma taxa de acerto mínima de 75% nas regiões da linha com menor fluxo e um máximo 94%, isso tanto na detecção quanto na classificação do vazamento.
3. O tempo de detecção é fortemente dependente da taxa de atualização do SCADA. Como no sistema foi utilizada uma taxa de atualização de 10 min, o tempo mínimo de detecção era de 2 h. Isso se deve principalmente à necessidade de um certo número de dados para se avaliar médias e espectros, por exemplo.

Um outro exemplo prático da aplicação de inteligência artificial para detecção de vazamentos é o apresentado por Kerkhof et al. (2002). Ao contrário dos sistemas apresentados anteriores, este é baseado em sistemas especialistas e não em aprendizado de máquina.

Este sistema foi implementado para monitorar o “TransCanada Pipeline”, composto por cerca de 38.000 km de dutos, em alguns casos com até 7 linhas paralelas. O objetivo do sistema como um todo é o de ser um sistema “consultor”, capaz de fornecer informações complementares às fornecidas pelo SCADA e de diminuir a pressão sobre os operadores auxiliando no monitoramento dos estados dos sensores do gasoduto.

Uma das funções realizadas por esse sistema é o monitoramento de anomalias de pressão, que podem ser indicativas de vazamentos ou fechamento de válvulas de bloqueio, e busca de padrões que podem indicar um vazamento. A implementação desse sistema foi feita utilizando-se ao mesmo tempo regras e processamento de sinais. O módulo de processamento de sinais

reduz os sinais lidos do SCADA a uma série de características que são então passadas ao sistema especialista. Este contém regras que, a partir destas características e de relações esperadas entre os valores de cada estação de medição, detectam uma série de condições operacionais de interesse.

2.5 Deficiências de sistemas de detecção de vazamentos por *software*

Como foi visto nas seções anteriores, em geral os SDVs por *software* utilizam uma combinação de técnicas de análise estatística e simulação dinâmica para determinar a presença de um vazamento. Também foi visto que o desempenho de cada técnica varia bastante de acordo com a implementação. Ainda assim, a maior dificuldade na detecção de vazamentos por *software* está na baixa qualidade dos sinais de entrada (ENERGY, 2004).

Como essas análises estão baseadas em informações provenientes de sensores que foram instalados no duto por outras razões, esses sensores não foram otimizados para a tarefa de detecção de vazamentos, uma vez que as outras tarefas de gerenciamento e controle do gasoduto impõe restrições mais brandas na qualidade e quantidade das medições do gasoduto.

Assim, são poucos os sistemas que se dizem capazes de detectar vazamentos menores que 1% do fluxo de massa de um gasoduto (BELSITO et al., 1998), mesmo em se tratando de dutos de transporte de líquidos (que, devido ao grande módulo de compressibilidade, permitem uma sensibilidade maior). Mesmo entre os produtos comerciais, nenhum dos pesquisados se disse capaz de detectar vazamentos menores que 0,5% do fluxo de massa, mesmo com as melhores condições possíveis.

Apesar desses números parecerem razoáveis, é preciso notar que em um duto de transporte de gás de grande capacidade um vazamento de 1% pode significar o mesmo consumo que uma estação de entrega e que mesmo um vazamento com metade dessas dimensões seria prontamente detectado por qualquer comunidade próxima, devido ao intenso ruído que o gás produz ao escapar para a atmosfera.

Portanto, devido a essas limitações e apesar de serem usados com frequência na prática, a eficácia operacional dos SDVs por *software* ainda é um tema controverso na indústria (NICHOLAS et al., 1992; WESTHOFF, 1999).

Não obstante esses problemas, deve-se levar em consideração que durante a operação normal de um gasoduto os controladores realizam alguns testes e observam alguns sinais específi-

cos para detectar vazamentos, normalmente baseados em heurísticas sobre o funcionamento do gasoduto. Portanto, uma alternativa de desenvolvimento ainda relativamente pouco explorada é mimetizar em um *software* as ações desses controladores, de modo a realizar uma parte da tarefa e diminuir o estresse sobre eles.

2.6 Sistema instalado na empresa-alvo

O sistema de detecção de vazamentos utilizado pela empresa-alvo faz parte de um pacote de administração do gasoduto chamado PMS (*Pipeline Management System*), fornecido pela *Energy Solutions International*.

O princípio de funcionamento do PMS é bem simples: fazer a diferença entre os valores medidos no gasoduto e os valores obtidos através de simulação, e tentar identificar nestas diferenças evidências de um vazamento. Os valores de entrada para a simulação são pressão, vazão e temperatura. Para se obter uma boa sensibilidade na detecção de vazamentos é importante que esses valores sejam precisos.

Por esta razão, o sistema utiliza principalmente os dados das estações de compressão, porque todas possuem medidores de pressão e a maior parte delas possui medidores de vazão (em algumas estações, por um erro de projeto, o medidor de pressão está dentro da região do *bypass*, tornando-se inativo sempre que a estação está com todos os compressores desligados). Já as estações de entrega não possuem medidores da vazão através do gasoduto, apenas da vazão para o cliente. Além disso, o sistema de comunicação das ECOMPs é do tipo VSAT, que permite uma frequência de transmissão de dados grande, de cerca de uma a cada 3 segundos, ou por uma conexão terrestre. Já as estações de entrega funcionam com o sistema Inmarsat, que é basicamente uma linha discada via satélite. O custo de transmissão por esse sistema é mais alto, de modo que as comunicações entre EEs e a CSC são bem menos frequentes.

Para que o sistema seja eficaz, deve ser possível fazer simulações bastante rápidas, de modo a permitir verificações frequentes (da ordem de uma a cada 20 segundos). Para atingir esta taxa é necessário utilizar um modelo simplificado do duto. Por exemplo, no modelo utilizado pela empresa-alvo as estações de entrega são consideradas derivações simples.

Para fazer a simulação, é preciso dividir as variáveis de processo em dois grupos: umas, as conhecidas, serão alimentadas no modelo e as outras serão previstas por ele. Assim, para se utilizar todas as variáveis de processo, o PMS utiliza dois modelos, chamados de PQ e QP.

No modelo PQ são definidas como variáveis conhecidas a pressão a montante (i.e. na ECOMP a montante do trecho de interesse, quando essa ECOMP possui um sensor de vazão fora do *by-pass*) e a vazão a jusante. Assim o modelo deve prever a vazão a montante e a pressão a jusante. No modelo QP é feito o inverso: são definidas como variáveis de entrada a vazão a montante e a pressão a jusante e se busca a pressão a montante e a vazão a jusante.

Desse modo, é possível fazer comparações de todas as pressões e vazões medidas com as simuladas. Essas comparações resultam em 4 filtros: o filtro de pressão inesperada (UP – *unexpected pressure*) e de vazão inesperada (UF – *unexpected flow*), para a estação a montante e a jusante. Cada um desses filtros é calculado subtraindo-se do valor esperado (ou simulado) o valor medido. A figura 2.2 explica essa comparação.

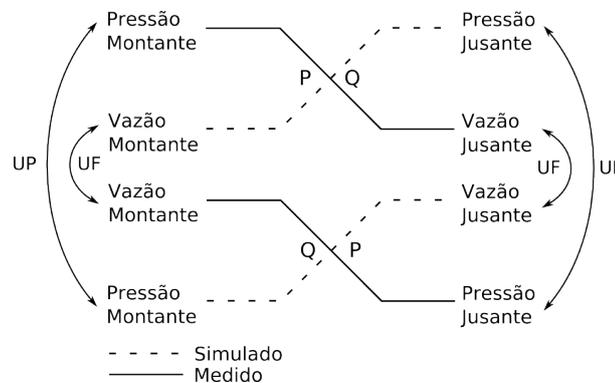


Figura 2.2: Esquema do funcionamento dos filtros PQ e QP.

Além disso, um outro parâmetro que deve ser definido é o tempo de simulação. São utilizados 3 tempos de simulação diferentes para cada filtro, totalizando desse modo 12 filtros. Os filtros são chamados de *slow*, *medium* e *fast*, para tempos de simulação de 100, 30 e 5 min, respectivamente.

Apesar do menor tempo de simulação ser de 5 min, isso não quer dizer que só ocorra uma simulação a cada 5 min. O que ocorre é que a cada ciclo de aquisições (20 s) é disparada uma simulação do tipo *slow*, uma do tipo *medium* e uma do tipo *fast*, para cada um dos modelos, PQ e QP, ou seja, 6 simulações diferentes. Desse modo, a cada 20 s o valor dos desbalanceamentos apresentados por esses doze⁵ filtros é atualizado.

De posse destas informações, o programa então as compara com um limiar. Esse limiar é calculado em duas partes: a base e o complemento.

A base é definida previamente através de um estudo, chamado de LSS – *Leak Sensitivity Study*, feito pela *Energy Solutions*, que determina qual o tamanho máximo do desbalancea-

⁵Cada simulação produz 2 filtros: um para a estação a montante e outra para a estação a jusante.

mento sem que haja suspeita de vazamento. Valores acima desse parâmetro podem indicar um vazamento.

O complemento é calculado pelo PMS, para cada tempo de simulação (*slow, medium e fast*) e tipo de filtro (UF e UP), levando em consideração condições operacionais, como ocorrência de transientes na linha, utilizando métodos próprios. Esse complemento é então adicionado ao limiar base.

A princípio, detectar se um desses limiares foi ultrapassado já faria um sistema de detecção de vazamentos razoável. No entanto, ele seria muito suscetível a alarmes falsos, caso ocorra uma situação em que apenas um dos filtros ultrapasse o limiar em apenas uma das simulações.

Para diminuir a possibilidade de alarmes falsos é utilizado ainda um sistema de contagem de vezes em que o sistema ultrapassou o limiar. Cada filtro tem associado a si um contador. Este contador pode ter os valores -1, 1, 2, 3 e 4, e inicia em 1. A cada vez (i.e. a cada período de 20 s) que o filtro apresenta um valor acima do limiar, o contador é incrementado. Do mesmo modo, cada vez que o desbalanceamento fica abaixo do limiar, o contador é decrementado. Quando o desbalanceamento se torna negativo, o contador decresce para -1.

Desse modo, tem-se uma escala crescente de alertas de vazamento. De acordo com o manual (VALERIANO, 2001), o nível 2 significa vazamento possível; o nível 3 alerta de vazamento; o nível 4 vazamento confirmado; -1 significa empacotamento (ou seja, um desbalanceamento negativo) e o nível 1 é situação normal. Através desse sistema é possível diminuir o número de alarmes falsos.

Assim que pelo menos 3 contadores a jusante e 3 contadores a montante estiverem com nível 4, o sistema inicia o processo de localização de vazamentos. Por ser um processo iterativo, conforme o tempo passa, o sistema melhora sua estimativa da posição do vazamento. O sistema de localização de vazamento retorna três números, em quilômetros: um limite inferior para a posição do vazamento, um limite superior e a posição mais provável.

A partir desta visão geral do funcionamento do sistema, existem alguns detalhes que precisam ser notados. O primeiro deles se refere à velocidade de resposta de cada um dos tipos de filtro: UF ou UP. Os filtros de vazão (UF) são muito mais sensíveis que os filtros de pressão, e apresentam uma resposta muito mais rápida. Por essa razão, eles são extremamente sujeitos a transientes, e costumam disparar com uma frequência muito maior que os filtros de pressão (UP). É comum um transiente acionar todos os três filtros de vazão em uma determinada estação, esses filtros voltarem para o estado normal e só então o filtro lento de pressão acusar alguma coisa. Para verificar se o valor alto para um dos filtros de pressão em uma estação está

relacionado com um valor alto no filtro de vazão, é preciso observar o histórico dos filtros de vazão até 2 horas antes.

Como os filtros de vazão são muito sensíveis a transientes, qualquer estado de alarme em um (ou mesmo nos três) filtro de vazão de uma estação deverá ser ignorado. Apenas se algum filtro na estação complementar também estiver em estado de alarme deverá o estado desses filtros ser levado em consideração.

No entanto, apesar dessa sensibilidade, não devem existir inconsistências entre filtros. Isso quer dizer que, por exemplo, o filtro de vazão de uma estação não pode indicar um valor de empacotamento e o filtro de pressão um valor de vazamento. Não obstante, em alguns casos, um alarme negativo na pressão e um positivo na vazão podem indicar uma situação em que o gasoduto está empacotando e houve um aumento de vazão naquela ECOMP.

Caso alguma inconsistência seja identificada, a maior probabilidade é de que se trata de sensores defeituosos ou dados forçados. Um dado é forçado quando, como foi explicado na seção 1.2, a CSC fixa o valor de um dos sensores do SCADA, de modo que a leitura realizada em campo seja sempre ignorada e apenas o valor fixado é registrado no banco de dados. Essas situações devem ser sempre avisadas, pois podem indicar algum defeito ainda não identificado pelos controladores.

Em geral, dados defeituosos que já são conhecidos pelo SCADA (como dados forçados ou leituras incorretas) são informados para o PMS, que ignora esses dados e os emula, ou utilizando uma média histórica, ou inferindo o valor a partir de simulação. O sistema também aumenta o nível do limiar, de modo que esse dado não cause alarmes falsos. No entanto, um erro de calibração de sensor de vazão em uma EE pode aparecer como uma suspeita de vazamento. Para dirimir essa dúvida, deve-se buscar as seguintes informações:

- O *trend* de vazão da EE. Caso a vazão esteja crescendo muito rapidamente, ou tenha crescido muito recentemente, a probabilidade de que seja um defeito de instrumentação ou um vazamento após os medidores de vazão da EE é grande;
- Caso o *trend* de vazão da EE indique um aumento brusco, deve-se confirmar esse aumento com o cliente. Isto pode ser indicação de um problema na linha do cliente, ou simplesmente de que ocorreu um aumento não previsto do consumo;
- Também deve-se rever o registro de calibração dos medidores de vazão da EE em questão. Caso os medidores tenham sido calibrados a muito tempo pode-se desconfiar de um erro de calibração.

2.7 Conclusões do capítulo

Pelo que foi mostrado neste capítulo, fica claro que a detecção de vazamentos por *software* é uma área bem desenvolvida, com técnicas bastante elaboradas. No entanto, como foi visto na seção 2.5, existem grandes limitações na capacidade desses sistemas de ter sua sensibilidade aumentada. Por outro lado, uma via de desenvolvimento que até o momento não foi muito explorada é a possibilidade de, às análises realizadas pelo detector de vazamento, acrescentar outras que simulem, ou se aproximem, das análises que os controladores realizam durante a operação do duto.

Pode-se considerar que, nesse caso, está se desenvolvendo um novo detector de vazamentos, com mais conhecimento agregado e capaz de fornecer informações mais úteis aos operadores. Considera-se que isto justifica a abordagem proposta nesta dissertação, qual seja: aplicar a um sistema de detecção de vazamentos tradicional (nesse caso, o instalado na empresa-alvo) um sistema especialista capaz de analisar a saída desse sistema e, a partir de heurísticas e conhecimento operacional fornecido pelos próprios controladores do gasoduto, fornecer mensagens mais informativas aos operadores. Considera-se que um sistema desse tipo fornece ainda as seguintes vantagens (SAMPATH; YEE, 2000; JOHNSON, 1999; WALLOPPILLAI et al., 2000):

- Diminuição da carga de trabalho dos controladores, pois realizará parte das análises mais rotineiras;
- Melhorar a resposta da organização a situações de falha do gasoduto, uma vez que o sistema pode tornar disponíveis rapidamente várias informações úteis aos controladores;
- Identificar até pequenos problemas com o SCADA e/ou instrumentação.

No próximo capítulo será feita uma revisão dos conhecimentos específicos de inteligência artificial que foram utilizados no desenvolvimento do sistema.

3 Sistemas baseados em conhecimento

Como foi mencionado anteriormente, este trabalho pretende criar um sistema especialista (SE) para auxiliar na análise dos dados produzidos pelo detector de vazamentos da empresa-alvo. Sendo assim, este capítulo visa introduzir as técnicas de inteligência artificial (IA) que foram utilizadas no desenvolvimento do protótipo.

O capítulo está organizado do seguinte modo: inicialmente, é feita uma breve introdução sobre engenharia de conhecimento, abordando apenas os fatos relevantes para o desenvolvimento desse sistema. Em seguida, são apresentadas as técnicas de representação de conhecimento e inferência que foram utilizadas no projeto. E finalmente se trata um pouco das ferramentas computacionais utilizadas na implementação do sistema.

3.1 Engenharia do conhecimento

O campo de estudo da inteligência artificial abarca inúmeras áreas, que pesquisam, por exemplo, a representação de conhecimento, métodos de inferência, aprendizado, etc. Uma destas áreas, que acabou se desenvolvendo em uma disciplina própria, é a engenharia de conhecimento. Enquanto boa parte da pesquisa em inteligência artificial se voltava apenas para os aspectos computacionais, quando se iniciaram as primeiras aplicações práticas, na década de 70 e 80, percebeu-se a necessidade de solucionar, metodicamente, uma série de problemas ainda pouco explorados.

Estes problemas incluem, por exemplo, o gerenciamento do desenvolvimento dos sistemas e a aquisição do conhecimento que deve ser implementado nos mesmos. A engenharia de conhecimento é a disciplina que nasceu do estudo das soluções desses problemas. Ela é um ponto de contato entre a inteligência artificial e as teorias de administração, do mesmo modo que ocorreu com a engenharia de *software*, que se desenvolveu em um ponto de contato entre a administração e as ciências da computação, quando se iniciaram as aplicações industriais de *software* (SOMMERVILLE, 1995).

As aplicações práticas da inteligência artificial que deram origem à engenharia de conhecimento são chamadas de sistemas especialistas (SEs), ou de sistemas baseados em conhecimento. Estes sistemas são chamados assim pois visam implementar o mesmo tipo de análise que um especialista realiza. Isto porque descobriu-se que a especialização humana tende a ser sempre em um campo bem definido e normalmente bastante restrito (DYM; LEVITT, 1991). Essa característica permite ao mesmo tempo delimitar bem o conhecimento necessário para implementar o sistema e fazer com que o valor comercial do sistema seja grande, pois disponibiliza um conhecimento normalmente restrito a poucos especialistas dentro de uma empresa para um número maior de pessoas, efetivamente fazendo com que esse conhecimento possa produzir mais valor (SCHREIBER et al., 1999).

É interessante conhecer a definição de *conhecimento* no contexto da engenharia de conhecimento, e como este conceito se opõe a *dados e informação*. Isto auxilia a entender a diferença fundamental entre os sistemas baseados em conhecimento e outros sistemas tradicionais, como um sistema de controle de estoques, por exemplo. Assim (SCHREIBER et al., 1999, págs. 3-4, tradução nossa):

Dados: Dados são sinais *não interpretados*. Por exemplo, a luz vermelha em um farol de trânsito é um dado;

Informação: Informação são dados com *significado*. Assim, saber que a luz vermelha do sinal significa que é obrigatório parar é informação, ou seja, *o saber do significado do dado faz com que ele carregue informação*;

Conhecimento: Conhecimento é o conjunto total de dados e informações dos quais as pessoas fazem *uso prático em ações*, para levar a cabo tarefas e criar mais informação. Sendo assim, o conhecimento tem dois aspectos distintos: primeiramente, um senso de *propósito*, pois o conhecimento faz parte do maquinário intelectual para atingir alguma meta; em segundo lugar, uma *capacidade* de gerar mais informação.

No entanto, os mesmos autores alertam para o fato dessas definições serem úteis apenas na medida em que auxiliam a compreensão de um determinado problema e que a diferença entre dados, informação e conhecimento é inerentemente dependente do contexto. Apesar disso, na definição fica clara a característica principal de conhecimento, que é a capacidade de gerar mais informação. Entender esse fato ajuda a entender o que talvez seja a tarefa mais difícil no desenvolvimento de sistemas especialistas: a aquisição de conhecimento.

Como foi dito anteriormente, sistemas especialistas são implementados para realizar a mesma tarefa que um especialista humano. Sendo assim, dois processos fundamentais no de-

envolvimento do sistema são a aquisição de conhecimento do especialista e a representação abstrata desse conhecimento para que ele possa ser implementado no sistema.

Existem diferentes visões sobre como esses processos devem ser realizados. A visão nas décadas de 70 e início da década de 80 era de que o processo de desenvolvimento de um sistema especialista é um processo de transferência de conhecimento do especialista para a base de conhecimento. Este processo assume que o conhecimento que é requerido pelo sistema especialista já existe e necessita apenas ser coletado e implementado. Tipicamente, esse conhecimento era implementado em sistemas baseados em regras. Com o passar do tempo, uma análise cuidadosa de vários sistemas especialistas mostrou que as regras de produção eram utilizadas para representar diferentes tipos de conhecimento, como por exemplo: informações sobre a ordem em que as metas devem ser atingidas estava combinada com informações específicas do domínio de conhecimento. Essa combinação de diferentes tipos de conhecimento torna a manutenção do sistema difícil e lenta (STUDER et al., 1998). Além disso, sistemas baseados em regras costumam apresentar dificuldade de separação entre as diferentes funções que o conhecimento implementado exerce. Assim, apesar de regras serem bastante modulares, a base de conhecimento pode ficar pouco modular devido à interdependência entre as regras. Isto contribui para a dificuldade de manutenção e expansão dos sistemas.

Além disso, percebeu-se que a premissa da abordagem de transferência estava errada, pois o conhecimento tácito do especialista tem uma função muito importante em sua capacidade de resolver problemas. Todas estas desvantagens resultaram na mudança da abordagem de transferência para a abordagem de modelagem. Assim, atualmente, é consenso que, para desenvolver um sistema especialista, é preciso desenvolver um modelo computacional com capacidades de resolução de problemas comparáveis a do especialista (dentro do domínio de especialidade). É importante perceber que o objetivo não é construir um modelo adequado do ponto de vista cognitivo, mas sim um modelo que produza resultados similares aos do especialista. Isso porque o especialista não está consciente de grande parte de seu próprio conhecimento e portanto não é capaz de transferi-lo. Assim, o foco do processo de aquisição de conhecimento passou a ser este processo de construção de um modelo (STUDER et al., 1998).

Também durante a década de 80, análises feitas sobre sistemas especialistas já implementados mostraram que existiam, apesar das diferenças nos tipos de representação de conhecimento e mecanismos de inferência utilizados, semelhanças nos métodos utilizados para resolver os problemas. O primeiro estudo desse tipo foi realizado por Clancey (1985), que, estudando vários sistemas especialistas, descobriu um padrão que foi chamado de *classificação heurística*. Com o passar do tempo, outros métodos de solução foram descobertos e surgiram bibliotecas

de *métodos de solução de problemas*. Isso permitiu que sistemas especialistas pudessem ser construídos a partir de *componentes reutilizáveis*, o que torna o desenvolvimento mais rápido e livre de erros. Além disso, a existência desses métodos de solução pode guiar o processo de aquisição de conhecimento e modelagem, uma vez que, identificado o método de solução, o tipo de conhecimento necessário para a implementação do sistema já estará bem definido (STUDER et al., 1998).

Quanto ao processo de desenvolvimento do sistema, é consenso que o melhor método de desenvolvimento para um sistema especialista é o desenvolvimento baseado na espiral de Boehm (ALONSO et al., 2000; SCHREIBER et al., 1999). Neste método de desenvolvimento, o sistema passa por sucessivas iterações de especificação, aquisição e modelagem do conhecimento, implementação, testes, geração de protótipos e avaliação do protótipo. Estes ciclos devem ser em rápidos, preferencialmente com poucos meses entre cada ciclo.

3.2 Representação de conhecimento e inferência

Historicamente, o foco da comunidade de inteligência artificial oscila entre dois assuntos: as teorias de conteúdo e as teorias de mecanismo. As teorias de conteúdo versam sobre o modo como o conhecimento pode ser representado em um programa e as teorias de mecanismo determinam o modo como novas conclusões podem ser obtidas a partir do conhecimento representado. Sem dúvida, essa separação nem sempre é clara, sendo que alguns mecanismos de inferência funcionam melhor com algumas técnicas de representação e vice-versa.

Não obstante, esta divisão é útil, pois permite considerar diferentes métodos de representação e mecanismos como possíveis componentes do sistema, cada um com suas vantagens e desvantagens. Pode-se então selecionar o componente mais adequado, ou a combinação mais adequada de componentes para a solução do problema. Sendo assim, o objetivo desta seção é apresentar as diferentes teorias de conteúdo e mecanismo utilizadas no desenvolvimento do projeto.

Para a representação de conhecimento foram utilizadas ontologias, quadros e redes semânticas. Na verdade, essas três teorias têm tantos pontos de contato, e, historicamente, seu desenvolvimento esteve tão conectado, que é praticamente impossível tratar de uma sem levar em consideração pelo menos alguns aspectos das outras duas. Por conta disso, essas três teorias estão apresentadas juntas na seção 3.2.1.

Para realizar a inferência, foram utilizadas dois mecanismos: regras de produção e lógica difusa. A descrição desses mecanismos está apresentada nas seções 3.2.2 e 3.2.3, respecti-

vamente. Vale a pena comentar que, apesar destes dois mecanismos não serem normalmente vistos como teorias de conteúdo, de fato existe conhecimento representado nestes mecanismos. O conhecimento implementado é o conhecimento necessário para a resolução do problema, ou seja, o método de solução.

No sistema que foi implementado, estes dois mecanismos são utilizados concomitantemente, junto com técnicas de programação mais tradicionais, como programação orientada a objetos. Desse modo, é possível utilizar um para complementar as deficiências no outro, obtendo assim um sistema mais flexível e mais expressivo.

Na literatura existem exemplos de que a integração entre as várias teorias, tanto de representação quanto de mecanismo, pode fornecer melhores resultados. Um exemplo de um ambiente de desenvolvimento que permite especificamente a integração entre diversos paradigmas é o ambiente BEST (VRANEŠ; STANOJEVIĆ, 1995), que integra sistemas baseados em regras de produção, com programação lógica, quadros, lógica difusa e técnicas de programação procedural e orientada a objetos. Um exemplo de um sistema especialista desenvolvido em áreas de aplicação próximas do presente projeto pode ser visto em (BIAGETTI; SCIUBBA, 2004). O objetivo deste sistema é realizar o diagnóstico e produzir prognósticos de uma planta de cogeração a gás natural. Neste sistema foram utilizados regras de produção e lógica difusa, além de técnicas matemáticas de análise de sinais.

3.2.1 Ontologias, quadros e redes semânticas

Para explicar estas três teorias de representação de conhecimento e a relação entre elas, é interessante iniciar pelo conceito de ontologia pois este é, de certo modo, mais genérico que os conceitos de quadro e de rede semântica.

O termo ontologia foi emprestado pela IA da filosofia, onde foi originalmente cunhado. Na filosofia, ontologia é a teoria dos objetos conhecidos e cognoscíveis (MORENTE, 1967). Ou seja, ontologia é o ramo da filosofia que estuda os objetos e suas relações em geral, sem particularizar ou restringir seus estudos a nenhuma classe específica de objetos. Esta definição pode ser comparada com a definição de ontologia dentro da IA (CHANDRASEKARAN et al., 1999, tradução nossa):

“Ontologias são teorias de conteúdo sobre os tipos de objetos, suas propriedades e as relações entre os objetos que são possíveis dentro de um determinado domínio de conhecimento. Elas fornecem termos para descrever o conhecimento sobre o domínio.”

Assim, em IA, ontologias descrevem, formalmente, conceitos que podem ser utilizados pelos programas. Uma ontologia define tanto um vocabulário de representação, normalmente específico a um determinado domínio de conhecimento, quanto um corpo de conhecimento descrevendo algum domínio (CHANDRASEKARAN et al., 1999).

É importante notar que não é o vocabulário propriamente dito que define uma ontologia, mas as conceitualizações dos termos nos vocabulários. Desse modo, traduzir uma ontologia de uma língua para outra não deve alterá-la conceitualmente, apesar de que, como argumenta Hofstadter (1979), isso não é verdade para o caso geral, pois podem não existir mapeamentos perfeitos para o mesmo conceito em diferentes línguas. Uma “conceitualização” é um modelo abstrato de algum fenômeno no mundo real. Um aspecto importante de uma ontologia é que as conceitualizações que ela representa devem ser consensuais, ou seja, dentro da área de conhecimento sobre a qual versa a ontologia, vários especialistas devem concordar com as definições dos conceitos presentes na ontologia (STUDER et al., 1998).

Da explicação anterior se percebe que ontologias são ferramentas bastante genéricas. Elas podem ser utilizadas tanto para armazenar informações que serão utilizadas apenas por seres humanos, como para garantir um vocabulário comum entre vários especialistas de uma área ou como uma ferramenta de gerenciamento de conhecimento, armazenando a memória corporativa de uma empresa. Também, caso seja adequadamente desenvolvida e genérica o suficiente, uma ontologia pode ser utilizada por diferentes sistemas especialistas que tratam da mesma área.

Para tornar a discussão menos abstrata, a Figura 3.1 exemplifica o tipo mais simples de ontologia, uma taxonomia. Na figura, as linhas tracejadas representam relações do tipo “é-um”. Esta ontologia representa algum conhecimento sobre jogos e poderia ser usada, por exemplo, por um sistema de recomendação inteligente de uma loja de brinquedos. Em um processo de aquisição de conhecimento, esta ontologia pode ser parte do resultado das primeiras entrevistas de aquisição.

Vale a pena relatar um fato historicamente interessante sobre este tipo de ontologia: Borges (1981) mostra em seu ensaio “*El idioma analítico de John Wilkins*” alguns trabalhos que, do ponto de vista moderno, são ontologias. O mais interessante deles é o sistema proposto por John Wilkins, que dá nome ao ensaio.

Neste sistema, criado em 1664, se dividem todos os objetos em 40 categorias ou gêneros, cada um deles dividido em subcategorias chamadas *diferenças*, estas por sua vez divididas em *espécies*. Nesta taxonomia, cada gênero é representado por um monossílabo de duas letras; cada *diferença* por uma consoante e cada *espécie* por uma vogal. Deste modo seria possível criar um idioma universal para expressar qualquer conceito. É interessante contrastar essa idéia com um

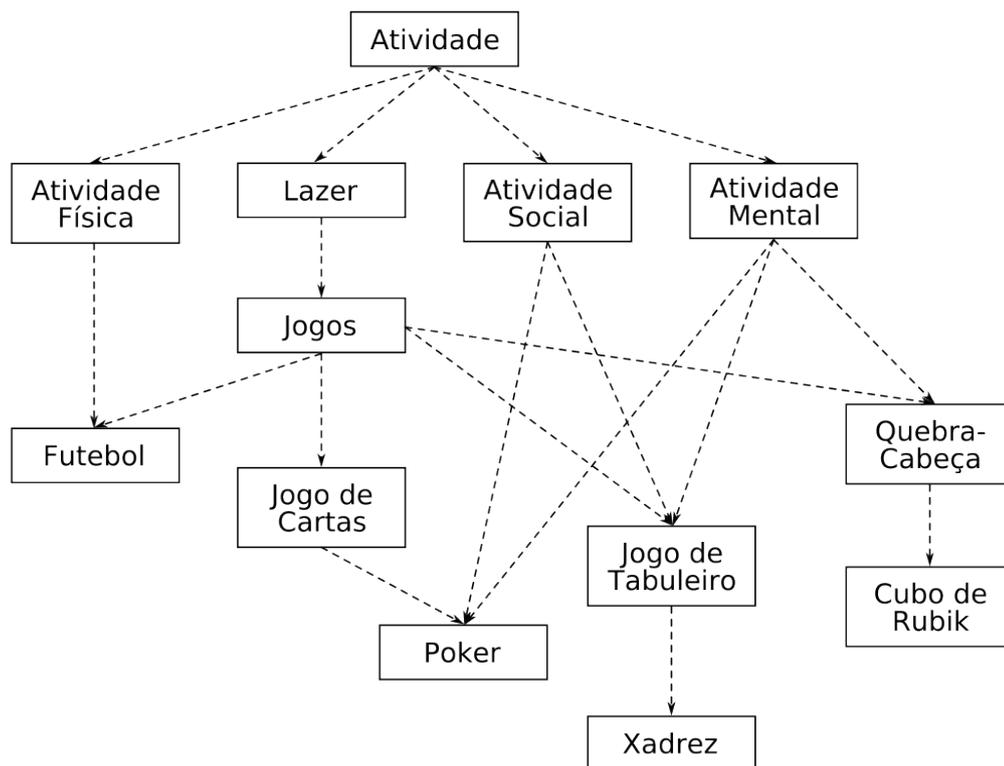


Figura 3.1: Exemplo de uma ontologia simples, que contém apenas relações do tipo “é-um”.

dos usos atuais das ontologias em IA, que é a definição de bases de conhecimento consensuais para cada área do conhecimento (STUDER et al., 1998).

A Figura 3.2 mostra uma ontologia derivada da anterior, porém mostrando mais relações entre os conceitos do que apenas a relação “é-um”. A semelhança entre o nome das relações utilizadas e as relações padrão utilizadas em modelagem orientada a objetos (LARMAN, 1998) não é acidental: um dos métodos de se representar computacionalmente uma ontologia é utilizando uma estrutura de classes e objetos. Outros métodos que podem ser utilizados são redes semânticas e, mais recentemente, *lógicas de descrição* (NARDI; BRACHMANN, 2003).

Como foi dito anteriormente, um dos modos de se representar ontologias é utilizar redes semânticas. Estas redes são compostas por nós e arcos, onde os nós normalmente representam *indivíduos* ou objetos e os arcos representam relações entre esses objetos. E de fato, as Figuras 3.1 e 3.2 são redes semânticas.

Em geral, uma rede semântica representa um conjunto de proposições sobre os objetos (GIARRATANO; RILEY, 1998, cap. 2) equivalentes às definidas em lógica de primeira ordem (BITTENCOURT, 2001, cap. 3). Esta relação com a lógica de primeira ordem se mantém desde os primeiros estudos sobre o assunto, realizados por Charles Peirce na primeira década do século XX, quando este estudava as relações entre seus *grafos existenciais* e a lógica proposicional

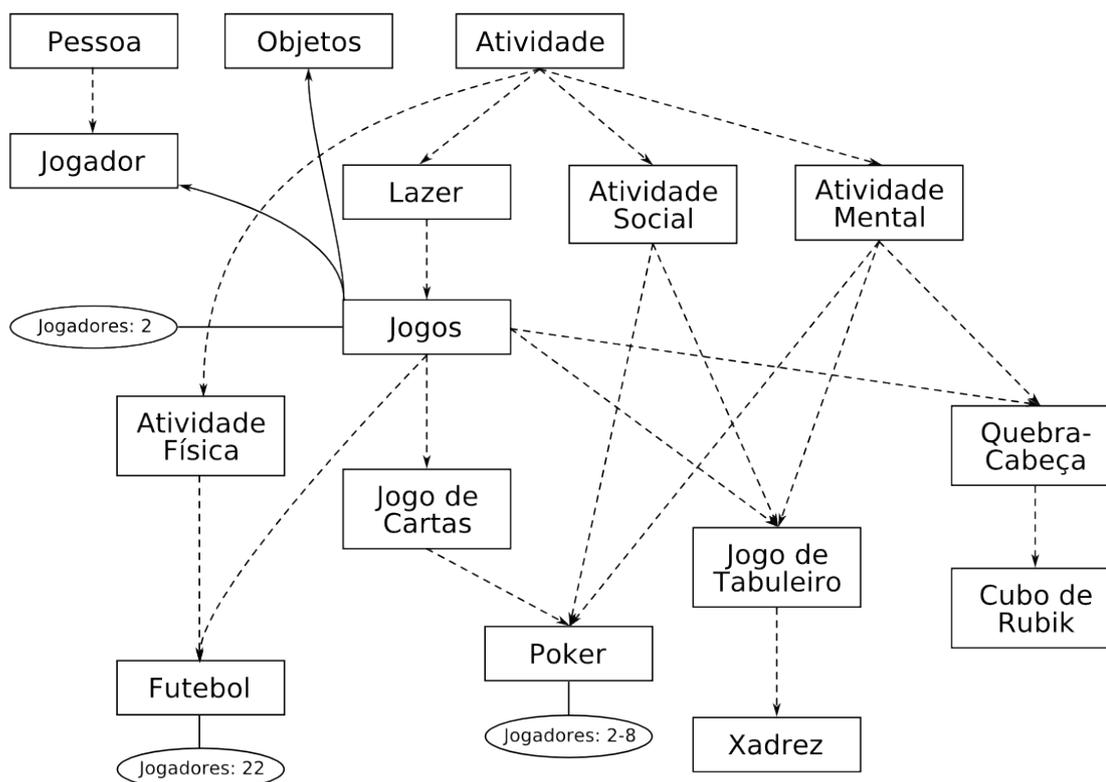


Figura 3.2: Exemplo de uma ontologia mais complexa, contendo, além de relações do tipo “é-um”, relações de “composto-por” (representadas por linhas contínuas) e alguns atributos para os objetos (representados dentro de elipses e conectados aos objetos por uma linha contínua).

(LEHMANN, 1992; RUSSELL; NORVIG, 2004), que por sua vez formaram a base da semiótica (PIERCE, 1999).

A Figura 3.3 mostra um exemplo de rede semântica, retirado de (LEHMANN, 1992), descrevendo a frase “Toby, o tigre faminto, segue sua mãe”. É interessante perceber que, assim como foi comentado anteriormente sobre ontologias, essa rede permaneceria a mesma para diferentes frases com o mesmo significado, como “Toby, filho da tigresa que ele está seguindo, está faminto”, “Aquele que deu à luz o faminto tigre Toby é por ele seguida” e mesmo “*Toby the hungry tiger follows his mother*”. Devido a esta invariância, os primeiros usos de redes semânticas em IA foram exatamente em processamento de linguagem natural e em tradução automática (LEHMANN, 1992; HOFSTADTER, 1979).

Para que uma rede semântica possa ter alguma utilidade, é preciso que ela esteja associada a um *interpretador*. Um interpretador é um mecanismo capaz de manipular as representações, e deduzir novas proposições a partir das que já estão representadas na rede (BITTENCOURT, 2001, cap. 2). Este processo corresponde a encontrar as conseqüências *implícitas* do conhecimento *explicitamente* representado na rede e é uma das habilidades fundamentais de um sistema dito inteligente (NARDI; BRACHMANN, 2003).

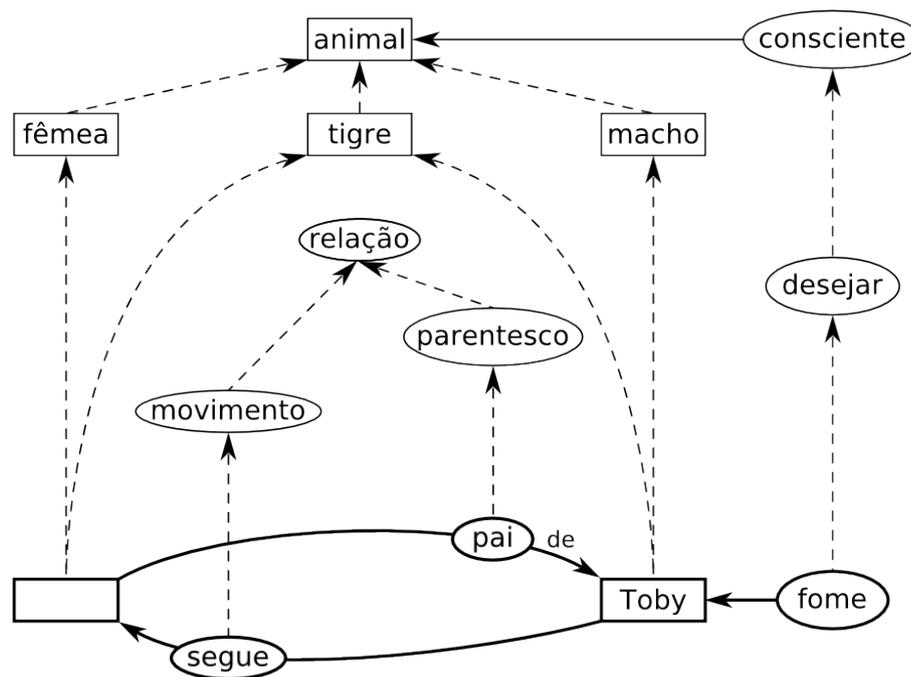


Figura 3.3: Exemplo de rede semântica representando a frase “Toby, o tigre faminto, segue sua mãe”. Nesta rede, as linhas finas tracejadas representam relações do tipo “é-um” e as linhas não tracejadas representam atributos. Os retângulos representam conceitos concretos e as elipses representam conceitos abstratos. Os objetos ou relações com linhas grossas representam asserções sobre indivíduos específicos e podem ser consideradas como conhecimento sobre o problema sendo tratado no momento. As linhas finas representam conhecimento prévio, independente de qualquer indivíduo específico (LEHMANN, 1992, pág 2).

Assim, dependendo do mecanismo utilizado para interpretar a rede semântica, ela pode ser mais ou menos expressiva. Em (LEHMANN, 1992) são dados vários exemplos de formalismos de redes semânticas. Em geral, estes mecanismos são definidos limitando-se o tipo e o número de relações entre os objetos possíveis de serem representadas. O exemplo mais simples é uma rede semântica que permite apenas a representações da relação “é-um”. Estas redes, apesar de simples, permitem representar perfeitamente taxonomias como as mostradas na Figura 3.1. Além disso, é possível fazer inferências rapidamente sobre essas redes (isto é, com baixa complexidade algorítmica). No caso dessa rede, o único tipo de inferência que ela é capaz de fazer é dizer se algum objeto é um tipo de outro objeto.

Caso se acrescente ao tipo de relações possíveis as relações “instância-de” e “implementa-interface”, obtém-se uma rede semântica capaz de representar a estrutura de classes e objetos presentes na maior parte das linguagens orientadas a objeto. Inclusive, durante o processo de compilação de um programa em Java ou C++, o procedimento de verificação de tipos realizado pelo compilador nada mais é do que uma série de consultas feitas à rede semântica que descreve todos os tipos definidos pelo programa.

Esta semelhança entre as linguagens orientadas a objetos e redes semânticas capazes de representar relações do tipo “é-um” não é meramente acidental. Durante o desenvolvimento do paradigma de orientação a objetos, além da influência da pesquisa em simulação de sistemas discretos houve influência dos resultados de pesquisas em inteligência artificial (GRAY, 1994), sobretudo através de uma outra técnica de representação, chamada *quadros*, que apareceu pela primeira vez no início da década de 70, inicialmente como um modelo cognitivo do processo de compreensão de uma cena ou uma história e como um modelo do funcionamento da memória (MINSKY, 1975).

Um quadro (do inglês *frame*) é uma estrutura de dados que representa um determinado indivíduo ou situação, de modo semelhante aos nós das redes semânticas mostradas anteriormente. Associado ao quadro estão vários tipos de informações: sobre como utilizar o quadro, sobre as relações entre esse quadro e outros quadros e sobre atributos específicos a esse objeto ou situação. Na linguagem utilizada quando se fala de quadros, estas informações são chamadas de *terminais* ou atributos. Cada terminal pode ter um nome e ou indicar uma conexão com outro quadro ou estar associado a um determinado valor.

Com isso é possível conectar um conjunto de quadros entre si, com os atributos de um quadro referenciando alguns outros. Um conjunto de quadros interconectados desse modo forma um *sistema de quadros*. Estes quadros formam uma rede equivalente a uma rede semântica. De fato, atualmente se considera quadros um dos melhores modos para implementar computacionalmente uma rede semântica (LEHMANN, 1992). A Figura 3.4 dá um exemplo de um sistema de quadros, junto com o exemplo de um quadro.

MINHA_CASA	
Atributo	Valor
Instância-de	CASA
Cor	BRANCO
Local	LUGAR_MINHA_CASA
Cômodos	(MEU_QUARTO,BANHEIRO,...)
Telhado	(TELHADO_DA_GARAGEM,...)
Fachada	FACHADA_DE_CASA

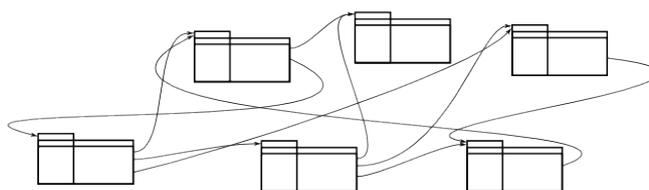


Figura 3.4: Exemplo de um quadro representando a idéia de “MINHA_CASA”. Os nomes em letras maiúsculas representam ponteiros a outros quadros. Note que alguns atributos podem receber uma lista de valores. Na direita, têm-se o exemplo de um sistema de quadros, mostrando a relação entre eles.

Associados a idéia de quadros estão três conceitos muito úteis para a representação de conhecimentos: o *raciocínio padrão*, o uso de *facetras* nos atributos e os *sacis* (BITTENCOURT, 2001).

O raciocínio padrão é a habilidade de um quadro recém-criado herdar valores padrão para os atributos que não puderam ser preenchidos explicitamente pelo processo de criação. Essa

idéia é mais facilmente compreendida com um exemplo prático. Antes de mais nada suponha que quadros são um modelo cognitivo plausível para o funcionamento de memória. Depois peça a uma pessoa que imagine uma casa. Apesar de não haver sido fornecido nenhum detalhe sobre a casa, essa pessoa provavelmente formará em sua mente uma imagem específica de uma casa, talvez a de uma casa pedagógica com quatro paredes do mesmo tamanho e um telhado de duas águas, ou a casa onde ela mora ou morou na infância. Pelo modelo de quadros, o que se pediu foi que a pessoa instanciasse um quadro de “casa” em sua mente. Como não foi fornecida uma descrição precisa da casa, os atributos descrevendo as características da casa receberam valores padrão (RUSSELL; NORVIG, 2004). Uma aplicação desse tipo de raciocínio pode ser vista na rede semântica da Figura 3.2. Nela, o número padrão de jogadores de um objeto pertencente a classe “Jogo” é dois. Assim, é possível determinar automaticamente que o número de jogadores em um jogo de xadrez também é dois, pois esse valor padrão será herdado da classe “Jogo”.

Facetas são restrições que podem ser aplicadas a valores de objetos. Por exemplo, na Figura 3.4, o atributo “Local” pode ter uma restrição que indica que esse atributo só pode receber como valor referências a quadros da classe “LUGAR” e que o número máximo de valores (cardinalidade) que podem estar nesse atributo é 1. Outros tipos de restrições podem ser colocadas, dependendo da implementação do sistema.

Finalmente, a terceira característica importante são os *sacis* (*daemons* em inglês). Sacis são funções procedurais que podem estar associadas a atributos e que são executadas em determinadas ocasiões. Ocasões comuns são: sempre que o atributo for consultado, sempre que o atributo for modificado, ou sempre que o atributo for apagado (BITTENCOURT, 2001). Os sacis permitem implementar facilmente atributos calculados, como por exemplo, a área de um telhado a partir da descrição de sua geometria. Eles também permitem embutir na representação verificações automáticas de consistência, como no caso da modificação do valor de um atributo.

3.2.2 Sistemas de produção

Sistemas de produção são um dos métodos de inferência mais bem sucedidos na área de sistemas especialistas. Estes sistemas são baseados na idéia de Emil Post, que propôs em 1936 um modelo genérico de computação baseado na aplicação sucessiva de um conjunto de regras, chamadas *regras de produção*, sobre cadeias de caracteres (BITTENCOURT, 2001). Neste sistema, cada regra é da forma $R_i : L_E \rightarrow L_D$, onde tanto L_E quanto L_D indicam cadeias de caracteres. Ao se aplicar uma regra dessa forma sobre uma cadeia de caracteres qualquer, todas as ocorrências de L_E serão substituídas por L_D .

Assim, um sistema de Post é composto por um conjunto ordenado de regras, chamado de

base de regras, e por uma seqüência de caracteres, chamada de memória de trabalho. Além disso, deve existir um componente ativo, chamado *interpretador*, que, para cada regra, caso a seqüência em L_E esteja contida na memória de trabalho, substitui na memória de trabalho a primeira ocorrência de L_E por L_D e volta a verificar se a regra pode ser aplicada. Caso ela não possa, o interpretador passa para a próxima regra e assim sucessivamente até o fim do conjunto de regras. Neste ponto, o interpretador retorna à primeira regra no conjunto de regras e continua o processo até que não seja mais possível associar o lado esquerdo de nenhuma regra à cadeia de caracteres na memória de trabalho.

Para exemplificar, seja o seguinte grupo de regras:

$$R_0 : 101 \rightarrow 111$$

$$R_1 : 111 \rightarrow 101$$

Ao se aplicar este conjunto a esta cadeia inicial de caracteres:

0101011

obtem-se o seguinte resultado:

$$0 \mathbf{101} 011 \rightarrow R_0 \rightarrow 0 \mathbf{111} 011$$

$$011 \mathbf{101} 1 \rightarrow R_0 \rightarrow 011 \mathbf{111} 1$$

$$0 \mathbf{111} 111 \rightarrow R_1 \rightarrow 0 \mathbf{101} 111$$

$$010 \mathbf{111} 1 \rightarrow R_1 \rightarrow 010 \mathbf{101} 1$$

...

Neste exemplo não é difícil perceber que o processo jamais chega ao fim, uma vez que a última cadeia mostrada no exemplo é idêntica a cadeia inicial. Um exemplo mais elaborado das possibilidades dessas transformações pode ser encontrado em (BITTENCOURT, 2001), na pág. 252.

O sistema mostrado acima, apesar de simples, é a base do funcionamento do mecanismo de inferência de grande parte dos sistemas especialistas. Nestes sistemas, ao invés de se usar letras, na memória de trabalho estão armazenadas estruturas de dados mais complexas, os *fatos*. As regras, ao invés de substituírem cadeias de letras por cadeias de letras, verificam a existência de certos fatos e os modificam ou criam novos fatos a partir deles. Como nesses sistemas a regra é o principal componente, eles também são chamados de sistemas baseados em regras. Um sistema baseado em regras é dividido nos componentes mostrados na Figura 3.5.

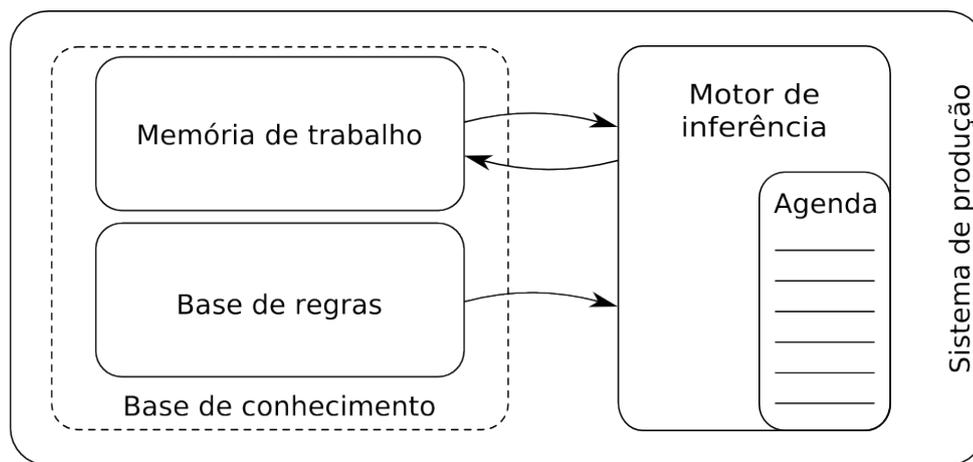


Figura 3.5: Arquitetura de um sistema de produção.

A base de conhecimento pode ser dividida em duas partes: a base de regras e a memória de trabalho. Na memória de trabalho normalmente estão fatos que estão sendo utilizados pelo sistema especialista nas inferências que ele está realizando naquele momento. Para tornar a discussão menos abstrata, aqui estão dois exemplos possíveis de fatos na memória de trabalho de um sistema especialista:

(estacao-entrega Joinville pressao baixa)

(estacao-entrega Joinville vazao baixa)

Estes fatos provavelmente foram resultado de análises anteriores realizadas pelo sistema que determinaram que a pressão e a vazão na estação de entrega de Joinville estavam baixas. As regras na base de regras contêm estruturas que podem ser associadas a esses fatos e executar ações caso ocorra uma associação. Essas ações podem ser tanto a produção de novos fatos na memória de trabalho quanto a execução de operações procedurais. Por exemplo, pode existir na base de regras uma regra com a seguinte forma:

```
(estacao-entrega ?estacao pressao baixa)
^ (estacao-entrega ?estacao vazao baixa)
→
(assert (disparar-alarme estacao-entrega ?estacao))
```

Nesta regra, o símbolo \wedge significa um “e” lógico, o termo “?estacao” é uma variável que pode assumir qualquer valor e a seta \rightarrow separa o lado esquerdo do lado direito da regra. O comando “assert” indica que se deseja criar um novo fato na memória de trabalho, com o conteúdo “disparar-alarme estacao-entrega ?estacao”. Esta regra pode ser interpretada textualmente

como: Caso haja na memória de trabalho um fato indicando que uma determinada estação de entrega está com a pressão baixa, e caso exista um outro fato na memória de trabalho indicando que a mesma estação está com a vazão baixa, então colocar na memória de trabalho um fato indicando que um alarme deve ser disparado para esta estação de entrega.

O motor de inferência é o mecanismo que faz a comparação entre os fatos e as regras e define quais regras apresentam o seu lado esquerdo satisfeito. Quando uma regra tem todas as suas pré-condições satisfeitas, diz-se que essa regra está *ativa*. É importante notar que uma regra ativa ainda não foi executada. Em um sistema especialista grande é extremamente comum que várias regras estejam satisfeitas (e portanto ativas) ao mesmo tempo. No entanto, o sistema é capaz executar apenas uma regra de cada vez. Assim, o motor de inferência coleta todas as regras ativas em uma lista, chamada *agenda*, e, utilizando algum critério (que normalmente pode ser alterado pelo programador do sistema), escolhe uma das regras para ser executada. Nesse ponto o motor de inferência executa a regra (também se diz que a regra foi *disparada*). Após o término da execução da regra o controle retorna ao motor de inferência, que atualiza a agenda em função das modificações que provavelmente ocorreram na memória de trabalho e seleciona a próxima regra a ser disparada. Esse processo continua até que alguma regra execute um comando para sair do sistema ou que não existam mais regras na agenda.

Um detalhe que deve ser levado em consideração, e que pode passar despercebido ao ler a descrição acima, é que, como o sistema normalmente não retira os fatos que causaram o disparo da regra, ela continuaria ativa, estando apta a disparar no próximo ciclo do motor de inferência. Normalmente este não é o comportamento esperado do sistema e para evitar esse problema muitos sistemas de produção implementam uma propriedade chamada *refração* (GIARRATANO; RILEY, 1998). Esta propriedade faz com que uma regra só seja disparada uma vez para um determinado conjunto de fatos que satisfaça suas pré-condições, evitando assim que o sistema fique executando a mesma regra indefinidamente.

Um dos gargalos de eficiência de um sistema de produção é o algoritmo que determina se as pré-condições das regras estão satisfeitas. Em um sistema especialista grande, em um determinado momento do processo de inferência, podem existir milhares de regras (por exemplo, r regras), cada uma com várias pré-condições (em média, c condições) e centenas ou milhares de fatos na memória de trabalho (essa quantidade será chamada de f). Em uma situação desse tipo, caso não se explore nenhuma possibilidade de otimização, será necessário verificar, a cada regra disparada, se cada pré-condição de cada regra está ou não satisfeita. Esse processo envolve rcf verificações. Como r e f são números grandes, normalmente na ordem de milhares, a cada disparo de regra seria necessário fazer um número de verificações da ordem de milhões.

Para evitar esse problema, pode-se lançar mão de certas otimizações. Um algoritmo bastante eficiente e muito utilizado na implementação de sistemas de produção é o algoritmo *Rete* (que é o latim para rede), proposto por Forgy (1982). Sem entrar em detalhes do funcionamento desse algoritmo, vale a pena mostrar os dois processos principais que ele utiliza para otimizar o processo de verificação das pré-condições, até porque isso permite entender melhor o funcionamento de sistemas especialistas.

A primeira otimização que se faz é tirar vantagem de que, apesar do número de fatos na memória de trabalho ser normalmente grande, o disparo de uma regra costuma modificar pouco a memória de trabalho, normalmente acrescentando dois ou três fatos novos e talvez retirando um ou dois fatos já presentes (pode-se considerar que a modificação de um fato nada mais é que a deleção do fato antigo seguida da introdução do novo fato na memória de trabalho). Sabendo disso, pode-se, ao invés de a cada disparo de regra verificar todos os fatos da memória de trabalho, verificar apenas os fatos que foram alterados. Assim, com essa otimização, ao invés de se comparar todas as rc pré-condições com todos os f fatos, basta realizar essa comparação com apenas alguns poucos fatos. Levando-se em consideração os valores mencionados anteriormente, isso já representa uma diminuição de duas ou três ordens de grandeza no número de operações que devem ser realizadas após cada disparo de regra.

Uma segunda otimização ainda pode ser feita caso se leve em consideração que muitas regras têm pelo menos algumas pré-condições idênticas. Essa característica pode ser explorada encontrando-se estas pré-condições idênticas e, ao invés de testá-las várias vezes, testá-las apenas uma vez e propagar o resultado para todas as regras que a utilizam. Um modo de se implementar isso em um programa utiliza uma estrutura de dados que, caso fosse representada graficamente, seria semelhante a uma rede, daí o nome do algoritmo. Esta otimização, dependendo da estrutura do sistema, pode diminuir ainda mais o número de verificações necessárias, fazendo com que a verificação das pré-condições em um sistema de produção seja um processo bastante rápido.

3.2.3 Lógica difusa

No processo de modelagem do conhecimento normalmente se faz necessário modelar o modo pelo qual os especialistas tratam incertezas. Ao longo do tempo a comunidade da IA desenvolveu diferentes métodos para tratamento de incertezas. Entre eles estão a teoria de Dempster-Shafer (GIARRATANO; RILEY, 1998), a estatística, em particular as redes bayesianas (PEARL, 1986; NEAPOLITAN, 1990) e a teoria dos conjuntos difusos (ZADEH, 1965).

A pesquisa em conjuntos difusos originou a lógica difusa, que é uma extensão da lógica bivalorada capaz de tratar de diferentes “graus” de veracidade. Devido a sua flexibilidade, a lógica difusa foi escolhida para esse trabalho como melhor método de representação de incerteza. Esta seção visa descrever brevemente esta lógica e desenvolver o formalismo que foi utilizado neste protótipo. A maior parte do conteúdo desta seção foi retirado de (KAUFMANN, 1975) e de (DUBOIS; PRADE, 1980).

Seja um conjunto clássico qualquer X de objetos, enumerável ou não¹. Um dos modos de se representar um subconjunto $A \in X$ é através de sua função de pertinência $\mu_A : X \mapsto \{0; 1\}$. Esta função costuma ser representada por:

$$\mu_A(x) = \begin{cases} 1 & \text{sse } x \in A \\ 0 & \text{sse } x \notin A \end{cases} \quad (3.1)$$

Esta função retorna para cada elemento do *universo de discurso* X um valor indicando se o elemento x pertence (1) ou não pertence (0) ao conjunto A .

A teoria dos conjuntos difusos estende este conceito permitindo graus de pertinência diferentes de 1 ou 0, ou seja, esta teoria permite representar a *incerteza de classificação* (KLIR; YUAN, 1995) de um determinado elemento x em uma determinada categoria A . Esta extensão é feita expandindo-se o codomínio de μ_A (também chamado de conjunto de pertinência) para um conjunto totalmente ordenado, como $[0; 1]$, por exemplo. Neste trabalho será utilizado tanto \underline{A} quanto $\mu_{\underline{A}}$ para representar um conjunto difuso.

Sem dúvida, outros conjuntos além do definido pelo intervalo $[0; 1]$ podem ser utilizados como codomínio da função de pertinência. Em (KAUFMANN, 1975, cap. 5) pode ser visto um tratamento aprofundado dessas possibilidades, mas nesse trabalho será considerado sempre o intervalo $[0; 1]$ por ser o que melhor se adequa ao problema em questão.

Com essa nova definição de função de pertinência, é preciso redefinir as três operações básicas sobre conjuntos: a intersecção (\cap), a união (\cup) e o complemento ($\bar{}$). Pode-se redefinir estes conceitos de vários modos diferentes. As definições a seguir são comuns na literatura:

$$\mu_{\underline{A} \cup \underline{B}} = \max(\mu_{\underline{A}}, \mu_{\underline{B}}) \quad (3.2)$$

$$\mu_{\underline{A} \cap \underline{B}} = \min(\mu_{\underline{A}}, \mu_{\underline{B}}) \quad (3.3)$$

$$\mu_{\bar{\underline{A}}} = 1 - \mu_{\underline{A}} \quad (3.4)$$

¹Neste trabalho, por abuso de linguagem, conjuntos enumeráveis serão chamados de conjuntos discretos e conjuntos não-enumeráveis serão chamados de conjuntos contínuos.

Estas definições têm a vantagem de preservar as características das operações tradicionais quando aplicadas a conjuntos ordinários.

A partir destes conceitos é possível definir uma lógica que será chamada de *lógica difusa*. Ao contrário da lógica de primeira ordem, que considera apenas dois valores-verdade possíveis – 0 para absolutamente falso e 1 para absolutamente verdadeiro – a lógica difusa permite tratar de todas as gradações possíveis entre esses dois valores. Existem várias interpretações possíveis para essa gradação. Neste trabalho, se considera que 0 significa absolutamente falso, 1 absolutamente verdadeiro e $\frac{1}{2}$ absolutamente ambíguo.

O primeiro passo para se definir a lógica difusa é o conceito de variável difusa. Uma variável difusa será representada como:

$$\underline{a} = \mu_{\underline{A}}(x) \in M = [0; 1], \quad x \in X \quad (3.5)$$

Nesta definição, X é o universo de discurso ou conjunto de referência da lógica difusa. Todas as variáveis sempre se referem a elementos desse conjunto. O conjunto \underline{A} é um subconjunto difuso de X . Assim podemos definir as seguintes operações sobre variáveis difusas:

$$\underline{a} \vee \underline{b} = \max(\underline{a}, \underline{b}) \quad (3.6)$$

$$\underline{a} \wedge \underline{b} = \min(\underline{a}, \underline{b}) \quad (3.7)$$

$$\neg \underline{a} = 1 - \underline{a} \quad (3.8)$$

Estas operações representam, respectivamente, o “ou” lógico ou disjunção, o “e” lógico ou conjunção e a negação. É interessante notar a semelhança entre estas definições e as definições 3.2, 3.3 e 3.4. E de fato, assim como na lógica clássica, para cada definição para as operações entre conjuntos difusos existe uma definição de lógica difusa correspondente. Em (KANDEL, 1986) pode ser visto como diferentes definições para 3.2, 3.3 a 3.4 dão origem a lógicas difusas com propriedades diferentes.

Um outro modo para se representar as operações definidas em 3.6, 3.7 e 3.8 é utilizando uma notação funcional. Nesta notação a operação $\underline{a} \vee \underline{b}$ passa a ser $\vee(\underline{a}, \underline{b})$. Esta notação será mais conveniente quando forem definidas operações que podem ser aplicadas a um número arbitrário de variáveis e portanto será a preferida neste trabalho.

O próximo conceito necessário é o de *função de variáveis difusas*. Uma função \underline{h} construída apenas com variáveis difusas $\underline{a}, \underline{b}, \dots$ é uma função difusa caso $0 \leq \underline{h} \leq 1$. É fácil ver que uma função formada apenas pela composição das funções $\vee(\cdot)$, $\wedge(\cdot)$ e $\neg(\cdot)$ é uma função difusa.

Para que se tenha uma lógica no sentido próprio da palavra, é preciso ainda apresentar alguma definição para os quantificadores universal e existencial. Uma definição possível para esses quantificadores é:

$$\forall \underline{x} \underline{h}(\underline{x}) = \inf_{\underline{x}} (\underline{h}(\underline{x})) \quad (3.9)$$

$$\exists \underline{x} \underline{h}(\underline{x}) = \sup_{\underline{x}} (\underline{h}(\underline{x})) \quad (3.10)$$

onde a expressão $\forall \underline{x}$ significa “para qualquer \underline{x} ” e $\exists \underline{x}$ significa “existe pelo menos um \underline{x} ”. O operador $\sup_{\underline{x}}$ ($\inf_{\underline{x}}$) significa o maior (menor) valor da função \underline{h} , substituindo-se todos os valores possíveis da variável difusa \underline{x} dentro do universo de discurso.

Esse conjunto de definições é uma lógica em que a proposição atômica, isto é, a proposição mais simples possível, é a pertinência de um determinado elemento a um determinado conjunto difuso. Esta proposição é representada por uma variável difusa. E, de fato, adotando essa definição qualquer função de variáveis difusas formada apenas pela composição das funções $\vee(\cdot)$, $\wedge(\cdot)$ e $\neg(\cdot)$ é também uma proposição.

Pode-se dizer que essa lógica se reduz a uma lógica tri-valorada no sentido que, caso duas proposições difusas de n variáveis tenham os mesmos valores para $\{0; \frac{1}{2}; 1\}^n$, então as duas proposições terão os mesmos valores para $[0; 1]^n$ (DUBOIS; PRADE, 1980, pág. 162).

Apesar de ser possível fazer o desenvolvimento de um sistema de programação lógica utilizando algumas variações da lógica mostrada acima (EBRAHIM, 2001; VOJTÁŠ, 2001), neste trabalho o maior interesse está em utilizar proposições difusas dentro de um sistema baseado em regras. Para tanto, será utilizado um dos tipos de regra mostrados em (CODÓN et al., 1999). Na notação usada, uma regra difusa tem a seguinte forma:

$$R_k : \text{Se } \underline{h}_k(\underline{a}_1, \dots, \underline{a}_n) \text{ então } \underline{b}_1, \dots, \underline{b}_m \text{ com grau de certeza } r_k \quad (3.11)$$

Esta regra associa o valor-verdade de uma proposição \underline{h}_k no antecedente com o valor-verdade de uma série de variáveis difusas no consequente. A regra em si tem um valor de certeza r_k , que será utilizado junto com o valor-verdade de \underline{h}_k para determinar o valor-verdade das variáveis $\underline{b}_1, \dots, \underline{b}_m$.

O processamento de uma regra desse tipo segue o seguinte algoritmo:

1. Determinar o valor-verdade da função \underline{h}_k a partir do valor-verdade de $\underline{a}_1, \dots, \underline{a}_n$. Esta etapa é chamada de associação;

2. Determinar o grau de ativação da regra a partir do valor-verdade de \underline{h}_k e do valor de r_k . A função que calcula esse valor é chamada de *função de ativação*. Ela será representada por \underline{a} ;
3. Opcionalmente pode-se aplicar uma função $\underline{w} : [0; 1] \mapsto [0; 1]$ ao valor de ativação. Esta função será chamada *função peso*;
4. Alterar na base de conhecimento o valor-verdade de cada b_i de acordo com o valor de ativação.

A função utilizada para realizar a ativação e a função \underline{w} devem ser escolhidas de acordo com a aplicação.

Normalmente em um sistema desse tipo existirão várias regras com o mesmo conseqüente b_i . Assim, é possível que várias regras tentem definir o valor de b_i . O modo mais genérico de resolver esse conflito é utilizando uma função que será responsável por determinar o valor de b_i a partir de todos esses valores. Essa função será chamada de *função de agregação* e será representada por \underline{g} . Uma característica fundamental para a função de agregação é que ela tenha sempre o mesmo valor para qualquer permutação de seus argumentos.

Assim, selecionando uma função de ativação, uma função de agregação e uma função \underline{w} se define completamente o sistema de inferência difuso. A escolha dessas funções precisa ser feita com cuidado e depende da aplicação. Uma maior flexibilidade pode ser obtida ao se permitir que cada regra utilize funções peso e de agregação diferentes (\underline{w}_k e \underline{a}_k).

Uma característica importante deste sistema difuso é que em qualquer momento todas as regras estão ativas (SONG; LEE, 2002). Isto porque, ao contrário do mecanismo mostrado para um sistema baseado em regras convencional, este sistema não considera a existência ou não de fatos em acordo com um determinado padrão e sim o valor-verdade das proposições que estão no antecedente das regras. Desse modo todas as regras são capazes de produzir um conseqüente, mesmo que normalmente a regra tenha um valor de ativação 0.

Esta característica causará problemas quando o antecedente de uma regra (por exemplo, a regra 2) depender do conseqüente de outra (por exemplo, a regra 1). Neste caso, a regra 1 deve obrigatoriamente disparar antes da regra 2. Este problema deve ser levado em consideração explicitamente pelo sistema de inferência. Algumas soluções possíveis para esse problema estão exemplificadas em (TORRA; SODAN, 1999) e (HALL, 2001). No sistema implementado neste trabalho, esse encadeamento precisa ser definido explicitamente pelo engenheiro de conhecimento. Para isso, o sistema fornece vários níveis de inferência e um mecanismo que garante

que todas as regras que estão em um nível de inferência irão disparar antes que qualquer regra no nível seguinte.

As definições dadas anteriormente para os operadores (ou funções) $\vee(\cdot)$, $\wedge(\cdot)$ e $\neg(\cdot)$ são interessantes do ponto de vista teórico, por fazerem com que o sistema se reduza a um sistema de lógica booleana caso todas as proposições tenham valores-verdade em $\{0; 1\}$. No entanto, para muitas aplicações este tipo de comportamento pode não ser o desejado (COX, 1998). Por isso, para aumentar o poder de expressão do sistema, se permite utilizar na composição de uma proposição difusa qualquer função difusa. Para exemplificar as possibilidades dessa abordagem, considere-se o seguinte exemplo:

Suponha um sistema pneumático contendo três medidores de pressão dispostos em pontos diferentes do sistema. Suponha também que este sistema possua um mecanismo de controle específico para evitar sobrepressões. Em um determinado momento, este mecanismo pode estar em funcionamento ou em falha. Uma análise do sistema pode produzir a seguinte regra:

“Caso o mecanismo de controle esteja em falha, quanto maiores forem as pressões medidas pelos sensores, maior será o dano ao equipamento.”

Um aspecto interessante neste problema é o fato de que o dano é dependente da combinação dos valores de três sensores de pressão, condicionado ao fato do mecanismo de controle de sobrepressão estar em falha. Utilizando apenas os três operadores $\vee(\cdot)$, $\wedge(\cdot)$ e $\neg(\cdot)$ definidos anteriormente a representação dessa regra em um sistema especialista seria trabalhosa. No entanto, introduzindo o seguinte operador, que calcula a média aritmética:

$$\underline{m}(a_1, \dots, a_n) = \frac{\sum_{i=1}^n a_i}{n} \quad (3.12)$$

e as seguintes variáveis:

\underline{p}_1 : Pertinência da pressão medida no sensor 1 ao conjunto difuso “pressão alta”;

\underline{p}_2 : Pertinência da pressão medida no sensor 2 ao conjunto difuso “pressão alta”;

\underline{p}_3 : Pertinência da pressão medida no sensor 3 ao conjunto difuso “pressão alta”;

\underline{s} : Estado do controle de sobrepressão: 0 para funcionamento e 1 para falha;

\underline{d} : Pertinência do estado de dano do equipamento ao conjunto difuso “muito danificado”;

pode-se representar a regra mostrada acima como:

$$R: \text{ Se } \wedge(\underline{s}, \underline{m}(\underline{p}_1, \underline{p}_2, \underline{p}_3)) \text{ então } \underline{d} \text{ com grau de certeza } 1,0 \quad (3.13)$$

Esse exemplo mostra como o uso de vários operadores permite expressar um requerimento razoavelmente complexo em uma regra simples. Neste exemplo, sabe-se que só ocorrem danos caso o mecanismo de controle esteja em falha. A regra 3.13 terá um valor de ativação 0 sempre que o mecanismo de controle estiver em funcionamento. Caso o mecanismo de controle esteja em falha, a regra 3.13 terá um valor de ativação equivalente a média dos valores-verdade de \underline{p}_1 , \underline{p}_2 e \underline{p}_3 . Várias funções e classes de funções difusas podem ser encontradas em (KLIR; YUAN, 1995) e (COX, 1998).

Finalmente, em algumas aplicações pode ser interessante determinar qual de duas variáveis difusas é a mais ambígua. Esta relação será representada por \succeq e é uma relação de ordenamento parcial tal que:

$$\forall \underline{a}, \underline{b} \in [0; 1] \quad \underline{a} \succeq \underline{b} \text{ caso } \begin{cases} \frac{1}{2} \leq \underline{a} < \underline{b} \\ \frac{1}{2} \geq \underline{a} > \underline{b} \end{cases} \text{ ou} \quad (3.14)$$

Quando $\underline{a} \succeq \underline{b}$ diz-se que “ \underline{a} é mais ambíguo que \underline{b} ”. Caso $\underline{a} \in [\frac{1}{2}; 1]$ e $\underline{b} \in [0; \frac{1}{2}]$ ou $\underline{a} = \underline{b}$ esta comparação não está definida (DUBOIS; PRADE, 1980, pág. 152).

3.3 Ferramentas

O objetivo desta seção é descrever as ferramentas utilizadas na implementação do protótipo. Ele tratará principalmente das duas linguagens utilizadas no desenvolvimento: C++ (STROUS-TRUP, 1997) e CLIPS (CLIPS..., 2005).

CLIPS (*C Language Expert System Shell*) é uma linguagem e um ambiente de execução especialmente desenvolvidos para a implementação de sistemas especialistas (esses ambientes são normalmente chamados de *shells*). Ela teve seu desenvolvimento iniciado na NASA, em 1984 e foi criada com o objetivo específico de permitir a integração entre uma linguagem para sistemas especialistas e linguagens de programação convencionais, como o C, por exemplo. Além disso, a linguagem deveria ser capaz de ser executada em diferentes plataformas e também ser utilizada como componente de programas maiores.

Apesar do paradigma de programação inicial da linguagem ser um sistema de produção (utilizando o algoritmo *Rete*), durante o desenvolvimento da linguagem foram incluídos mais dois paradigmas: programação procedural, como a encontrada em linguagens como C e Fortran, e programação orientada a objetos. Inclusive, a linguagem de programação orientada a objetos do CLIPS implementa vários dos recursos descritos na seção sobre quadros, como raciocínio padrão, algumas facetas e, ainda que com algumas particularidades, também implementa sacis.

Para que a linguagem pudesse ser portada para várias plataformas diferentes, e por questões de eficiência, ela foi implementada em C. Posteriormente, foi feito um esforço para tornar o código diretamente compatível com C++, o que se conseguiu quase totalmente.

Como um ambiente de desenvolvimento, o CLIPS apresenta as seguintes vantagens:

- Código de domínio público, tornando o desenvolvimento barato e permitindo a criação de extensões personalizadas da linguagem;
- Sistema de produção rápido e confiável. Devido ao longo tempo de desenvolvimento e à grande comunidade de usuários praticamente não existem erros no algoritmo de inferência;
- Linguagem de orientação a objetos bem integrada com o mecanismo de inferência e com muitos recursos;
- Grande facilidade para criar extensões da linguagem. A interface de extensão é simples, bem documentada e oferece muitas possibilidades;
- Linguagem está bem documentada e tem uma comunidade de usuários ativa.

No entanto, a linguagem apresenta as seguintes desvantagens:

- Biblioteca de funções e estruturas de dados relativamente pobre, com dificuldade para criar algumas estruturas de dados, como listas aninhadas e mapas associativos;
- Falta de um mecanismo para separar estruturas de dados que devem ser consideradas como representação de conhecimento e resultado de inferências de estruturas auxiliares. Do modo como a linguagem foi projetada, todas as estruturas de dados são consideradas como representação de conhecimento;
- Ausência de algumas estruturas utilizadas em programação funcional, como as encontradas nos vários dialetos de Lisp e em Haskell. Entre essas estruturas estão as funções anônimas (funções λ) e a utilização de funções como argumentos para outras funções, como membros de listas ou mapas associativos etc.
- Alguns problemas, absolutamente contornáveis, de compatibilidade direta com o C++;

O CLIPS é uma linguagem madura e bem adaptada para o desenvolvimento de sistemas especialistas, mas é uma linguagem difícil de se trabalhar quando se deseja realizar análises numéricas e implementar algoritmos procedurais muito longos. Como para o protótipo foi

necessário realizar algumas análises numéricas, foi utilizado C++ para criar uma extensão ao CLIPS específica para realizar o acesso aos dados e essas análises. Se escolheu esta linguagem sobretudo pela facilidade de interfaceamento com o CLIPS.

O C++ apresenta a vantagem de ser orientada a objeto e incluir alguns paradigmas bastante poderosos, como programação genérica. Por outro lado, é uma linguagem não gerenciada, ou seja, o gerenciamento de memória deve ser realizado pelo próprio programador. Isto faz com que o desenvolvimento de um programa em C++ seja consideravelmente mais lento e suscetível a erros que o desenvolvimento em uma linguagem gerenciada (como o próprio CLIPS).

3.4 Conclusões do capítulo

Neste capítulo foram mostradas algumas técnicas utilizadas em inteligência artificial. Foram abordadas as técnicas utilizadas no desenvolvimento e no gerenciamento de projetos de sistemas especialistas, técnicas para representação de conhecimento, para inferência e para tratamento de incerteza. Deve estar claro, pelo desenvolvimento do capítulo, que em qualquer projeto de sistemas especialistas é preciso fazer a integração de diferentes técnicas.

No próximo capítulo está descrito como as técnicas apresentadas aqui foram utilizadas no desenvolvimento do protótipo. É possível ver como estas diferentes técnicas foram integradas entre si, e alguns problemas relacionados ao desenvolvimento do sistema.

4 Descrição do protótipo

Neste capítulo está descrita a implementação do protótipo, procurando mostrar todos os aspectos do processo de desenvolvimento, desde a definição dos requisitos até os testes realizados sobre o sistema. Considera-se que esta informação é de valor por apresentar um caso completo de desenvolvimento de um sistema especialista. Como se verá, ocorreram problemas durante o desenvolvimento do sistema. Estes problemas serão sumarizados e rapidamente comentados no final do capítulo.

Assim, esse capítulo tem um duplo objetivo: o primeiro é apresentar a implementação de um sistema especialista para auxiliar a detecção de vazamentos. Os detalhes dessa implementação se encontram principalmente na seção 4.3. O segundo objetivo é fazer um estudo de caso do processo de desenvolvimento de um sistema especialista, mostrando todas as etapas e problemas que foram encontrados, e comparando estes problemas com os que já haviam sido indicados na literatura sobre o assunto.

Para contextualizar o desenvolvimento deste protótipo, é preciso dizer que ele está inserido no projeto SEGRED, que foi descrito na seção 1.3. Neste projeto, o módulo de detecção de vazamentos teve seu desenvolvimento iniciado no mês de outubro de 2004, e um prazo de finalização até novembro de 2005. Devido a atrasos por parte da empresa-alvo, este prazo foi estendido até abril de 2006.

No desenvolvimento deste protótipo, assim como nos outros módulos do SEGRED, foi adotado um modelo de desenvolvimento em espiral. Neste modelo, o desenvolvimento do sistema ocorre em uma série de ciclos rápidos de aquisição de conhecimento, representação de conhecimento, implementação e testes (GIARRATANO; RILEY, 1998). Além disso, antes do início deste ciclo existe uma etapa de definição de requisitos.

Sendo assim, este capítulo se inicia com a especificação dos requisitos. Em seguida, na seção 4.2, é mostrado o processo de aquisição de conhecimento, com descrições das entrevistas de aquisição de conhecimento e do conhecimento adquirido. Após essa descrição, é mostrada a arquitetura do protótipo que foi implementado. Depois, na seção 4.4 são mostrados os testes

realizados sobre o sistema. Na seção 4.5 é apresentada uma proposta de método para gerar explicações no sistema especialista difuso. Finalmente, na seção 4.6 são feitos comentários sobre o desenvolvimento do protótipo.

Como neste capítulo se entrará em detalhes sobre o funcionamento do gasoduto Bolívia-Brasil e sobre o funcionamento do sistema de detecção de vazamentos da empresa-alvo (o PMS), vários dos conceitos descritos nas seções 1.2 e 2.6 serão usados.

4.1 Requisitos do sistema

Como nota inicial, é preciso dizer que houve certa indefinição com relação aos requisitos até que se chegasse a especificação final dos requisitos para o módulo de detecção de vazamentos. Estes só foram completamente definidos em novembro de 2005, em aquisição de conhecimento realizada com o principal especialista na sede da empresa-alvo. Nesta entrevista, ficou definido que o objetivo do sistema era:

“Educar os controladores a não confiarem apenas no *leak detection graph* e observarem mais a tela do sistema de detecção de vazamentos do PMS. Desse modo, o módulo de detecção de vazamentos deve fazer inferências a partir dos filtros do detector de vazamentos e informar ao controlador o resultado.”

Essa informação, junto com o resultado da aquisição de conhecimento com os controladores do gasoduto e com o próprio especialista, permitiu concluir que o interesse estava em se detectar as seguintes situações:

1. Erro de instrumentação: algum instrumento está produzindo valores incorretos;
2. Ocorrência de transientes na linha que, de algum modo, provoquem indicações no detector de vazamento;
3. Falhas de comunicação que provoquem indicações no detector de vazamentos;
4. Valores forçados no SCADA que resultem em indicações no detector de vazamentos;
5. Mudanças bruscas de comportamento de EEs;
6. Mudanças bruscas de comportamento de ECOMPs;
7. Fechamento de válvulas de bloqueio;

8. Vazamentos.

Por uma questão de completude da base de conhecimento, devem ser consideradas ainda estas duas situações:

9. Situação normal. Não existe nada de interesse do controlador ocorrendo no duto;

10. Situação desconhecida. A situação operacional não se encaixa em nenhum dos casos apresentados anteriormente.

Vale ressaltar que, em discussões anteriores com a empresa-alvo, já havia sido decidido que o programa não duplicaria a funcionalidade do PMS, sendo na verdade um pós-processador das informações produzidas por ele. Desse modo, as análises realizadas pelo sistema especialista têm como entradas o histórico dos valores dos filtros do detector de vazamentos do PMS e o histórico dos sensores presentes no SCADA.

Além disso, durante as aquisições de conhecimento, a empresa-alvo manifestou interesse em implementar heurísticas que auxiliassem na detecção do fechamento de válvulas de bloqueio de trecho. Desse modo, considera-se ainda o seguinte requisito:

11. Implementação de heurísticas para localização de válvula de bloqueio fechada.

Até o momento foram tratados de requisitos relacionados aos tipos de análise que o sistema especialista deve fazer. Além desses, existem outros requisitos para um sistema desse tipo, que são:

12. Implementação de um módulo de explicações para a base de conhecimento;

13. Capacidade de funcionar por um longo prazo ininterruptamente;

14. Capacidade de obter dados automaticamente a partir dos bancos de dados da empresa-alvo. Esta capacidade pode ser dividida em duas:

(a) Leitura automática dos dados do SCADA;

(b) Leitura automática dos dados do PMS;

15. Interface com o usuário;

16. Verificação do sistema pelos desenvolvedores. Em virtude do funcionamento do sistema, este requisito pode ser dividido em dois:

- (a) Verificação das análises realizadas apenas sobre os filtros do PMS;
- (b) Verificação das análises realizadas sobre os valores do SCADA;

17. Validação do sistema junto a especialistas da empresa-alvo.

Esta é a lista de requisitos que foi considerada no desenvolvimento do módulo. Uma das características do estabelecimento desta lista é que, como foi dito, houveram atrasos na definição dos requisitos. Por culpa destes atrasos, o desenvolvimento do sistema se iniciou sem que a lista de requisitos estivesse completamente definida. Isto provocou vários problemas, entre eles o retrabalho, pois algumas partes do sistema tiveram de ser alteradas para aderir às mudanças nos requisitos. Além disso, alguns requisitos presentes no início do desenvolvimento foram depois retirados. Isto provocou um desperdício de trabalho, pois toda a pesquisa preliminar realizada para a implementação destes requisitos não foi utilizada no desenvolvimento final.

Com relação à importância relativa entre os requisitos, não foi realizado nenhum processo formal para determiná-la, em parte devido a fluidez dos mesmos durante boa parte do desenvolvimento. No entanto, estabeleceu-se que os requisitos de 1 a 11 e 15 são mais importantes que os demais.

Na próxima seção será explicitado o resultado das aquisições de conhecimento, e o tipo de análise necessária para cada um dos requisitos. Também será possível observar a evolução do estabelecimento de alguns itens da lista apresentada acima.

4.2 Aquisição de conhecimento

Como já foi comentado na seção 3.1, a etapa de aquisição de conhecimento é fundamental para o desenvolvimento de sistemas especialistas. Ela toma, inicialmente, a forma de pesquisas bibliográficas sobre o assunto e entrevistas não estruturadas com especialistas. Conforme o entendimento dos engenheiros de conhecimento aumenta, as entrevistas se tornam progressivamente mais estruturadas, incluindo questionários e estudos de caso (SCOTT et al., 1991).

Antes do início da aquisição, é fundamental que se tenha as seguintes informações:

- O escopo do sistema deve estar bem definido;
- O especialista ou grupo de especialistas deve ter sido selecionado;

Como foi mencionado na seção anterior, o escopo do sistema só foi completamente definido na última entrevista de aquisição de conhecimento. Isso teve um impacto bastante negativo

sobre o desenvolvimento do projeto. Com relação ao número de especialistas, Dym & Levitt (1991, sec. 12.4) discutem sobre a importância de realizar a aquisição de conhecimento com vários especialistas. Alguns dos argumentos são:

- Maior facilidade em encontrar uma solução que representa um consenso na área, e não uma solução que represente fielmente as idiosincrasias de um especialista;
- Fazer com que um especialista produza casos para outro permite que pontos pouco compreendidos do problema apareçam, pois os casos tendem a ser complexos e representar o conhecimento do especialista que produziu o caso;
- O estudo do modo como vários especialistas resolvem uma tarefa pode identificar atalhos usados por cada um deles que podem tornar o método de solução do problema mais eficiente quando combinados.

Para este protótipo, foi feita a aquisição de conhecimento com mais de um especialista. No entanto, a empresa-alvo só disponibilizou estes especialistas na última entrevista de aquisição, já muito próximo do prazo para a entrega do protótipo. Durante a maior parte do desenvolvimento, houve apenas um único especialista disponível para o projeto.

A seguir será apresentado um resumo das entrevistas de aquisição de conhecimento voltadas especificamente para a funcionalidade de detecção de vazamentos. Após cada resumo serão feitos comentários sobre o conteúdo e utilidade das entrevistas.

Entrevista realizada em 2005-02-11

Foi realizada uma entrevista introdutória sobre o assunto com dois engenheiros do escritório da empresa-alvo responsável pelo trecho sul do gasoduto. Nesta entrevista foram abordados: o método de detecção de vazamentos via *software* utilizado atualmente pela empresa; eventos que podem desencadear um vazamento, como trincas no duto e intervenções de terceiros; uma análise, realizada pela empresa, do impacto de um vazamento sobre as comunidades ao longo do duto; e a frequência com que os PIGs são passados pelo gasoduto.

Comentários: Os dois engenheiros entrevistados não podem ser considerados especialistas em detecção de vazamentos, pois não estão diretamente envolvidos com as operações de controle do gasoduto. No entanto, esta entrevista foi importante para estabelecer um conhecimento preliminar sobre o assunto, necessário para orientar a pesquisa bibliográfica e as entrevistas de aquisição de conhecimento posteriores. Como costuma ser o caso que especialistas têm muito

pouco tempo disponível, por serem muito solicitados (HART, 1992), nas fases iniciais da aquisição de conhecimento pode ser melhor tratar com outras pessoas, de maior disponibilidade, até que os engenheiros de conhecimento adquiram um entendimento mínimo sobre o assunto.

Entrevista realizada em 2005-04-14

Nesta data o especialista que a empresa designou para o projeto fez uma apresentação sobre o sistema de detecção de vazamentos utilizado pela empresa, dando uma visão geral sobre seu funcionamento e explicando quais são suas entradas e saídas. Foi sugerido que um dos modos que o SEGRED poderia operar é como um filtro sobre os alarmes gerados pelo sistema de detecção de vazamentos do PMS, com o intuito de diminuir o número de alarmes falsos e ao mesmo tempo identificar corretamente as situações operacionais que produziram este alarme falso. O especialista forneceu algumas heurísticas para a detecção dessas situações. Mais heurísticas deveriam ser levantadas em reuniões posteriores.

Além disso, foi considerado importante que o SEGRED implementasse toda a série de funcionalidades necessárias para a detecção de vazamentos utilizando modelos dinâmicos. Isto inclui: modelos para simulação dinâmica, sistema classificador de vazamento e sistema especialista (como foi comentado na seção 2.3.3, pág. 18). Também foi sugerida a possibilidade do SEGRED acessar diretamente a saída do PMS, utilizando estas informações para inferência sobre detecção de vazamentos.

Foi decidido que os trabalhos seguiriam a seguinte ordem: inicialmente implementar o sistema especialista para filtrar os alarmes gerados pelo PMS, em seguida o algoritmo utilizado para detecção de vazamentos e, finalmente, trabalhar nos modelos de simulação.

Comentários: Observou-se nesta entrevista uma indefinição do escopo do sistema de detecção de vazamentos. Apesar de ter sido decidido que a implementação se iniciaria com o sistema especialista fazendo inferências sobre os dados do PMS, ainda fazia parte do escopo a implementação de um SDV baseado em modelos dinâmicos. Note que um dos pontos que depois se mostrou principal na especificação atual, o de “educar os controladores a utilizar mais as telas do PMS” ainda não se encontra presente. Outro ponto que não se encontra presente é o uso de dados do SCADA junto com as análises das respostas do PMS. Também se discutiu possibilidades que se mostraram tecnicamente irrealizáveis, como a disponibilização dos dados do PMS, que não permite que o resultado de suas análises seja exportado para outros sistemas.

Entrevista realizada em 2005-11-11

Foi feita uma entrevista com o especialista principal do projeto para elucidar conceitos sobre o sistema de detecção de vazamentos da empresa-alvo. A entrevista previa também uma avaliação do protótipo que havia sido desenvolvido, uma vez que, mesmo já estando disponíveis duas versões do protótipo, uma entregue em 19 de julho e outra em 7 de outubro, nenhuma versão havia sido avaliada. No entanto, após a entrevista, o especialista preferiu não avaliar o protótipo, pois foram descobertos na entrevista alguns conceitos que estavam implementados incorretamente. Por exemplo, considerava-se que os filtros do detector de vazamento podiam assumir valor 0, o que não é correto. Foram apresentados rapidamente alguns casos simples para resolução pelo especialista. Foi marcada uma visita à CSC para aquisição de conhecimento até o fim do mês.

Comentários: É importante notar que a maior parte dos conceitos incorretos encontrados nesta entrevista teriam sido percebidos caso pelo menos um dos protótipos houvesse sido avaliado. Isto mostra a importância de se manterem todas as fases do desenvolvimento em espiral. Além disso, percebe-se o enorme hiato entre duas entrevistas de aquisição, no caso, de abril até novembro (cerca de 6 meses). Percebe-se ainda que, mesmo neste ponto já bem avançado do projeto, próximo da data de entrega do produto, o escopo do projeto ainda não havia sido definido de forma satisfatória¹.

Entrevistas realizadas de 2005-11-23 a 2005-11-25

Nesta visita de três dias à CSC foram tratados vários assuntos relacionados ao projeto SE-GRED como um todo, mas o assunto principal foi detecção de vazamentos. De todas as entrevistas de aquisição de conhecimento, esta foi a mais produtiva. Esta foi a primeira entrevista feita com mais de um especialista. No caso, os novos especialistas envolvidos no projeto foram os controladores do gasoduto.

O conjunto de entrevistas feitas com os controladores permitiu captar heurísticas importantes para complementar as análises feitas sobre os filtros do PMS. Foram adquiridas uma série de heurísticas associadas a *trends* de sensores e as relações entre estes *trends* e os estados dos filtros do PMS. Uma análise mais profunda do que os controladores observam quando avaliam os *trends* se fez necessária, mas não foi possível de se realizar durante esta visita pois, apesar do longo tempo gasto junto dos controladores, estes tinham, concomitantemente, que realizar

¹ Apesar que, nesta altura já se sabia que a implementação de um sistema de detecção de vazamentos completo era absolutamente impraticável.

suas atividades de turno. Portanto, o tempo que estes efetivamente tinham disponível para a aquisição era pequeno.

A entrevista com o principal especialista do projeto na empresa-alvo forneceu a definição final do escopo do sistema (esta definição inclui análises dos dados do SCADA e treinamento dos controladores). Nesta mesma entrevista ele também relatou a necessidade de se avaliar o histórico dos filtros do PMS por pelo menos duas horas, para se determinar as interações entre os filtros. Isto porque os filtros de desbalanceamento de pressão respondem mais lentamente que os de desbalanceamento de vazão.

Também foram discutidos os detalhes de acesso ao SCADA. Embora isto não esteja diretamente relacionado às entrevistas de aquisição de conhecimento, também é necessário para a implementação do protótipo. Durante esta visita à CSC foram dadas as permissões gerenciais necessárias e o acesso aos dados pôde ser implementado.

Comentários: Apesar da descrição sucinta que foi feita desta visita, ela foi bastante produtiva. A Tabela 4.1 detalha os tipos de análises que serão feitas pelo sistema, a maior parte delas baseada nas informações adquiridas nesta entrevista.

Na próxima seção está descrito o conhecimento utilizado na implementação do sistema, proveniente, em sua maior parte, desta última entrevista de aquisição de conhecimento.

4.2.1 Análises que devem ser realizadas pelo sistema

Aqui serão apresentadas as análises necessárias para implementar os requisitos de 1 a 10. Cada uma destas situações possíveis foi chamada de situação operacional. Para facilitar a análise, elas foram hierarquizadas como mostra a Figura 4.1. Note-se que o caso de transientes (req. 2) foi desdobrado nos requisitos de mudanças em EEs (req. 5), ECOMPs (req. 6) e em empacotamento ocorrendo na seção. Considerou-se que qualquer outro caso, que a princípio deveria ser classificado como transiente, deve ser tratado como situação não identificada para indicar a necessidade de um refinamento da base de conhecimento. Isto para tentar evitar a classificação incorreta de uma situação importante como apenas um transiente.

A partir dos dados da entrevista de aquisição e utilizando um método exemplificado em (ALONSO et al., 2000) foi feito um mapa do conhecimento necessário para cada classificação. A Tabela 4.1 mostra este mapa.

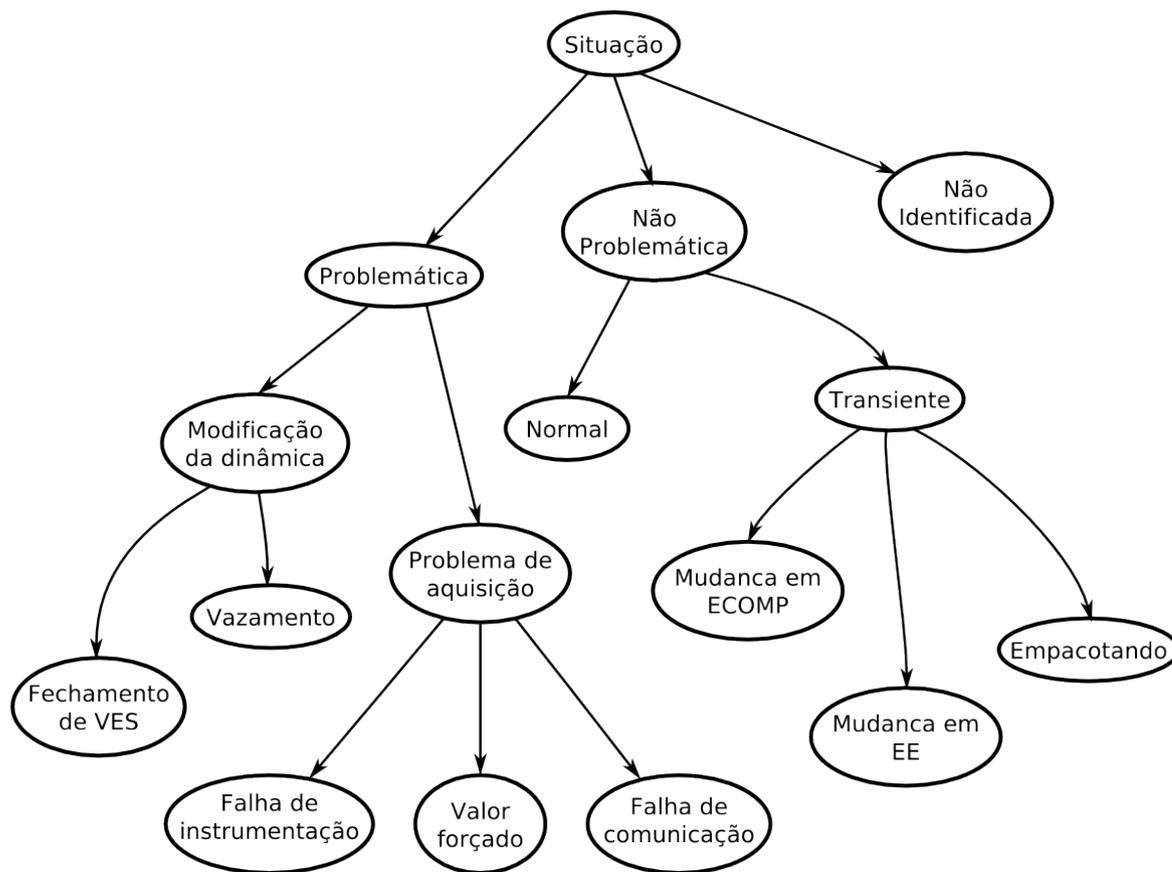


Figura 4.1: Ontologia de situações operacionais possíveis para o sistema especialista. Cada folha da árvore representa uma situação “conhecida” pelo sistema. As setas representam a relação “é-um”.

Tabela 4.1: Mapa de conhecimento para o sistema.

Situação	Análises
Situação normal	<ul style="list-style-type: none"> • Filtros normais; • <i>Trends</i> dos sensores não indicam variações de interesse; • <i>Trend</i> na estação indica mudança; • Verificar registros de calibração dos medidores de vazão da estação (calibração muito antiga pode indicar um problema no medidor, e não um aumento de consumo);
Mudança bruscas de comportamento de EE	<ul style="list-style-type: none"> • Comparar com o consumo previsto para aquela estação (caso o aumento for muito maior que o consumo previsto, pode ser um problema na estação ou no consumidor); • Verificar com o consumidor se ele realmente está solicitando esse consumo (caso contrário, pode ser uma indicação de vazamento na rede do consumidor).

Continua na próxima página

Continuação da Tabela 4.1

Situação	Análises
Mudança bruscas de comportamento de ECOMP	<ul style="list-style-type: none"> • <i>Trend</i> nos sensores da ECOMP indica mudança na vazão ou pressão; • Um valor negativo nos filtros de pressão e um positivo nos filtros de vazão do PMS associados aquela ECOMP podem indicar que a vazão da ECOMP aumentou recentemente e que está ocorrendo empacotamento no duto.
Empacotamento	<ul style="list-style-type: none"> • Filtros do PMS com valores negativos tanto a montante quanto a jusante são indicação de empacotamento ocorrendo no trecho;
Falha de comunicação com a CSC	<ul style="list-style-type: none"> • Observar alarmes do SCADA;
Valor forçado	<ul style="list-style-type: none"> • Indicação no SCADA; • Mudança recente do <i>quality bit</i> de normal para forçado ou vice-versa; • Valores inconsistentes entre os filtros do PMS;
Erro de instrumentação	<ul style="list-style-type: none"> • <i>Trend</i> do instrumento indica mudança brusca recente; • Instrumento indicando fundo de escala (superior ou inferior); • Valores inconsistentes entre os filtros do PMS; • Se for uma EE, verificar registros de calibração dos medidores de vazão da estação (sensores com calibração muito antiga tendem a dar mais desvios que sensores com calibração recente).
Fechamento de válvula de bloqueio	<ul style="list-style-type: none"> • Aumento de pressão a montante da válvula fechada; • Diminuição de pressão a jusante da válvula fechada; • Filtros do PMS negativos a montante e positivos a jusante; • A proporção entre os gradientes de pressão nas estações de entrega pode dar uma indicação da posição da válvula de bloqueio fechada.
Vazamento	<ul style="list-style-type: none"> • Pelo menos 3 filtros do PMS a montante e a jusante da seção no nível máximo;
Situação não-identificada	<ul style="list-style-type: none"> • Quando não é possível classificar em nenhuma outra;

Dois aspectos saltam aos olhos ao se observar a tabela. O primeiro é a necessidade de acessar os registros de calibração ou contatar os consumidores para a classificação de mudança

brusca de comportamento em EE e de erro de instrumentação. Isto não pode ser feito automaticamente. Portanto, para a implementação destas classificações, foram utilizadas apenas as análises que podem ser feitas automaticamente. Na mensagem produzida pelo sistema foi incluído como recomendação a busca por essas informações, como forma de confirmar o diagnóstico.

O segundo é a indefinição de alguns termos, como *trend*, por exemplo. Para se definir corretamente esse termo e capturar todos os seus desdobramentos em diferentes situações, seriam necessárias novas entrevistas de aquisição de conhecimento, com estudos de caso bem detalhados. Na implementação atual, sempre que se faz uma análise de *trend* em algum ponto do sinal, se busca determinar a derivada do sinal naquele ponto. Maiores detalhes sobre o método utilizado estão na seção 4.3.2.

4.3 Implementação da base de conhecimento

Esta seção descreve a implementação do sistema. Ela está dividida em três partes. A primeira apresenta uma visão global da arquitetura sem entrar em detalhes de implementação. A segunda parte trata da implementação das análises numéricas feitas sobre os valores dos sensores. A terceira parte trata dos detalhes da implementação do sistema de lógica difusa.

4.3.1 Arquitetura do sistema

O sistema utiliza o método de resolução de problemas *classificação heurística* (CLANCEY, 1985). Este método de solução é endossado pelo arcabouço CommonKADS (SCHREIBER et al., 1999) como o melhor para resolver problemas de classificação e diagnóstico, e funciona sempre que se tem uma hierarquia de classes pré-definidas para o problema de classificação. Como mostra claramente a figura 4.1 (pág. 68), este é o caso deste projeto. Este método se divide em três partes, relacionadas como mostra a Figura 4.2:

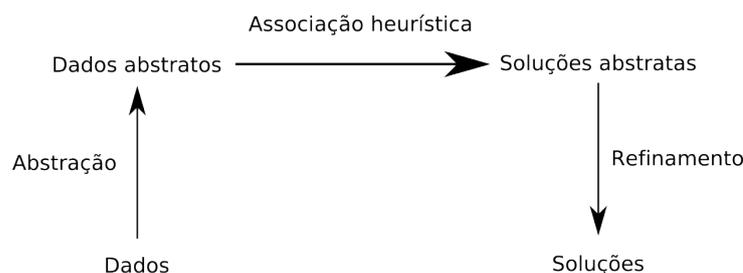


Figura 4.2: Diagrama do processo de inferência da classificação heurística (CLANCEY, 1985).

Abstração dos dados: É o processo pelo qual o sistema especialista recebe dados do mundo real e retira deles as características fundamentais, ao mesmo tempo diminuindo a quantidade de informação a ser tratada.

Associação heurística: Neste ponto, o sistema deve associar uma série de conjuntos abstratos de dados a uma série de possíveis soluções. Por exemplo, o sistema pode associar um determinado estado dos sensores a uma situação, algo como: Se filtros do detector de vazamentos estão normais então a situação é normal.

Refinamento: Nesta etapa, com uma ou algumas possibilidades de solução já previamente escolhidas, o sistema acrescenta detalhes a cada solução, por exemplo, especificando qual válvula foi fechada.

Uma vez que foi escolhido este método de solução, é preciso detalhar:

- Implementação do processo de abstração, associação e refinamento:

Decidiu-se que o melhor seria utilizar um sistema híbrido, baseado em lógica difusa e sistemas de produção. Portanto, foi desenvolvido em CLIPS um sistema para tratamento de proposições difusas que permite, com certa facilidade, a criação de regras difusas e a combinação dessas regras com regras comuns de um sistema de produção. Com relação ao refinamento, optou-se por utilizar o mesmo sistema, porém com um maior uso de regras tradicionais.

- Representação dos dados:

A maior parte dos dados provém do SCADA, e são séries temporais. Para representá-los, foi criada uma extensão ao CLIPS em C++ a fim de armazená-los e realizar alguns tratamentos numéricos, como por exemplo, o cálculo de derivadas.

- Representação do conhecimento:

Foram criadas classes específicas para armazenamento dos valores do sistema de detecção de vazamentos da empresa-alvo, armazenamento de possíveis soluções e tratamento de proposições difusas, detalhamento das funções de cada sensor do gasoduto, etc. Além disso foram implementadas regras difusas para representar o modo como as características de uma situação operacional se relacionam com as possíveis soluções.

Para tornar a descrição mais lógica, serão inicialmente descritas as classes utilizadas na representação de conhecimento. Pode-se classificar este conhecimento em 5 grupos:

1. Representação do gasoduto;
2. Representação das conclusões;
3. Representação dos valores dos filtros do PMS;
4. Representação dos dados do SCADA;
5. Representação do mecanismo de inferência e das análises;

A representação dos dados do SCADA está bem descrita na seção 4.3.2. Para representar o gasoduto, foi utilizada uma ontologia semelhante à que foi usada em outros módulos do SEGRED, com pequenas alterações para incluir de forma mais detalhada o conceito de sensor. A Figura 4.3 mostra o diagrama de classes (SCHREIBER et al., 1999, cap. 14) que implementa esta ontologia.

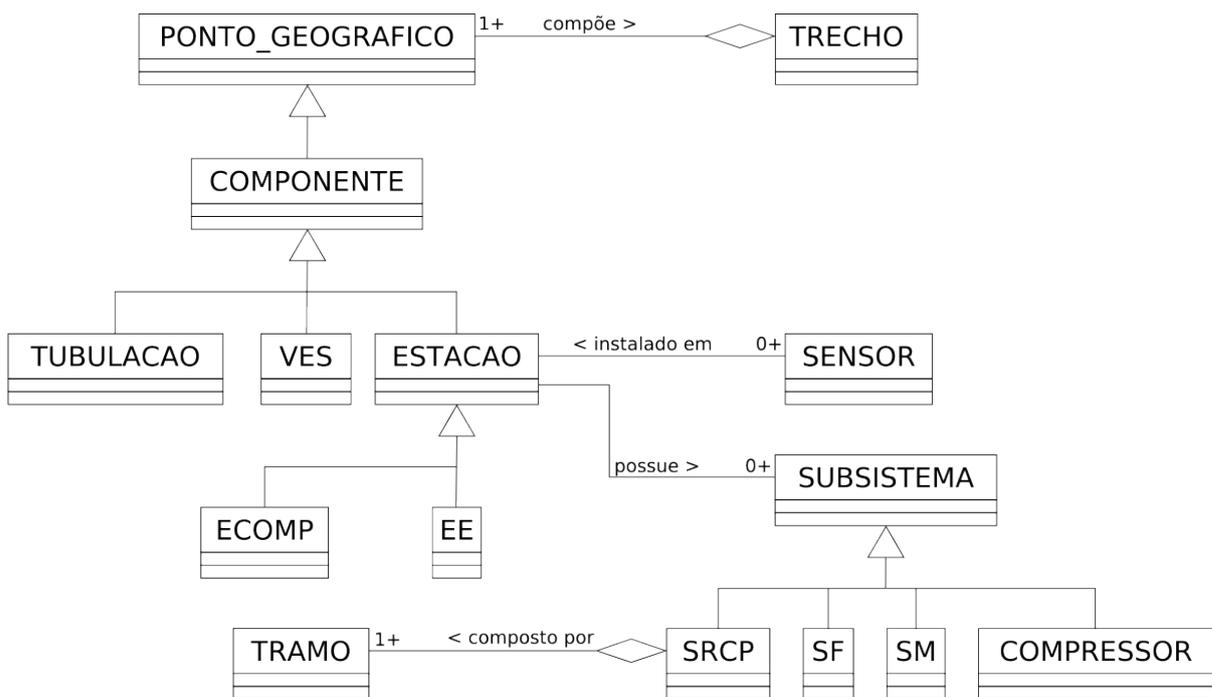


Figura 4.3: Diagrama das classes utilizadas para representar o gasoduto. Estas classes codificam tanto aspectos da geografia do duto quanto aspectos funcionais dos elementos.

Neste diagrama, a classe TRECHO contém uma série de elementos do tipo PUNTO_GEOGRAFICO, que basicamente são elementos que podem ocupar uma posição geográfica bem definida no gasoduto. A cada ponto geográfico está associado um valor que representa a distância, em quilômetros, daquele ponto a um ponto de referência utilizado pela empresa-alvo. A classe TRECHO provê serviços para fazer todas as manipulações de natureza topológica sobre os elementos que ela contém, como, por exemplo, encontrar o elemento de um determinado tipo que

está a jusante de um determinado elemento. A classe PONTO_GEOGRAFICO permite representar qualquer coisa que possa estar sobre a linha do gasoduto, como o local onde está ocorrendo alguma intervenção, o local onde um reparo foi realizado, etc. Nesta implementação o único tipo de PONTO_GEOGRAFICO implementado é o COMPONENTE.

Um COMPONENTE é qualquer componente do gasoduto, como um trecho de TUBULACAO, uma válvula de bloqueio de trecho (VES), ou uma ESTACAO (que pode ser uma EE ou ECOMP). A cada estação está associada uma série de sensores, isto é: cada estação tem uma série de sensores instalados, cada um com suas características de fundo de escala, banda morta, etc. A classe SENSOR armazena todas essas características e ainda encapsula a interface de acesso aos dados do histórico daquele sensor. Além disso, esta classe guarda um atributo que contém uma descrição funcional do sensor. Utilizando este atributo é possível criar uma regra que, por exemplo, dispare para todo sensor que meça a pressão na entrada de uma EE (quer dizer, à montante da estação). As demais classes mostradas no diagrama não são utilizadas pelo módulo de detecção de vazamentos, mas sim por outras funções do SEGRED. Elas representam os diferentes subsistemas que compõe uma estação, como um COMPRESSOR ou o sistema de regulação e controle de pressão (SRCP).

Para a representação das conclusões e dos valores dos filtros do PMS, se utilizou a estrutura descrita na Figura 4.4. A classe DV_CONCLUSAO encapsula as informações necessárias para uma conclusão, como o texto descrevendo a conclusão e a posição dela na hierarquia de situações possíveis apresentada na Figura 4.1. Estas classes são passadas para a classe DV_MENSAGEM_HTML, que produz o arquivo com todas as conclusões do sistema.

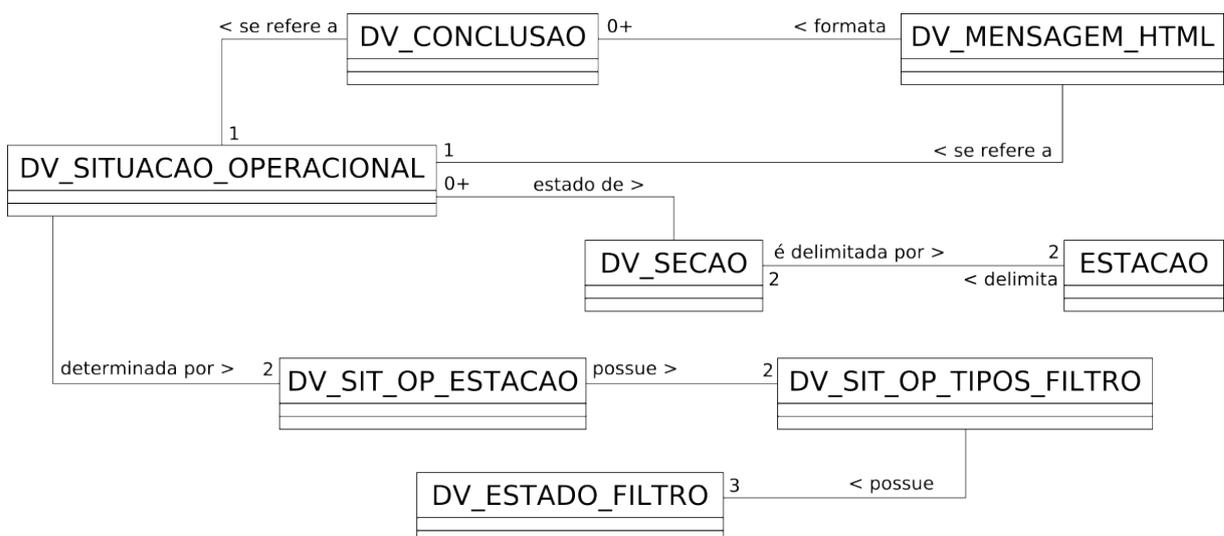


Figura 4.4: Diagrama com as classes utilizadas para a representação dos estados dos filtros do PMS e das conclusões do detector de vazamentos.

A classe DV_SECAO representa uma seção do detector de vazamentos do PMS. A seção é sempre delimitada por duas estações que contém sensores com uma taxa de aquisição alta. Uma configuração específica dos filtros do PMS é representada pela classe DV_SITUACAO_OPERACIONAL. Esta classe contém um indicador do *time stamp*² da situação, e ponteiros para dois objetos da classe DV_SIT_OP_ESTACAO, um representando a estação a montante e o outro a estação a jusante daquela seção. Cada objeto DV_SIT_OP_ESTACAO contém duas referências a objetos da classe DV_SIT_OP_TIPOS_FILTRO. Esta classe indica se o filtro é do tipo UF ou UP. Finalmente, cada objeto da classe DV_SIT_OP_TIPOS_FILTRO referencia três objetos do tipo DV_ESTADO_FILTRO, que representa o valor de um dos filtros do PMS: *slow*, *medium* ou *fast*.

Esta estrutura, talvez excessivamente hierarquizada, é vantajosa quando se deseja fazer asserções sobre todos os filtros de um determinado tipo na estação a jusante da seção, por exemplo. De qualquer forma, caso não se deseje usá-la, ela é completamente transparente para o programador, com a classe DV_SITUACAO_OPERACIONAL fornecendo uma interface simples para acessar todos os valores dos filtros.

Finalmente, o mecanismo de inferência e análises é implementado como regras no CLIPS e como proposições no sistema de inferência difusa. Neste sistema, que será descrito em detalhes na seção 4.3.3, as regras progressivamente combinam proposições em novas proposições cada vez mais abstratas, até as regras que executam a classificação heurística propriamente dita.

Um modo de se visualizar o funcionamento da base de regras é utilizando um recurso que será chamado de diagrama de influência. Este diagrama mostra quais proposições difusas influenciam o valor verdade de cada proposição. Apesar deste diagrama não descrever completamente a base de conhecimento, é um bom esboço das inter-relações entre os conceitos. Estes diagramas também foram utilizados durante o desenvolvimento do sistema, e foram importantes para o desenvolvimento e para a auditoria da base de conhecimento.

Para facilitar tanto a visualização quanto o trabalho na base de conhecimento, esta foi dividida em várias partes. As proposições geradas em cada parte concernem análises de alguns aspectos específicos do problema. Cada uma dessas partes, neste sistema especialista, foi chamada de contexto. Dessa forma, a base de conhecimento trata dos seguintes contextos:

Contexto filtros: Análises associadas apenas aos filtros do PMS;

Contexto análises-sensores: Análises sobre os sensores do SCADA;

Contexto posição-geográfica: Análises que levam em consideração relações geográficas entre

²O *time stamp* é uma indicação inequívoca da data e hora em que foi feita uma determinada medição. O termo foi mantido em inglês para ser fiel aos termos utilizados pelos funcionários da empresa-alvo.

os objetos, como, por exemplo, a distância;

Contexto análise-intermediária: Análises que fundem resultados das análises anteriores;

Contexto análise-final: Regras que executam a classificação heurística.

A Figura 4.5 mostra a cadeia de influência de cada proposição do contexto filtros. É possível perceber como o nível de abstração aumenta gradativamente a partir do valor dos filtros. Nos diagramas, a cadeia de nós na parte inferior, onde está escrito “Ordem 1”, “Ordem 2”, etc indica a profundidade de encadeamento de cada proposição. As proposições que estão na fase de “Inicialização” são criadas a partir do momento em que o sistema é iniciado. Pode-se considerar que essas proposições são os “axiomas” do sistema, uma vez que elas jamais devem ser alteradas. No diagrama, cada nó desta fase corresponde a um conjunto difuso discreto, cujos valores estão na Tabela 4.2.

Tabela 4.2: Valores dos conjuntos difusos utilizados na análise dos filtros.

Conjunto	Valores dos filtros				
	-1	1	2	3	4
Valor-normal	0.2	1.0	0.8	0.3	0.0
Valor-alto	0.0	0.0	0.3	0.8	1.0
Valor-negativo	1.0	0.0	0.0	0.0	0.0

Para que se compreenda melhor o funcionamento do sistema, vale a pena descrever algumas inferências. Um bom exemplo é a cadeia que parte do conjunto Valor-alto (ver Figura 4.5). Este conjunto, cujos valores estão na Tabela 4.2, representa o conceito de valor alto para um filtro do PMS. Ele representa a idéia que, conforme o valor de um filtro do PMS se aproxima de 4 (o máximo), mais plausíveis são as conclusões que dependem do fato dele estar com um valor alto.

Este conjunto é utilizado, por exemplo, pela regra que produz a proposição Alto UF montante ?sit. Esta proposição significa que, para uma determinada situação operacional (a variável ?sit é substituída por um ponteiro para o objeto que representa a situação operacional atual), os valores de todos os filtros UF da estação a montante do trecho do detector de vazamentos estão altos. Esta regra pode ser traduzida em linguagem natural como:

Dada uma situação operacional ?sit, na mesma medida em que for verdade que:

- O valor do filtro UF *slow* da estação que limita o trecho do detector de vazamentos a montante pertence ao conjunto difuso Valor-alto; e



Figura 4.5: Diagrama de influência das análises no contexto de filtros. As análises neste contexto procuram identificar algumas configurações de interesse para os valores dos filtros do PMS, como uma situação indicativa de fechamento de válvula de bloqueio de trecho, ou uma situação normal.

- O valor do filtro UF *medium* da estação que limita o trecho do detector de vazamentos a montante pertence ao conjunto difuso Valor-alto; e
- O valor do filtro UF *fast* da estação que limita o trecho do detector de vazamentos a montante pertence ao conjunto difuso Valor-alto;

então será verdade que Alto UF montante ?sit .

É muito importante notar que o qualificador “na mesma medida que” é um dos principais diferenciais entre um sistema especialista tradicional e um sistema difuso. Essa frase significa que, caso apenas alguns dos filtros estiverem altos, ou, de outro modo, caso os filtros estejam com valores *mais ou menos* altos, então a proposição Alto UF montante ?sit será *mais ou menos* verdade. Desse modo o sistema pode realizar uma gradação entre o absolutamente falso (uma proposição com valor-verdade 0) e o absolutamente verdadeiro (uma proposição com valor-verdade 1).

Este exemplo também ilustra o significado de influência positiva no diagrama de influência, representada por uma seta. Significa que, caso a proposição na cauda da seta seja verdade, então a proposição que está na ponta da seta também deverá ser verdade.

Para exemplificar o significado de influência negativa, e ao mesmo tempo elucidar o que aparentemente é uma contradição no diagrama, será feita uma descrição em linguagem natural da regra que determina a proposição Transiente ?sit :

Dada uma situação operacional ?sit , para cada uma das estações que limitam o trecho do detector de vazamento (*i.e.* à montante e à jusante) e para cada tipo de filtro do detector de vazamentos (*i.e.* UF e UP), na mesma medida em que for verdade que:

- O valor do filtro *slow* da estação que limita o trecho do detector de vazamentos a montante pertence ao conjunto difuso Valor-alto; e
- O valor do filtro *fast* da estação que limita o trecho do detector de vazamentos a montante *não* pertence ao conjunto difuso Valor-alto;

então será verdade que Transiente ?sit .

Em outras palavras, Transiente ?sit só será *absolutamente* verdade caso *todos* os filtros *slow* estiverem altos, e *todos* os filtros *fast* estiverem baixos. Fica claro, nesse exemplo, o significado de influência negativa: ela ocorre quando um valor-verdade alto em uma proposição implica um valor-verdade baixo em outra, como na relação entre “filtro *slow* alto” e Transiente

?sit. Também ficam claras nesse exemplo as limitações de representação do diagrama: não se possuem todos os detalhes do domínio sobre o qual as regras se aplicam. Mas deve estar claro também a utilidade dos diagramas: permitir uma visão global, ainda que não muito detalhada, das regras da base de conhecimento.

A Figura 4.6 mostra as análises realizadas sobre os valores dos sensores do SCADA. Nesta figura, da mesma forma que na anterior, os nós que estão na fase de “Inicialização” correspondem aos conjuntos que existem desde o início do programa. Neste caso, trata-se apenas de conjuntos contínuos, aplicados a análises feitas sobre os sinais do SCADA. Os gráficos de pertinência destes conjuntos não serão mostrados aqui pois, como já foi dito, precisam ser muito melhorados com mais aquisições de conhecimento. Neste contexto, as análises focam principalmente as taxas de variação dos valores dos sensores, a proximidade desses valores dos fundos de escala e alterações nos valores dos indicador de qualidade do dado. As análises também relacionam o tipo de estação onde está ocorrendo a variação.

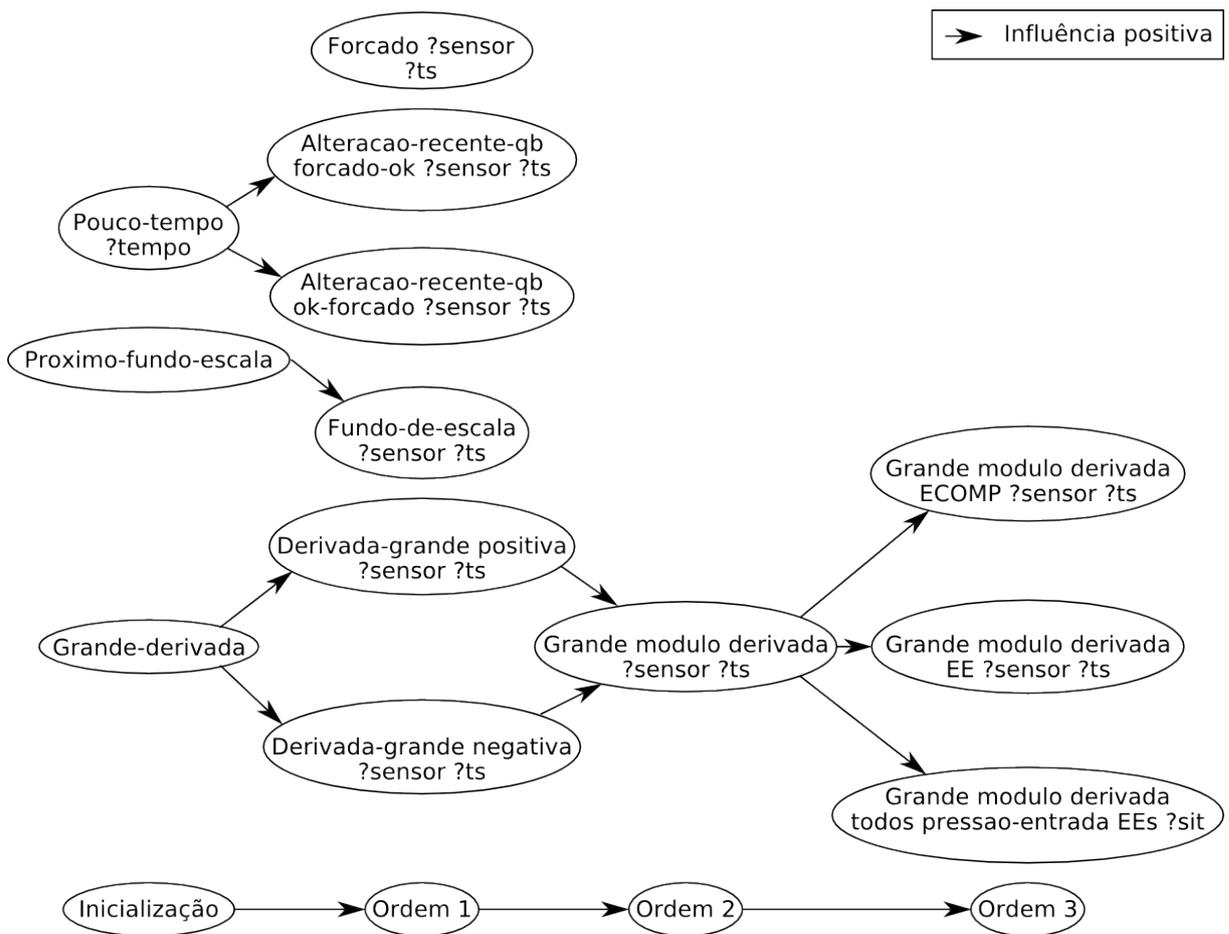


Figura 4.6: Diagrama de influência das análises no contexto de análises dos sensores.

O contexto de posição geográfica, mostrado na Figura 4.7 é o mais simples de todo o sistema, realizando análise apenas da proximidade de uma estação de entrega do vazamento determinada pelo PMS. Neste contexto, próximo significa menor que 5 km.

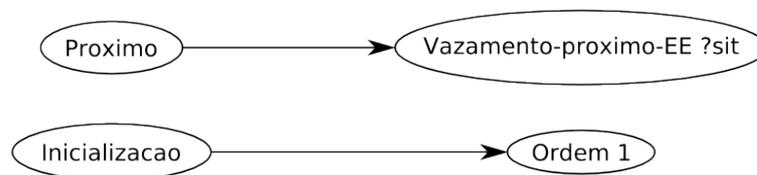


Figura 4.7: Diagrama de influência das análises no contexto de posição geográfica.

Uma vez feitas as análises que levam em consideração apenas um aspecto do problema, as informações resultantes são combinadas em análises intermediárias, que misturam informações de diferentes fontes. A Figura 4.8 mostra estas análises, mostrando também as proposições que servem de base para elas. Percebe-se que não existem muitas análises que levem em consideração ao mesmo tempo dados dos sensores do SCADA e dados dos filtros do sistema de detecção de vazamentos do PMS. Isto se deve, como já foi explicado, ao pequeno número de entrevistas de aquisição de conhecimento. Neste diagrama todas as proposições que não têm nenhuma seta chegando a elas se originam nos contextos mostrados anteriormente. Algumas proposições, que na figura estão com o mesmo nome (como a proposição Empacotando ?sit), na verdade pertencem a contextos diferentes (no caso, o de análise de filtros e o de análises intermediárias).

Finalmente, após as análises intermediárias, que representam o ápice do processo de abstração deste sistema, é feita a classificação. Nesta fase, são geradas proposições com termos idênticos aos mostrados na árvore da Figura 4.1. Esta fase está mostrada na Figura 4.9. Por ela percebe-se que o sistema realiza, no momento, inferências sobre a maior parte dos requisitos apresentados anteriormente, sendo que apenas a detecção de falhas de comunicação não é realizada. Esta é a razão do elemento representando esta solução estar completamente desconectado. Também se percebe o modo como se implementou a detecção de uma situação não-identificada, explicitamente como equivalente a negação de todas as outras possibilidades.

Na figura os retângulos representam a última etapa de processamento antes da geração de conclusões. Nesta fase, cada proposição do tipo Solucao-possivel <solucao> é associada a um dos nós da hierarquia apresentada na Figura 4.1. Caso o nó que esteja na proposição não seja um nó-folha, a pertinência é automaticamente propagada até todos os nós-folha filhos daquele nó. Por exemplo, a pertinência da proposição Solucao-possivel transiente será automaticamente propagada para as soluções mudanca-ECOMP, empacotando e mudanca-EE. Desse modo é possível fazer asserções sobre classes completas de solução. Além disso, apesar de a Figura

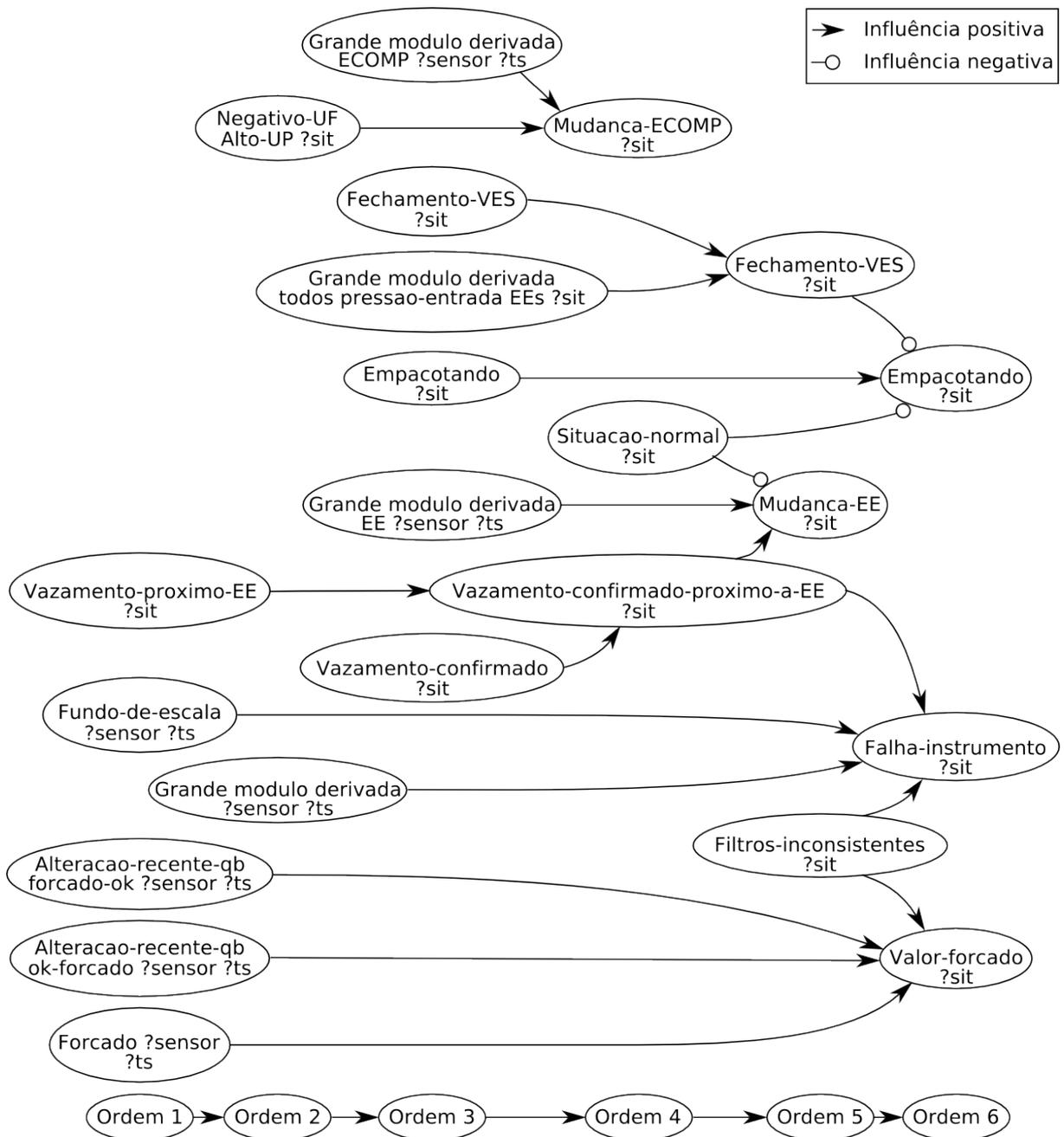


Figura 4.8: Diagrama de influência das análises intermediárias, mostrando também as proposições de outros contextos utilizadas nestas análises.

4.1 representar uma árvore, o sistema funciona para qualquer grafo acíclico direcionado. Outro detalhe da Figura 4.9 é a ausência de nós nos níveis de encadeamento 7 e 8. Isto é para permitir a expansão da base de conhecimento sem que se precise alterar o nível de encadeamento de nenhuma regra.

Enfim, o último processo a ser descrito é o de refinamento. Neste protótipo, este processo se resume a incluir o nome do trecho em algumas soluções. Alguns aspectos de uma implementação mais avançada do refinamento podem ser encontrados na discussão da seção 4.5.

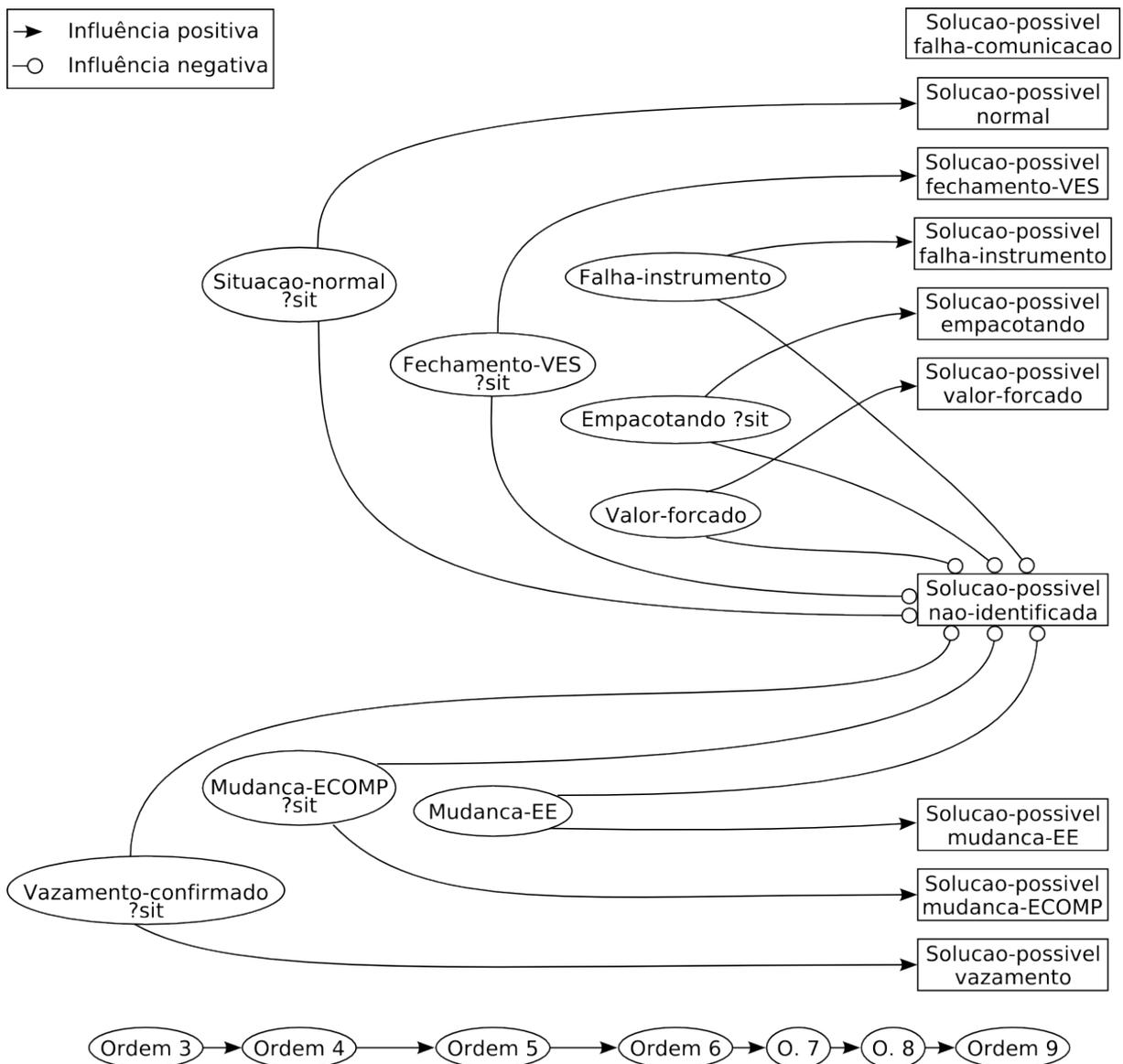


Figura 4.9: Diagrama de influência da parte final das análises. Os nomes dos níveis 7 e 8 foram abreviados para “O. 7” e “O. 8” para tornar o diagrama mais claro.

Um aspecto que vale a pena notar no desenvolvimento deste protótipo foi o problema de integração com a base de conhecimento já desenvolvida. Apesar da base de conhecimento do SEGRED Rede ter sido desenvolvida de modo relativamente modular, a introdução de uma alteração grande como este protótipo corria o risco de introduzir erros que seriam difíceis de localizar. Para evitar esse problema, optou-se por implementar o protótipo do detector de vazamentos como um executável separado, que é executado através da interface do SEGRED Rede. Desse modo evita-se o problema de integração e se mantém uma interface consistente para o usuário. Essa dificuldade é uma confirmação das críticas relacionadas a manutenibilidade de sistemas baseados em regras mostradas na seção 3.1.

4.3.2 Análise dos sinais do SCADA

Nesta seção está descrita a implementação das análises numéricas executadas pelo protótipo. A principal função destas análises é representar o conceito de *trend* utilizado pelos controladores durante as entrevistas de aquisição de conhecimento.

Como já foi explicado anteriormente, devido ao pequeno número de entrevistas realizadas, existem grandes lacunas no conhecimento adquirido. Uma delas é no significado da expressão “analisar o *trend*”. Para esta implementação, decidiu-se que o *trend* de um determinado sinal em um ponto é a derivada do sinal naquele ponto. Como se sabe da teoria de análise de sinais, derivar um sinal tende a diminuir a razão sinal/ruído, o que dificulta a correta interpretação dos dados. Ainda mais quando se considera que os sinais obtidos do SCADA são bastante ruidosos.

Dentre as soluções pesquisadas, Pollock (1999) indica a possibilidade de fazer uma aproximação por polinômio. Essa abordagem tem a vantagem de permitir obter a derivada de forma analítica, a partir do polinômio aproximado. Tem ainda a vantagem de poder ser aplicada mesmo para uma série temporal que não é regularmente espaçada, o que é o caso de todos os dados presentes no SCADA. Portanto, esta foi a estratégia adotada.

Ao se buscar a derivada em um determinado ponto de um sensor, um polinômio (que, após alguns experimentos, foi fixado como de grau 3) será aproximado sobre a série que se inicia 5 horas antes do instante pedido, e que termina exatamente no instante pedido (quando este existe realmente no SCADA), ou no instante presente no banco de dados mais próximo do instante pedido (quando não existe no SCADA nenhuma leitura para o instante pedido, que é o caso mais comum). Caso esse período de 5 horas contenha menos de 5 pontos, os últimos 5 pontos antes do instante pedido serão utilizados, independentemente do *time stamp* dos pontos.

Uma vez determinado o número de pontos, é realizada a aproximação utilizando um método de mínimos quadrados generalizado (PRESS et al., 2002, cap. 15), para encontrar os 4 coeficientes do polinômio. Este método utiliza uma decomposição em valores singulares (SVD) para encontrar a solução. O método foi implementado baseando-se no algoritmo descrito em (PRESS et al., 2002, cap. 2), com pequenas alterações. Este algoritmo tem a vantagem de ser numericamente estável e rápido. Para fazer com que a aproximação funcione melhor próximo do ponto de interesse, é imposta uma restrição sobre o polinômio: a segunda derivada do polinômio no ponto de interesse deve ser nula.

Como um modo de melhorar o algoritmo, foram utilizados pesos sobre a série, fazendo com que os pontos mais próximos do ponto de interesse tenham um peso maior na solução que os pontos mais distantes. A Figura 4.10 mostra uma comparação do sinal com o polinômio

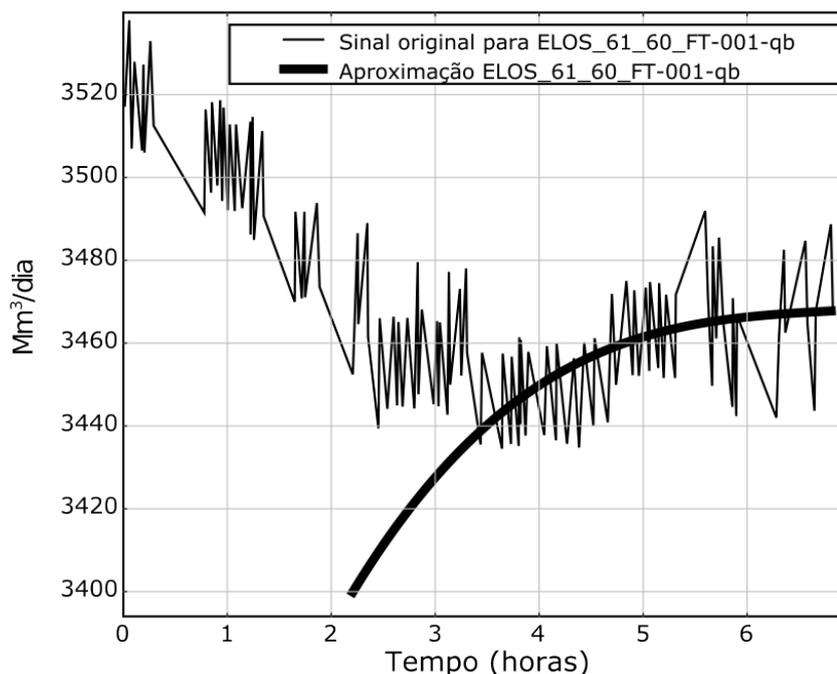


Figura 4.10: Exemplo de aproximação produzida pelo algoritmo. A unidade do eixo das ordenadas é milhares de metros cúbicos por dia, que é a unidade padrão utilizada pela empresa-alvo para medição de vazão.

aproximador. Note como a curva se aproxima mais do ponto de interesse (ponta direita do sinal), e se afasta bastante do sinal original conforme ela se afasta desse ponto, por resultado da aplicação dos pesos. Esta curva foi produzida com dados do sensor ELOS_61_60_FT-001-qb de 19 de janeiro de 2006, das 2 às 9 horas da manhã.

Esse algoritmo foi considerado satisfatório, uma vez que, na avaliação do engenheiro de conhecimento, a tendência calculada se aproximava da tendência do sinal. Maiores detalhes sobre os testes deste algoritmo se encontram na seção 4.4.1.

Apesar do método implementado ser simples, os testes realizados (ver seção 4.4.1) indicaram um bom desempenho. Mesmo assim, este é um dos pontos do protótipo que poderia ser bem aprimorado com mais aquisição de conhecimento.

Além de realizar as análises numéricas, esse subsistema é o responsável por ler os arquivos dos sensores armazenados pelo SEGRED, e garantir que eles estejam sempre atualizados. Como as tarefas realizadas por esse subsistema são basicamente análises numéricas e de acesso a dados, o CLIPS não é uma linguagem conveniente para o desenvolvimento. Assim, foi escolhida a linguagem C++ para a implementação do subsistema, devido a facilidade em interfacear com o CLIPS e a facilidade para se implementar análises numéricas. Como as análises realizadas pelo sistema são relativamente simples, optou-se por utilizar uma biblioteca extremamente

simples para a representação de matrizes e vetores³ e implementar diretamente os outros algoritmos necessários.

A Figura 4.11 mostra o diagrama de classes da implementação realizada. Este diagrama mostra tanto as classes que representam os sensores quanto as classes utilizadas para realizar as análises numéricas.

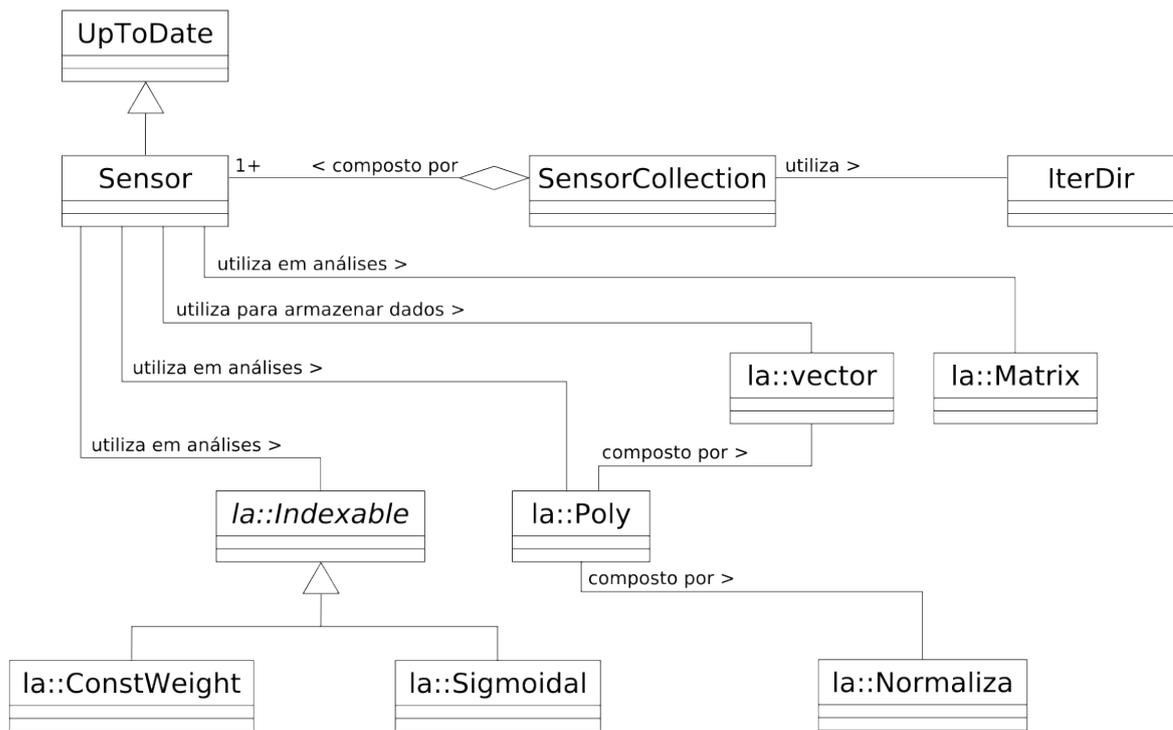


Figura 4.11: Diagrama de classes da implementação das análises sobre os dados dos sensores.

Neste diagrama, as classes prefixadas com “la” são as classes relacionadas com as operações de álgebra linear. As classes `la::vector` e `la::Matrix` representam, respectivamente, vetores e matrizes. A classe `la::Normaliza` permite encapsular as ações e constantes necessárias para se normalizar um vetor, isto é, fazer com que todos os valores do vetor se mantenham entre 0 e 1. A normalização é necessária para aumentar a estabilidade da decomposição em valores singulares. Esta classe também permite aplicar a transformação inversa, isto é, “desnormalizar” o vetor. Neste caso, os valores do vetor retornam à condição original.

A classe `la::Poly` representa um polinômio. Ela utiliza a classe `la::vector` para representar os coeficientes do polinômio. Esta classe também permite a associação de duas transformações, representadas pela classe `la::Normaliza`, uma à variável dependente do polinômio e outra à variável independente. Isso também foi feito para auxiliar na estabilidade das soluções numéricas.

³Criada por Steve Pelikan e disponível em <http://math.uc.edu/~pelikan/CODE/code.html>.

A classe abstrata `la::Indexable` expõe uma interface semelhante a classe `la::vector`, mas permite que, ao invés de utilizar dados discretos associados a cada elemento do vetor, se defina uma função analítica que será avaliada para cada elemento. Esta classe foi criada para facilitar a implementação dos pesos no algoritmo de aproximação. Esta classe tem duas classes filhas: `la::ConstWeight`, que faz com que todos os pontos da série tenham o mesmo peso, e `la::Sigmoidal`, que faz com que os pesos sejam distribuídos de acordo com uma função sigmoideal, com os maiores pesos mais próximos do ponto de interesse.

As classes `UpToDate`, `Sensor`, `SensorCollection` e `IterDir` estão associadas ao acesso aos dados dos SCADA. No SEGRED, os dados do SCADA ficam armazenados em arquivos, um para cada sensor. Assim, a classe `SensorCollection` fornece uma interface para carregar os valores de cada sensor. Ela utiliza a classe `IterDir` como classe auxiliar para iterar sobre os arquivos no diretório onde estão armazenados os dados dos sensores. Para cada arquivo encontrado, a classe `SensorCollection` instância um objeto da classe `Sensor`. Esta classe fornece uma interface para o acesso a esses dados e implementa todas as análises numéricas descritas anteriormente. A classe `Sensor` descende da classe `UpToDate`, que fornece serviços para sincronizar os valores armazenados na própria classe com os valores armazenados em um arquivo, sempre que estes mudarem. Isto é necessário porque a cada 30 min o SEGRED recebe novos dados do SCADA.

4.3.3 Sistema de inferência difusa

Para realizar o tratamento de incertezas foi desenvolvido para este protótipo um sistema de inferência difusa baseado no que foi apresentado na seção 3.2.3. Esta seção descreve este sistema.

O desenvolvimento feito na seção 3.2.3 tem como proposição atômica uma variável difusa, que foi definida como a pertinência de um determinado elemento do universo de discurso a um conjunto difuso específico. Também foi definido que no lado direito de uma regra só poderiam existir proposições atômicas. Esta característica foi utilizada nesta implementação para criar uma sintaxe intuitiva para o acesso às proposições. Neste sistema, uma proposição tem a seguinte forma:

$$\langle \text{Proposição} \rangle ::= \langle \text{Predicado} \rangle \langle \text{termo} \rangle [\langle \text{termo} \rangle]^* [\text{contexto } \langle \text{contexto} \rangle]$$

onde `::=` significa definição, os termos que não estão entre “<” devem ser tomados literalmente, os termos que estão entre “[]” podem aparecer uma ou nenhuma vez, e o termo que precede “*” pode aparecer quantas vezes for necessário.

A proposição é composta por um predicado, que se aplica a um ou mais termos. O predicado funciona como um conjunto difuso, normalmente discreto, e retorna o valor de pertinência da tupla formada pelos vários <termo>. Por exemplo, a proposição Alto 2, é equivalente ao grau de pertinência do número 2 ao conjunto Alto. Do mesmo modo, a proposição Proximo-fundo-escala sensor FT-001 tempo 10 é equivalente a pertinência da tupla (sensor, FT-001, tempo, 10) ao conjunto difuso Proximo-fundo-escala.

O <contexto> serve para diferenciar entre predicados com o mesmo nome, mas que se aplicam a contextos diferentes. Por exemplo, no contexto de filtros de detecção de vazamentos, o predicado Alto pode ter um significado diferente que o predicado Alto em se tratando de posição geográfica. Um filtro do PMS com valor 4 é considerado alto, mas, em um contexto de análise de dados altimétricos de estações, Alto pode ser definido como acima de 800 m de altitude, por exemplo. Na prática, o contexto e o predicado determinam univocamente um conjunto difuso. Em termos de implementação, a base de conhecimento é composta por uma série de objetos representando os diferentes contextos. Cada objeto representando um contexto é um vetor associativo, que associa o nome do predicado a um objeto descrevendo aquele conjunto difuso.

Do ponto de vista do usuário do sistema, existem duas atividades que se pode fazer em uma base de conhecimento difusa. A primeira é fazer uma consulta, que retorna o valor verdade de uma determinada proposição. Para esta consulta, utiliza-se a seguinte função:

(p <Proposição>)

Para fazer uma asserção, quer dizer, estabelecer o valor de pertinência de um determinado predicado, utiliza-se a função:

(assert-p <Proposição> <pertinência>)

Esta função, que tem uma sintaxe semelhante a função (p), associa um valor-verdade <pertinência> à proposição <Proposição>. Existem duas opções possíveis para esse valor-verdade: ele pode ser um número de ponto flutuante entre 0 e 1, ou pode ser um objeto representando um conjunto contínuo, caso em que não podem ser utilizados os [<termo>]*. Assim se permite, utilizando a mesma interface, representar uma gama muito ampla de proposições diferentes.

Como foi tratado na seção 3.2.3, é comum ocorrer mais de uma asserção sobre a mesma proposição, como no exemplo a seguir.

```
(assert-p Com-problema sensor FT-001 contexto sensores 0.9)
(assert-p Com-problema sensor FT-001 contexto sensores 0.45)
```

Neste caso, é preciso agregar os dois valores de pertinência de algum modo significativo. Neste sistema, essa agregação pode ser controlada globalmente redefinindo a função (agregacao). Portanto, alterando a definição dessa função, pode-se alterar globalmente o modo como é feita a combinação de valores-verdade.

Para aumentar a expressividade do sistema também foram considerados, de forma simplificada, dois tipos de asserções propostas em (DUBOIS; PRADE, 1996). O objetivo dessas asserções é definir a veracidade mínima e/ou máxima de uma determinada proposição. Para isso, existem duas funções:

```
(assert-p-min <Proposição> <pertinência>)
(assert-p-max <Proposição> <pertinência>)
```

Estas funções se aplicam somente a proposições implementadas como conjuntos discretos. Elas definem, respectivamente, o valor verdade mínimo e máximo da proposição, independente do valor retornado pela função de agregação. Assim, por exemplo, a seqüência de comandos:

```
(assert-p-min Vazamento Situacao-0 0.5)
(assert-p Vazamento Situacao-0 0.3)
```

estabelece que (p Vazamento Situacao-0) deve retornar 0,5 como pertinência, e não 0,3.

As funções apresentadas anteriormente definem o funcionamento do sistema de proposições. Para utilizá-las convenientemente, é preciso determinar o funcionamento das regras de produção do sistema. Como definido anteriormente, uma regra é da forma:

$$R_k : \text{Se } \underline{h}_k(\underline{a}_1, \dots, \underline{a}_n) \text{ então } \underline{b}_1, \dots, \underline{b}_m \text{ com grau de certeza } r_k$$

onde k é o número da regra, $\underline{a}_1, \dots, \underline{a}_n$ e $\underline{b}_1, \dots, \underline{b}_m$ são proposições como definidas anteriormente e r_k é a confiança na regra. A função \underline{h}_k é uma função difusa qualquer, composta de funções difusas mais simples.

Para implementar estas regras difusas no sistema especialista, cada uma foi inserida dentro de uma regra de produção padrão do CLIPS. O encadeamento de regras, para que cada regra só dispare após todas as regras que alteram as proposições em seu antecedente tenham sido disparadas, foi implementado de forma semi-automática. Cada regra está associada a um nível de inferência, e o sistema garante que todas as regras de um determinado nível disparem antes de qualquer regra do nível seguinte. Isto é realizado utilizando-se fatos de controle.

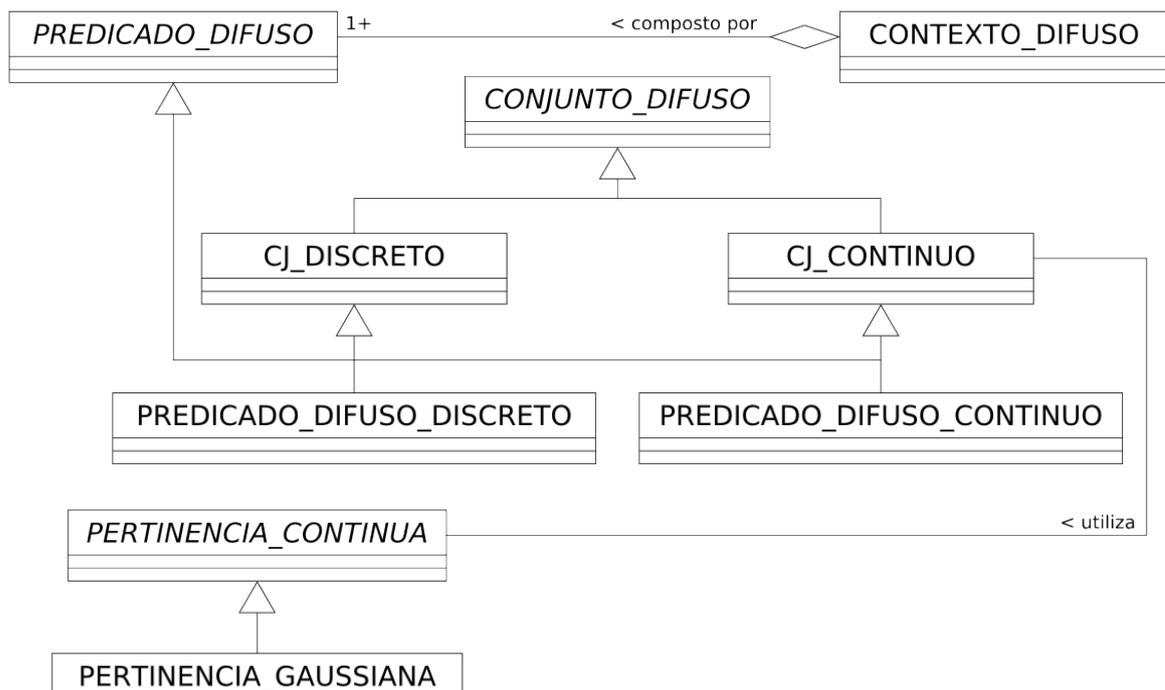


Figura 4.12: Diagrama das classes utilizadas no sistema de inferência difusa.

O sistema de proposições foi implementado utilizando a estrutura de classes mostradas na Figura 4.12. Esta arquitetura tem alguma semelhança com a descrita em (ŠENDELJ; DEVEDŽIĆ, 2004). Como se utiliza o sistema de produção para representar as regras, não foi necessário criar classes para representar os vários componentes de uma regra. Assim, a estrutura mostrada na figura representa apenas as proposições e os conjuntos difusos associados.

A classe CONTEXTO_DIFUSO fornece os serviços necessários para implementar a funcionalidade de contextos explicada anteriormente. Ela é composta basicamente por um vetor associativo que mapeia a cada nome de contexto um objeto do tipo PREDICADO_DIFUSO. A classe PREDICADO_DIFUSO é utilizada para representar um predicado. Esta classe delega praticamente toda a implementação para uma classe do tipo CONJUNTO_DIFUSO.

A classe CONJUNTO_DIFUSO tem duas implementações possíveis: a classe CJ_DISCRETO implementa a funcionalidade para associar valores de pertinência a tuplas e realizar a agregação quando são feitas várias asserções sobre a mesma tupla. A classe CJ_CONTINUO é bem mais simples e implementa um conjunto difuso contínuo. A função de pertinência do conjunto é definida por um objeto do tipo PERTINENCIA_CONTINUA. Objetos desse tipo nada mais são que funções definidas sobre o intervalo dos reais que retornam valores entre 0 e 1. No protótipo que foi implementado o único tipo de função que foi utilizado foi uma função gaussiana, implementada pela classe PERTINENCIA_GAUSSIANA.

4.4 Testes e avaliação de performance

Existem basicamente dois tipos de testes que são realizados com um sistema especialista. O primeiro, chamado de verificação, é realizado dentro da própria equipe de desenvolvimento. Consiste em verificar que o *software* funciona de acordo com a lista de requisitos que a equipe possui. O segundo tipo, chamado de validação, é realizado junto a especialistas e/ou usuários do *software*, normalmente em demonstrações. Este teste busca determinar interpretações incorretas dos requisitos, e, no caso de sistemas especialistas, interpretações incorretas do conhecimento adquirido. Como já foi exposto anteriormente, a empresa-alvo não realizou nenhuma validação do sistema, portanto este teste não pôde ser realizado. Sendo assim esta seção tratará apenas dos testes de verificação do sistema.

Serão mostrados os métodos utilizados para testar as análises numéricas, e os casos utilizados no teste do sistema como um todo.

4.4.1 Testes realizados sobre o subsistema de análise numérica

Como foi explicado na seção 4.3.2, a principal função desse subsistema é fornecer uma medida da derivada dos sinais do SCADA. Como não foram realizadas entrevistas de aquisição de conhecimento suficientes, não se possuía nenhum parâmetro objetivo para avaliar os resultados dos testes. Sendo assim, o teste deste subsistema foi efetuado levando em consideração apenas a intuição do engenheiro de conhecimento, não sendo possível realizar os testes de maneira mais rigorosa.

Devido ao nível de ruído do sinal, e ao pouco tempo disponível para desenvolver todo o subsistema de análise de sinais, os testes foram realizados apenas comparando visualmente a curva obtida pelo sistema com a curva original do sensor, e observando se as duas tinham a mesma tendência. Ainda que intuitivo, esse processo foi realizado de maneira metódica, sobre as curvas dos sensores analógicos atualmente sendo adquiridos pelo SEGRED. Os dados de 18 a 22 de janeiro de 2006 foram analisados em busca de casos “patológicos”, como erros de leitura e variações bruscas. Além disso, cerca de 100 testes sobre pontos escolhidos aleatoriamente, dentro desta faixa, foram realizados.

O resultado dos testes foi positivo, significando que o algoritmo forneceu, na maior parte dos casos, resultados compatíveis com o esperado pelo engenheiro de conhecimento. Apenas na proximidade de erros de fundo de escala o sistema não forneceu uma boa previsão da tendência do sinal.

Tabela 4.4: Resultados dos testes para as situações bem determinadas.

Caso	Situação testada	Resultado	Comentários
1	Situação normal	✓	Identificou corretamente.
2	Fechamento VES	✓	Identificou corretamente.
3	Vazamento	✓	Identificou corretamente.
4	Vazamento próximo a EE	✓	Identificou a possibilidade de defeito em instrumentação da EE e a possibilidade do vazamento.
5	Empacotamento ocorrendo no duto	✓	Identificou corretamente.
6	Mudança na vazão de uma ECOMP	✓	Identificou corretamente.

Tabela 4.5: Resultados dos testes para as situações mal determinadas.

Caso	Situação testada	Resultado	Comentários
7	Transiente	✓	Identificou corretamente como situação normal.
8	Transiente	✓	Identificou corretamente como situação normal.
9	Transiente	✓	Identificou corretamente como situação normal.
10	Transiente	✓	Identificou corretamente como situação normal.
11	Transiente	✓	Identificou corretamente como situação normal.
12	PMS indicando vazamento, mas não com todos os filtros com valor 4	×	Identificou como situação não reconhecida.
13	PMS indicando vazamento sobre EE, mas não com todos os filtros com valor 4	×	Identificou como situação não reconhecida.
14	Transiente, com fortes indicações de empacotamento	×	Identificou como situação não reconhecida.
15	Transiente, com fortes indicações de empacotamento	×	Identificou como situação não reconhecida.

✓ – Passou no teste; × – Não passou no teste.

Assim, os casos em que se considerou falha nos testes devem ser vistos na verdade como lacunas no conhecimento adquirido. Decidiu-se que não se deve ajustar o sistema para os 4 últimos casos de teste pois não se sabe qual a classificação correta para essas situações. Assim, ao invés de ver estes resultados como falhas no sistema especialista, eles devem ser vistos como oportunidades para melhorar o sistema. Em especial, os casos 12 e 13 são interessantes, e são bons casos para iniciar uma nova etapa de aquisição de conhecimento.

4.5 Arquitetura proposta para o módulo de explicação

Considera-se uma característica importante de sistemas especialistas que eles sejam capazes de fornecer uma explicação para o seu “raciocínio”. Isto auxilia na validação do conhecimento presente na base de conhecimento e permite avaliar criticamente o resultado da inferência do sistema especialista. Por outro lado, o simples fato de prover uma explicação faz com que a percepção do usuário final de que a resposta fornecida pelo sistema está correta aumente (SWINNEY, 1995). Não obstante este efeito potencialmente deletério da explicação, ainda considera-se que o módulo de explicação é um componente desejável em um sistema especialista. Por esta razão, o desenvolvimento deste módulo faz parte da lista de requisitos do protótipo (requisito 12). Esta seção descreve uma proposta para este módulo.

Um dos métodos mais comuns para gerar a explicação em um sistema especialista é manter um registro de todas as regras disparadas, anotando-se os fatos que influenciaram no disparo de cada regra e utilizar esse registro como base para produzir a explicação (WICK; SLAGLE, 1989). Um dos métodos para se gerar as explicações é a utilização de gabaritos com todas as respostas possíveis do sistema, e utilizar as informações presentes no registro para preencher as lacunas nos gabaritos. Uma abordagem deste tipo pode ser vista em (PAPAMICHAIL; FRENCH, 2003).

Um dos problemas em se adotar esta abordagem no sistema que foi implementado é o fato de, como já foi explicado na seção 3.2.3, todas as regras do sistema dispararem. Nesse caso, o registro ficaria cheio de dados que contribuiriam pouco para as conclusões do sistema. Mostrar para o usuário final todas as entradas do registro seria então contraproducente, uma vez que a maior parte das informações mostradas teria pouca influência no resultado final.

Em vista disso, esta seção visa propor um método para se determinar quais as informações que foram mais influentes na solução produzida pelo sistema especialista. Para isso será mostrado como o registro da inferência do sistema pode ser representado por uma estrutura em forma de árvore e será proposto um algoritmo para extrair dessa estrutura uma sub-árvore representando apenas as posições mais influentes na solução final.

Para registrar as etapas de inferência do sistema é importante notar que o valor de uma proposição \underline{p} é definido realizando-se uma agregação, representada pela função \underline{g} , sobre todas as n regras que têm \underline{p} em seu conseqüente. Assim, sendo $\underline{p}_1, \dots, \underline{p}_k, \dots, \underline{p}_n$ todos os valores para \underline{p} produzidos pelas n regras, temos que:

$$\underline{p} = \underline{g}(\underline{p}_1, \dots, \underline{p}_k, \dots, \underline{p}_n) \quad (4.1)$$

Cada \underline{p}_k é definido em função dos antecedentes da regra R_k , que serão chamados de $\underline{a}_{1k}, \dots, \underline{a}_{m_k}$, onde m_k é o número de antecedentes da regra R_k . Como já foi visto, \underline{p}_k pode ser calculado por:

$$\underline{p}_k = \underline{w} \left(\underline{a} \left(r_k, \underline{h}_k(\underline{a}_{1k}, \dots, \underline{a}_{m_k}) \right) \right) \quad (4.2)$$

Onde \underline{h} é a função difusa aplicada sobre os antecedentes da regra R_k , \underline{a} é a função de ativação e \underline{w} é uma função opcional que pode ser aplicada ao resultado da ativação.

É fácil perceber, observando-se as Equações 4.2 e 4.1, que é possível representar toda a inferência que leva ao valor de \underline{p} como uma árvore. A Figura 4.13 mostra essa árvore. Nesta árvore, cada nó representa a aplicação da função definida naquele nó sobre todos os valores provenientes dos nós inferiores.

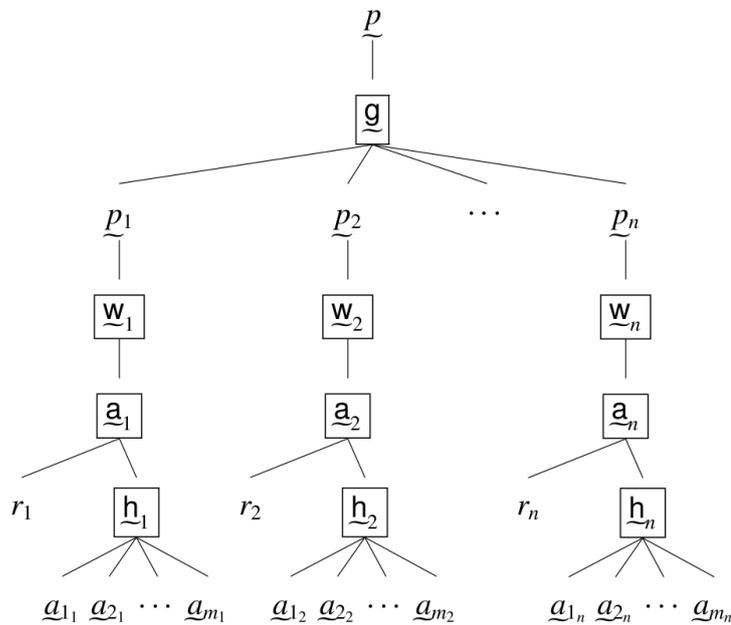


Figura 4.13: Árvore de inferência para a proposição \underline{p} .

A árvore mostrada na Figura 4.13 representa apenas algumas regras de uma cadeia de inferência que pode ter várias centenas de regras. Para gerar uma mensagem explicativa a partir de uma árvore desse tipo pode-se associar a cada regra um texto explicativo, ou um gabarito para

gerar um texto explicativo. O algoritmo pode então percorrer a árvore e montar a explicação da inferência a partir da explicação de cada regra. No entanto, é preciso selecionar apenas as regras que mais influenciaram no valor de cada proposição. Isso equivale a observar a função de agregação \underline{g} e determinar quais de seus argumentos mais influenciaram seu resultado.

Para realizar esta tarefa, é preciso definir algum critério que represente “influência”. Será proposto a combinação de dois critérios de influência: o nível de certeza de cada argumento e a sensibilidade de \underline{g} a uma variação em cada um de seus argumentos.

O nível de certeza pode ser definido como a distância do valor de uma determinada variável difusa de $\frac{1}{2}$. Define-se que este valor deve variar entre 0 a 1, onde 0 significa completamente ambígua e 1 significa completamente certa. Assim, $C(a)$ é:

$$C(a) = 2 \left| a - \frac{1}{2} \right| \quad (4.3)$$

Utilizando esta definição, é possível estabelecer que quanto maior o nível de certeza do conseqüente de uma regra, maior a sua importância para a explicação. Vale a pena ressaltar que esta definição de nível de certeza pode ser transformada na definição de uma entropia difusa. Basta fazer:

$$e(\underline{A}) = \sum_{x \in X} \left(1 - 2 \left| \mu_{\underline{A}}(x) - \frac{1}{2} \right| \right) \quad (4.4)$$

É fácil provar que esta definição está de acordo com a definição de entropia difusa (FAN; MA, 2002). Isto dá suporte para o uso de C como critério para a determinação de influência.

A segunda medida que será usada é a sensibilidade de \underline{g} a mudanças em cada um de seus argumentos. Especificamente, propõe-se testar a variação de \underline{g} substituindo-se seus argumentos um a um por $\frac{1}{2}$, enquanto os outros se mantêm constantes. Esta medida pode indicar um argumento que, apesar de ter um nível de certeza pequeno, causa uma grande variação no resultado caso tenha seu nível de certeza diminuído.

O resultado da avaliação destes dois critérios pode ser utilizado para selecionar quais os argumentos de \underline{g} têm mais influência sobre a função. Um algoritmo pode, por exemplo, considerar 20% dos argumentos com maior grau de certeza e 20% dos argumentos aos quais \underline{g} é mais sensível e utilizá-los na geração da explicação.

Este método de explicação não pôde ser implementado no protótipo por falta de tempo. No entanto, considera-se que sua implementação é pouco complexa, e capaz de produzir resultados aceitáveis. Além disso, a árvore descrita na Figura 4.13 pode ser utilizada no processo de refinamento das soluções obtidas pelo sistema especialista.

4.6 Comentários sobre o desenvolvimento

Pelo que foi dito anteriormente, percebe-se que o desenvolvimento do módulo de detecção de vazamentos do SEGRED sofreu muito devido aos atrasos da empresa-alvo. Estes atrasos ocorreram sobretudo nas seguintes tarefas:

1. Definição de requisitos;
2. Entrevistas de aquisição de conhecimento;
3. Avaliação dos protótipos;
4. Liberação do acesso aos dados do SCADA.

Para que se possa avaliar melhor o impacto desses atrasos, a Tabela 4.6 mostra qual o estado de implementação dos requisitos apresentados na seção 4.1 (pág. 61). É possível, por esta tabela, determinar a influência de cada atraso no projeto. Assim, vê-se que o atraso de um ano e quatro meses na liberação do acesso aos dados do SCADA impediu diretamente que 8 requisitos fossem corretamente implementados; e os atrasos na definição de requisitos e o pequeno número de entrevistas de aquisição de conhecimento realizadas também impediram que 8 requisitos⁴ fossem implementados satisfatoriamente.

Tabela 4.6: Situação da implementação de cada um dos requisitos do sistema.

Req.	Situação	Comentários
1	±	É necessário aprofundar as análises de sinais.
2	√	Pode ser melhorada implementando análises sobre um histórico de valores do PMS. Precisa ser melhor testada.
3	×	No momento, o SEGRED não recebe esse tipo de informação do SCADA. Seria necessária mais uma entrevista de aquisição para determinar quando se deve avisar uma falha de comunicação.
4	±	É preciso aprofundar as análises de sinais e do comportamento do PMS.
5	±	É preciso aprofundar as análises de sinais e do comportamento do PMS.
6	±	É preciso aprofundar as análises de sinais e do comportamento do PMS.

Continua na próxima página

⁴Respectivamente, os requisitos 1, 4, 5, 6, 7, 8, 11, 16b e 3, 4, 5, 6, 7, 8, 12, 17.

Continuação da Tabela 4.6

Req.	Situação	Comentários
7	±	É preciso aprofundar as análises de sinais e do comportamento do PMS.
8	±	É preciso aprofundar as análises de sinais e do comportamento do PMS.
9	√	As análises podem ser refinadas.
10	√	Simplemente determina se nenhuma outra situação foi identificada.
11	×	Devido aos atrasos para disponibilizar os dados do SCADA, não houve tempo hábil para desenvolver as análises necessárias.
12	×	Devido aos atrasos nas entrevistas de aquisição de conhecimento, não foi possível implementar esse módulo.
13	×	Este tipo de requerimento não se aplica a um protótipo.
14a	√	A implementação foi feita dentro dos limites impostos pela empresa-alvo, que permitem um período de atualização dos dados de apenas 30 min.
14b	×	A empresa-alvo não possui nenhum método para acessar esses dados.
15	√	Como não existe modo automático para se acessar os dados do PMS, a interface provê um modo para realizar a entrada de dados manualmente.
16a	√	Foram realizados teste para determinar as consistências das respostas. Os casos de teste utilizados encontram-se na seção 4.4.
16b	×	Apesar do arcabouço básico para as análises ter sido implementado, não houve tempo suficiente para testar as análises, pois este procedimento é bastante laborioso.
17	×	Não foi realizada pela empresa-alvo.

√ – Implementado; ± – Implementação parcial; × – Não implementado.

O interessante ao se observar as causas dos atrasos é que são dois aspectos bem tratados na literatura de sistemas especialistas: o interesse gerencial no projeto e o acesso aos especialistas. Com relação ao interesse gerencial, Schreiber et al. (1999, cap. 3) indicam que, em processos de engenharia de conhecimento bem sucedidos, a identificação dos envolvidos no projeto (os chamados *stakeholders*) e a correta compreensão da estrutura da empresa são fatores cruciais. Outro aspecto apontado é a importância do sistema para a companhia, ou seja: quanto menor

é o problema que o sistema busca resolver tanto menor será a importância dada ao projeto e menores as chances dele ser bem sucedido.

O objetivo do sistema descrito é ser um estudo de viabilidade tecnológica, representando um pequeno avanço com relação aos métodos empregados atualmente pela empresa-alvo. Sendo assim, e devido à característica essencialmente operacional da empresa-alvo, é natural que ao se contrastar um projeto que vá apresentar resultados palpáveis apenas a longo prazo com necessidades operacionais prementes, o projeto que apresentará resultados apenas a longo prazo passe a ter menor prioridade. Portanto é de se esperar que ela direcione os recursos disponíveis para as necessidades mais imediatas. Isto será ainda mais verdade caso estes recursos sejam escassos, o que é o caso dos recursos humanos da empresa-alvo. Acredita-se que isto explique os problemas de interesse gerencial no projeto.

A segunda questão está relacionada com a aquisição de conhecimento e com o acesso aos especialistas. A aquisição de conhecimento é sempre considerada o gargalo no processo de desenvolvimento de um sistema especialista (HART, 1992; DYM; LEVITT, 1991; SCOTT et al., 1991). Na literatura pesquisada costuma-se comentar dois problemas principais de acesso ao especialista: o tempo que o especialista tem disponível para as tarefas de aquisição de conhecimento e o interesse que o especialista tem no desenvolvimento do sistema.

Como já foi comentado, a empresa-alvo apresentava escassez de recursos humanos. Isto implica escassez de tempo dos especialistas. E de fato, o principal especialista do projeto teve menos tempo para se dedicar às entrevistas de aquisição de conhecimento que um projeto deste porte demanda. Os controladores do gasoduto que participaram de entrevistas de aquisição também tiveram pouco tempo disponível, uma vez que, durante as entrevistas, ainda tinham que executar suas tarefas de turno. Isto contribuiu para a dificuldade no desenvolvimento do protótipo.

5 Conclusão e recomendações para trabalhos futuros

Apesar das técnicas de detecção de vazamentos por *software* serem bem desenvolvidas, estas técnicas normalmente têm um desempenho limitado pelos tipos de sensores já instalados no duto. Devido a esta limitação, estas técnicas não costumam apresentar o mesmo desempenho que técnicas que adotam a instalação de novos sensores no duto para a detecção de vazamentos. Por outro lado, a detecção de vazamentos por *software* costuma ser a alternativa mais econômica para se obter um monitoramento constante do duto, o que a torna uma opção atraente para a solução do problema.

Na detecção de vazamentos por *software*, várias análises são efetuadas sobre os sinais obtidos dos sensores presentes no duto, normalmente levando em consideração as características físicas do duto. Estas análises carregam consigo outras informações além das necessárias para a detecção de vazamentos e podem indicar outras situações de interesse para a empresa controladora do gasoduto. E de fato os controladores do duto costumam utilizar informações provenientes de sistemas de detecção de vazamentos na tomada de decisões relacionadas a vários aspectos da operação do duto.

Isto indica que um dos caminhos para a melhoria de sistemas de detecção de vazamentos por *software* é incluir nas análises feitas pelo sistema o mesmo tipo de análise, normalmente de natureza heurística, realizada pelos controladores do duto. Isto pode apresentar as seguintes vantagens:

- Diminuição da carga de trabalho dos controladores e conseqüentemente liberação de tempo dos controladores para outras tarefas;
- Melhorar a resposta da organização a situações de falha do gasoduto, uma vez que o sistema pode tornar disponíveis rapidamente várias informações úteis aos controladores;
- Identificar problemas com o SCADA e/ou instrumentação que de outro modo seriam difíceis de perceber.

Para testar esta possibilidade, foi realizada a implementação de um protótipo de sistema de detecção de vazamentos baseado em conhecimento para a empresa que controla o gasoduto Bolívia-Brasil. Este protótipo toma como entrada os dados provenientes do sistema de detecção de vazamentos já instalado na empresa, junto com dados obtidos do banco de dados do SCADA e retorna as situações pelas quais o gasoduto está passando naquele momento.

Como resultado do desenvolvimento do protótipo, foi possível chegar a algumas conclusões com relação a aplicabilidade desta técnica. A principal é que o desenvolvimento de um sistema deste tipo é extremamente dependente da empresa que irá utilizar o sistema, principalmente com relação aos seguintes aspectos:

- A empresa precisa estar consciente das dificuldades do desenvolvimento de um sistema especialista e precisa saber que, além da liberação de recursos financeiros, um sistema especialista depende da interação com especialistas da empresa, e portanto da liberação também de recursos humanos;
- O sistema desenvolvido tende a ser específico para a situação presente na empresa no momento do desenvolvimento. Isto pode causar problemas caso, por exemplo, o sistema de detecção de vazamentos tradicional utilizado pela empresa seja trocado;
- Pode ser difícil acrescentar novos conhecimentos ao sistema, caso seja necessário. Isto porque esta tarefa deverá ser feita por um engenheiro de conhecimento, e mesmo assim, de preferência um que tenha um bom conhecimento do sistema. Isto pode tornar dispendiosa a manutenção do sistema. Este problema será pior caso seja necessário acrescentar novas regras difusas ou alterar as análises dos sinais do SCADA.

Ainda com relação ao processo de desenvolvimento do sistema, é preciso notar que podem surgir problemas de interação com a empresa durante o desenvolvimento do sistema porque empresas de petróleo e gás natural normalmente não têm uma cultura de desenvolvimento de *software*, e não estão habituadas com as particularidades do gerenciamento de projetos desse tipo.

Essas dificuldades se evidenciaram principalmente durante a etapa de definição de requisitos. Esta tarefa é complexa e exige um bom entendimento por parte da empresa de suas próprias expectativas com relação ao sistema. Um dos grandes problemas encontrados durante o desenvolvimento do protótipo foi a indefinição com relação aos requisitos. Considera-se que, caso um maior esforço para evitar essa indefinição houvesse sido feito, tanto da parte da equipe de desenvolvimento quanto da parte da empresa-alvo, o projeto teria sido mais bem sucedido.

Uma outra questão associada ao relacionamento entre a empresa e a equipe de desenvolvimento é a escolha das técnicas utilizadas na implementação do sistema. No caso do protótipo desenvolvido, se estudou preliminarmente o uso de técnicas estatísticas e redes neurais como parte do sistema. No entanto, estas possibilidades foram abandonadas por necessitarem de um volume de dados muito grande, e não se identificou uma disposição da empresa-alvo em fornecer a tempo os dados necessários. Também os próprios representantes da empresa não sabiam ao certo exatamente quais tipos de informação estavam disponíveis, o que influenciou na decisão de abandonar o estudo destas técnicas.

Com relação à utilidade das análises realizadas pelo sistema, como foi visto anteriormente, não foi possível realizar a validação do sistema pelos especialistas da empresa. Também, devido ao pequeno número de entrevistas de aquisição de conhecimento realizadas, não foi possível implementar nenhuma análise mais complexa sobre o funcionamento do duto. No entanto, segundo (KERKHOF et al., 2002), onde está descrita uma implementação de um sistema com o mesmo escopo do proposto aqui, a implementação de um sistema desse tipo tornou acessível aos operadores uma série de resultados que se mostraram vantajosos.

Finalmente, não foi possível estabelecer se o sistema é viável economicamente, ou seja, que o ganho obtido pela empresa com um sistema desse tipo compensa os gastos com o desenvolvimento. Apesar de não ter sido realizada nenhuma análise sobre esse aspecto, considera-se preliminarmente que o sistema é vantajoso na medida em que diminui o risco de acontecer um acidente, ou que pode tornar a reação a um acidente mais rápida. Dado o tamanho das perdas financeiras associadas a eventos desse tipo é bem possível que o sistema seja interessante do ponto de vista financeiro.

5.1 Recomendações para trabalhos futuros

Como recomendações para trabalhos futuros, o sistema implementado deixa várias questões em aberto. A primeira delas está relacionada com a eficácia do método de explicação proposto na seção 4.5. Seria preciso fazer testes para determinar se as soluções produzidas por uma implementação deste método seriam aceitáveis do ponto de vista de um controlador do gasoduto.

Também seria interessante testar a habilidade do sistema de lógica difusa implementado de lidar com um conjunto bem maior de análises. Questões de eficiência, facilidade de representação de conhecimento e facilidade de manutenção do sistema que apareceriam neste caso seriam interessantes de serem analisadas. Também é necessário realizar uma validação do sistema que

foi implementado para verificar se os resultados produzidos atualmente pelo sistema estão corretos. Também é preciso verificar a adequação da interface gráfica desenvolvida às necessidades dos controladores.

Uma outra questão que vale a pena analisar está relacionada com o problema da viabilidade do sistema, como foi discutido anteriormente. Seria preciso avaliar os custos envolvidos em um acidente e a capacidade do sistema de reduzir a probabilidade desses acidentes e a eficácia do sistema em diminuir as conseqüências de um acidente.

Finalmente, considerou-se que o uso dos diagramas de influência foi uma ferramenta crucial no desenvolvimento do sistema especialista. O estudo de uma linguagem de representação para esses diagramas que fosse capaz de expressar mais aspectos da base de conhecimento, e que não fosse dependente de uma representação específica provavelmente representaria um ganho em produtividade no desenvolvimento deste tipo de sistema.

Referências

- ALONSO, F. et al. An incremental solution for developing knowledge-based software: its application to an expert system for isokinetics interpretation. *Expert systems with Applications*, v. 18, n. 3, p. 165–184, apr 2000.
- ANP. *Glossário*. 2006. Disponível em: <<http://www.anp.gov.br/glossario/index.asp>>. Acesso em: 2006-04-04.
- BELSITO, S. et al. Leak detection in liquefied gas pipelines by artificial neural networks. *AIChE Journal*, v. 44, n. 12, p. 2675 – 2688, dec 1998.
- BENKHEROUF, A.; ALLIDINA, A. Y. Leak detection and location in gas pipelines. *IEEE Proceedings in Control Theory and Applications*, v. 135, n. 2, p. 142–148, mar 1988.
- BIAGETTI, T.; SCIUBBA, E. Automatic diagnostics and prognostics of energy conversion processes via knowledge-based systems. *Energy*, v. 29, n. 12-15, p. 2553–2572, 2004.
- BILLMAN, L.; ISERMANN, R. Leak detection methods for pipelines. *Automatica*, v. 23, n. 3, p. 381 – 385, may 1987.
- BITTENCOURT, G. *Inteligência Artificial - Ferramentas e teorias*. Florianópolis: Editora da UFSC, 2001. ISBN 85-328-0138-2.
- BORGES, J. L. El idioma analítico de John Wilkins. In: _____. *Otras Inquisiciones*. 3. ed. [S.l.]: Alianza Editorial, 1981. ISBN 84-206-1604-4.
- BRODETSKY, I.; SAVIC, M. Leak monitoring system for gas pipelines. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993. ICASSP-93*. [S.l.: s.n.], 1993. v. 3, p. 17–20.
- CASTELANI, M. R.; GALAZ, L. A.; SILVA, J. C. *Sistema especialista para gerenciamento operacional de redes de distribuição de gás natural*. 2002. COCIM.
- CHANDRASEKARAN, B.; JOSEPHSON, J. R.; BENJAMINS, V. R. What are ontologies, and why do we need them? *IEEE Intelligent Systems and Their Applications*, v. 14, n. 1, p. 20–26, feb 1999. ISSN 1094-7167.
- CLANCEY, W. N. Heuristic classification. *Artificial Intelligence*, v. 27, p. 289–350, 1985.
- CLIPS Reference Manual – Basic Programming Guide. [S.l.], jun 2005. Disponível em: <<http://www.ghg.net/clips/download/documentation/bpg.pdf>>. Acesso em: 2006-04-19.
- CODÓN, O.; JESUS, M. J.; HERRERA, F. A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate Reasoning*, v. 20, n. 1, p. 21–45, jan 1999.

- CONDACSE, V.; GUIDRY, B. *Practical Implementation Of A Leak Detection System In A Transient Thermal Environment*. sep 2002. PSIG – Pipeline Simulation Interest Group.
- COX, E. *The fuzzy systems handbook*. 2. ed. San Diego: Academic Press, 1998. ISBN 0-12-194456-5.
- DUBOIS, D.; PRADE, H. *Fuzzy Sets and Systems: Theory and Applications*. New York: Academic Press, 1980. ISBN 0-12-222750-6.
- DUBOIS, D.; PRADE, H. What are fuzzy rules and how to use them. *Fuzzy Sets and Systems*, v. 84, p. 169–185, 1996.
- DYM, C. L.; LEVITT, R. E. *Knowledge-based systems in engineering*. [S.l.]: McGraw-Hill, 1991. ISBN 0-07-018563-8.
- EBRAHIM, R. Fuzzy logic programming. *Fuzzy Sets and Systems*, v. 117, n. 2, p. 215–230, jan 2001.
- ENERGY SOLUTIONS INTERNATIONAL. *Best practices in leak detection*. mar 2004. White Paper.
- EUA, G. dos. *Code of Federal Regulations, Title 49, Part 192*. oct 2005. Disponível em: <http://www.access.gpo.gov/nara/cfr/waisidx_02/49cfr192_02.html>. Acesso em: 2006-04-05.
- FAN, J.-L.; MA, Y.-L. Some new fuzzy entropy formulas. *Fuzzy Sets and Systems*, Elsevier North-Holland, Inc., v. 128, n. 2, p. 277–284, jun 2002. ISSN 0165-0114.
- FORGY, C. L. Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence*, v. 19, p. 17–37, 1982.
- FUKUNAGA, K. *Introduction to Statistical Pattern Recognition*. 2. ed. San Diego: Morgan Kaufman, 1990. ISBN 0-12-269851-7.
- FUKUSHIMA, K. et al. Gas pipeline leak detection system using the online simulation method. *Computers & Chemical Engineering*, v. 24, n. 2-7, p. 453–456, jul 2000.
- GIARRATANO, J.; RILEY, G. *Sistemas expertos – Principios y programación*. 1. ed. México: International Thomson Editores, 1998. ISBN 970-686-059-2.
- GOODREAU, M.; BACHMAN, S. *Less is More: Accuracy versus Precision in Modeling*. oct 2000. PSIG – Pipeline Simulation Interest Group.
- GRAY, N. A. B. *Programming with Class*. 1. ed. Chichester: John Wiley & Sons, 1994. ISBN 0-471-94350-9.
- HALL, L. O. Rule chaining in fuzzy expert systems. *IEEE Transactions on Fuzzy Systems*, v. 9, n. 6, p. 822–828, dec 2001.
- HART, A. *Knowledge acquisition for expert systems*. 2. ed. New York: McGraw-Hill, 1992. ISBN 0-07-026911-4.
- HOFSTADTER, D. R. *Gödel, Escher, Bach: An Eternal Golden Braid*. [S.l.]: Basic Books, 1979. ISBN 0-465-02656-7.

- IOURTCHENKO, S.; LOSENKOV, A. *Detection of leaks artificially made in pipelines (by third party intervention)*. oct 2002. PSIG – Pipeline Simulation Interest Group.
- JOHNSON, A. T. *A Practical Approach to the Application of an Expert System to Gas Pipeline Operation and Data Integrity*. oct 1999. PSIG – Pipeline Simulation Interest Group.
- KANDEL, A. (Ed.). *Fuzzy mathematical techniques with applications*. [S.l.]: Addison-Wesley Publishing Company, Inc., 1986. ISBN 0-201-11752-5.
- KAUFMANN, A. *Introduction to the Theory of Fuzzy Subsets*. [S.l.]: Academic Press, Inc., 1975. ISBN 0-12-402301-0.
- KERKHOF, J. et al. Transcanada pipelines: an expert advisory system for pipeline operations. In: *Proceedings of OPC'02*. [S.l.: s.n.], 2002. 4th International Pipeline Conference.
- KLIR, G. J.; YUAN, B. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. 1. ed. [S.l.]: Prentice Hall PTR, 1995. ISBN 0-13-101171-5.
- KONVALINKA, I. et al. Decision trees development for leak detection on gas transmission system using stationary model and machine learning from examples. In: *First International Conference on Intelligent Systems Engineering*. [S.l.: s.n.], 1992. p. 330–335.
- KOOP, G. *Bayesian Econometrics*. Chichester: Wiley, 2003. ISBN 0-470-84567-8.
- LARMAN, C. *Applying UML and patterns*. Upper Saddle River: Prentice Hall PTR, 1998. ISBN 0-13-748880-7.
- LEHMANN, F. Semantic networks. *Computers Math. Applic.*, v. 23, n. 2-5, p. 1–50, 1992.
- LOPARO, K. A.; BUCHNER, M. R.; VASUDEVA, K. S. Leak detection in an experimental heat exchanger process: a multiple model approach. *IEEE Transactions on Automatic Control*, v. 36, n. 2, p. 167–177, feb 1991.
- MATKO, D.; GEIGER, G.; WERNER, T. *Leak detection and Localisation - A Survey*. oct 2003. PSIG – Pipeline Simulation Interest Group.
- MINSKY, M. A framework for representing knowledge. In: _____. *The Psychology of Computer Vision*. [S.l.]: McGraw-Hill, 1975.
- MORENTE, M. G. *Fundamentos de filosofia – lições preliminares*. 3. ed. [S.l.]: Editora Mestre Jou, 1967.
- NARDI, D.; BRACHMANN, R. J. An introduction to description logics. In: _____. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003. p. 5–44. Disponível em: <<http://www.sfu.ca/~mhatala/iat881/papers-/nardi-dlhb-01.pdf>>. Acesso em: 2006-04-13.
- NEAPOLITAN, R. E. *Probabilistic reasoning in expert systems*. 1. ed. [S.l.]: John Wiley & Sons, Inc., 1990. ISBN 0-471-61840-3.
- NICHOLAS, R. E.; REET, J. V.; WHALEY, D. R. S. *A Tutorial on Computer Based Leak Detection Methods*. 1992. PSIG – Pipeline Simulation Interest Group.

- PAPAMICHAIL, K. N.; FRENCH, S. Explaining and justifying the advice of a decision support system: a natural language generation approach. *Expert Systems with Applications*, v. 24, n. 1, p. 35–48, jan 2003.
- PEARL, J. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, v. 29, p. 241–288, 1986.
- PIERCE, C. S. *Semiótica*. 3. ed. São Paulo: Perspectiva, 1999. (Estudos). ISBN 8527301946.
- PIETSCH, F.; PETHERICK, L. M. *Effects of Errors In Linepack Calculations On Real-Time Computational Pipeline Monitoring*. sep 1994. PSIG – Pipeline Simulation Interest Group.
- POLLOCK, D. S. G. *A handbook of time-series analysis, signal processing and dynamics*. [S.l.]: Academic Press, 1999. (Signal Processing and its Applications). ISBN 0-12-560990-6.
- PRESS, W. H. et al. *Numerical Recipes in C: The Art of Scientific Computing*. 2. ed. New York, USA: Cambridge University Press, 2002. ISBN 0521437148. Disponível em: <<http://www.library.cornell.edu/nr/bookcpdf.html>>. Acesso em: 2006-04-19.
- RUSSELL, S.; NORVIG, P. *Inteligência Artificial*. 2. ed. [S.l.]: Elsevier, 2004. ISBN 85-352-1177-2.
- SALAS, I.; REID, F. *Evaluation Of Cost-Effective Rupture Detection*. 2002. PSIG – Pipeline Simulation Interest Group.
- SAMPATH, S.; YEE, S. *Automating the Predictor to Answer Routine Operational Questions*. oct 2000. PSIG – Pipeline Simulation Interest Group.
- SCHREIBER, G. et al. *Knowledge Engineering and Management – The CommonKADS Methodology*. [S.l.: s.n.], 1999.
- SCOTT, A. C.; CLAYTON, J. E.; GIBSON, E. L. *A practical guide to knowledge acquisition*. [S.l.]: Addison-Wesley Publishing Co., Inc., 1991. ISBN 0-201-14597-9.
- ŠENDELJ, R.; DEVEDŽIĆ, V. Fuzzy systems based on component software. *Fuzzy Sets and Systems*, v. 141, p. 487–504, 2004.
- SILVA, J. C. et al. *Sistema especialista para gerenciamento de redes de gás natural – SEGRED*. 2004. Rio Oil & Gas 2004.
- SILVA, J. C.; PORCIÚNCULA, G. S. *Sistema especialista para gerenciamento de redes de transporte de gás natural*. out 2003. Rio Pipeline Conference & Exposition.
- SILVA JR., A. C.; SILVA, J. C. *Integração entre sistemas especialistas e simulação para o monitoramento de redes de transporte de gás natural*. ago 2002. Congresso Nacional de Engenharia Mecânica, CONEM.
- SOMMERVILLE, I. *Software engineering*. [S.l.]: Addison Wesley Longman Limited, 1995. ISBN 0-201-42765-6.
- SONG, S.; LEE, S. A strategy of dynamic reasoning in knowledge-based system with fuzzy production rules. *Journal of Intelligent Information Systems*, v. 19, n. 3, p. 303–318, 2002.

- STROUSTRUP, B. *The C++ Programming Language*. 3. ed. [S.l.]: Addison-Wesley, 1997. ISBN 0-201-88954-4.
- STUDER, R.; BENJAMINS, V. R.; FENSEL, D. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, v. 25, n. 1-2, p. 161–197, mar 1998.
- SWINNEY, L. The explanation facility and the explanation effect. *Expert systems with Applications*, v. 9, n. 4, p. 557–567, 1995.
- TORRA, V.; SODAN, A. A multi-stage system in compilation environments. *Fuzzy Sets and Systems*, v. 105, n. 1, p. 49–61, jul 1999.
- TURNER, W. J. Better leak detection in gas pipelines. *Comput. Math. Applic.*, v. 15, n. 1, p. 69–75, 1988.
- TURNER, W. J.; MUDFORD, N. R. Leak detection, timing, location and sizing in gas pipelines. *Mathematical and Computer Modelling*, v. 10, n. 8, p. 609–627, 1988.
- URAIKUL, V.; CHAN, C. W.; TONTIWACHWUTHIKUL, P. Development of an expert system for optimizing natural gas pipeline operations. *Expert Systems with Applications*, v. 18, p. 271–282, 2000.
- VALERIANO, A. *PMS – Guia Rápido de Utilização*. [S.l.], apr 2001. Documento de circulação restrita à TBG.
- VOJTÁŠ, P. Fuzzy logic programming. *Fuzzy Sets and Systems*, v. 124, n. 3, p. 361–370, dec 2001.
- VRANEŠ, S.; STANOJEVIĆ, M. Integrating multiple paradigms within the blackboard framework. *IEEE Transactions on software engineering*, v. 21, n. 3, p. 244–262, mar 1995.
- WALLOOPPILLAI, R. K. et al. *Integrating an Expert System and Pipeline Simulator to Enhance Gas Pipeline Operation, Profitability and Safety*. oct 2000. PSIG – Pipeline Simulation Interest Group.
- WANG, G.; DONG, D.; FANG, C. Leak detection for transport pipelines based on autoregressive modeling. *IEEE Transactions on Instrumentation and Measurement*, v. 42, n. 1, p. 68–71, feb 1993.
- WESTHOFF, M. A. Using operating data at natural gas pipelines. In: NATIONAL TRANSPORTATION SAFETY BOARD. *International Symposium on Transportation Recorders*. [S.l.], 1999.
- WICK, M.; SLAGLE, J. An explanation facility for today's expert systems. *IEEE Expert*, v. 4, n. 1, p. 26–36, 1989. ISSN 0885-9000.
- ZADEH, L. A. Fuzzy sets. *Information and Control*, v. 8, p. 338–353, 1965.
- ZHANG, J. Designing a cost effective and reliable pipeline leak detection system. *Pipes & Pipelines International*, p. 20–26, jan 1997.

APÊNDICE A

Exemplos de uso do sistema

Nesta seção são mostradas algumas telas do sistema, e o modo de usá-lo. O programa possui apenas uma tela, que permite a entrada dos valores dos filtros do PMS. Com os valores ajustados, ao se pressionar o botão “Analisar”, no canto inferior direito da tela, os resultados da análise aparecerão na área de texto central. As figuras A.1 a A.4 mostram exemplos de uso do sistema.

SDV-BC: Sistema de detecção de vazamento

Intruções para o uso do protótipo de detecção de vazamentos baseada em conhecimento do SEGRED Rede

Este sistema é um protótipo para testar as heurísticas implementadas até agora no sistema. Ele recebe como entradas os valores que o sistema de detecção de vazamentos instalado na TBG (SDV-TBG) fornece como saída e analisa esses valores.

Como usar

Coloque os valores dos filtros no quadro ao lado. Caso algum dos filtros for maior do que três, a caixa para a entrada dos dados do posicionador de vazamentos será ativada, e é possível colocar a posição de vazamento que o SDV-TBG determinou.

Para entender o que significa cada um dos tipos de filtro, [clique aqui](#).

Detalhes

Posição Vazamento: Km

Analisar

	ARAUCARIA	JOINVILLE	GUARAMIRIM	GASPAR	BRUSQUE	TIJUCAS	BIGUAÇU	
Pressão Jusante	00	00	00	00	00	00	00	[kgf/cm ²]
Pressão Montante	00	00	00	00	00	00	00	[kgf/cm ²]
Vazão	00	00	00	00	00	00	00	[m ³ /dia]

Figura A.1: Tela inicial do sistema de detecção de vazamentos. Na caixa de texto central é mostrada uma pequena introdução sobre o programa. Quando alguma análise for executada o resultado aparecerá nesta caixa. À esquerda estão as caixas de entrada dos valores dos filtros do PMS. No centro, abaixo da caixa de texto com o texto introdutório, encontra-se a caixa de entrada para a posição do vazamento.

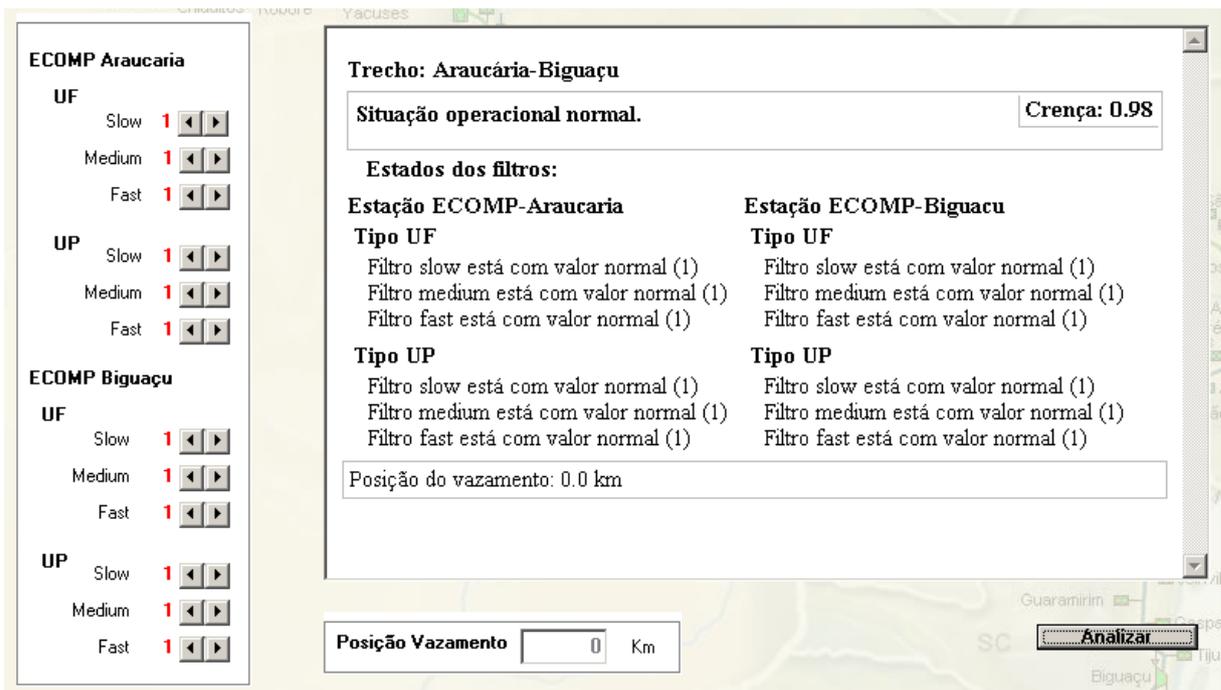


Figura A.2: Exemplo de análise do detector de vazamentos. Nesta análise, todos os filtros foram colocados na posição normal. Observe que o único resultado obtido foi uma situação normal. Neste exemplo, a crença no resultado não é 1 porque alguma das análises dos sensores deve estar indicando alguma variação nos valores.

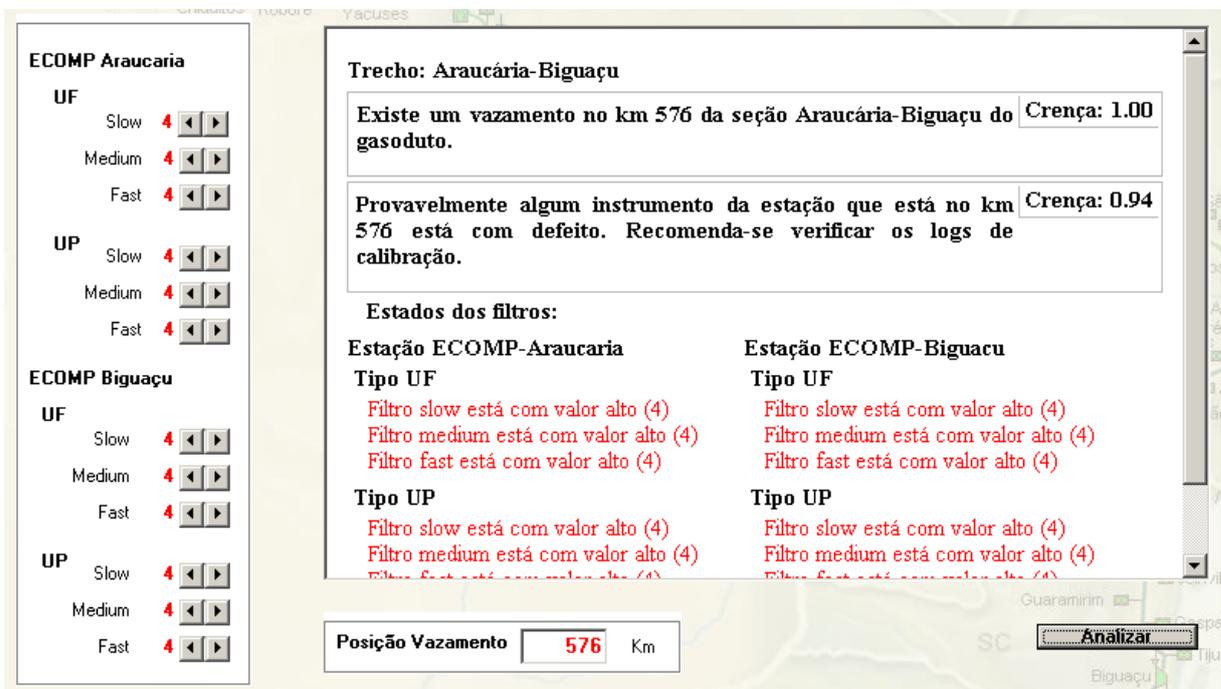


Figura A.3: Neste exemplo, os filtros foram colocados todos na posição indicativa de vazamento, e a posição do vazamento foi colocada sobre a estação de Joinville. Observa-se que o sistema retorna como mais provável a hipótese de vazamento no local, e, em segundo lugar, a hipótese de algum problema em algum instrumento da estação.

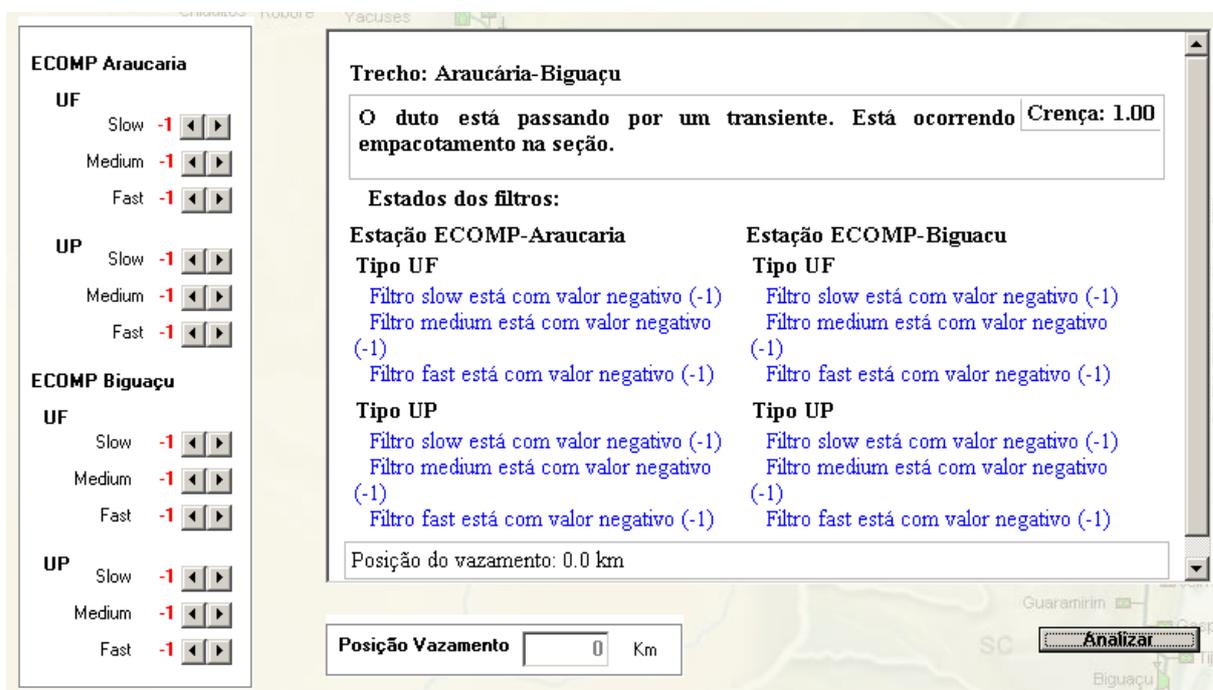


Figura A.4: Neste exemplo, os filtros foram colocados todos na posição indicativa de empacotamento. O sistema retorna uma resposta de acordo com a situação.

APÊNDICE B

Vocabulário técnico sobre gasodutos

A ANP (2006) estabelece um glossário técnico sobre petróleo e gás natural. Os termos deste glossário relevantes a este trabalho foram repetidos aqui para referência:

Campo de Petróleo ou de Gás Natural Área produtora de petróleo ou gás natural, a partir de um reservatório contínuo ou de mais de um reservatório, a profundidades variáveis, abrangendo instalações e equipamentos destinados à produção.

Capacidade Nominal Capacidade de processamento para a qual uma planta industrial é projetada.

City Gate Conjunto de instalações contendo *manifolds* e sistema de medição, destinado a entregar o gás natural (oriundo de uma concessão, de uma UPGN, de um sistema de transporte ou de um sistema de transferência de Custódia de Gás Natural).

Condensado Frações líquidas do gás natural obtidas no processo de separação normal de campo, mantidas na fase líquida nas condições normais de pressão e temperatura.

Derivados de Petróleo Produtos decorrentes da separação física ou da transformação química do petróleo.

Derivados Energéticos de Petróleo Derivados de petróleo utilizados predominantemente como combustíveis, isto é, com a finalidade de liberar energia, luz ou ambos a partir de sua queima. Esta denominação abrange os seguintes derivados: GLP, gasolina A, gasolina de aviação, querosene iluminante, querosene de aviação, óleo diesel e óleo combustível.

Derivados Não-Energéticos de Petróleo Derivados de petróleo que, embora tenham significativo conteúdo energético, são utilizados para fins não-energéticos. Esta denominação abrange os seguintes derivados: graxas, lubrificantes, parafinas, asfaltos, solventes,

coque, nafta, extrato aromático, gasóleo de vácuo, óleo leve de reciclo, diluentes, n-parafinas, outros óleos de petróleo, minerais betuminosos, bem como outros produtos de menor importância.

Distribuição Atividade de comercialização por atacado com a rede varejista ou com grandes consumidoras de combustíveis, lubrificantes, asfaltos e gás liquefeito envasado, exercida por empresas especializadas, na forma das leis e regulamentos aplicáveis. A atividade de distribuição compreende a aquisição, armazenamento, transporte, comercialização e o controle de qualidade dos combustíveis líquidos derivados de petróleo, álcool combustível e outros combustíveis automotivos.

Distribuidora Agente cuja atividade caracteriza-se pela aquisição de produtos a granel e sua revenda a granel (por atacado) para a rede varejista ou grandes consumidores (vide Distribuição).

Duto Designação genérica de instalação constituída por tubos ligados entre si, destinada à movimentação de petróleo, seus derivados e gás natural. Movimenta produtos líquidos (oleoduto) e gasosos (gasoduto). Pode ser classificado como duto de Transporte ou de Transferência.

Gás Canalizado Gás produzido a partir da nafta, consumido predominantemente pelo setor residencial. É distribuído nos centros urbanos, através das redes de distribuição das companhias estaduais de gás.

Gás de Refinaria Mistura contendo principalmente hidrocarbonetos gasosos (além de, em muitos casos, alguns compostos sulfurosos) produzida no craqueamento e refino do petróleo. Os componentes mais comuns são hidrogênio, metano, etano, propano, butanos, pentanos, etileno, propileno, butenos, pentenos e pequenas quantidades de outros componentes, como o butadieno. É utilizado principalmente como matéria-prima na fabricação de produtos petroquímicos, na produção de gasolina de alta octanagem e na síntese orgânica de alcóois.

Gás Liquefeito de Petróleo (GLP) Mistura de hidrocarbonetos líquidos obtidos do gás natural através do processamento de gás natural nas UPGNs, ou em processo convencional nas refinarias de petróleo. Conhecido como gás de cozinha, composto de propano e butano. Sua maior aplicação é na cocção dos alimentos. também é utilizado em empilhadeiras, soldagem, esterilização industrial, teste de fogões, maçaricos e outras aplicações industriais.

Gás Natural Associado Gás natural produzido de jazida onde ele é encontrado dissolvido no petróleo ou em contato com petróleo subjacente saturado de gás.

Gás Natural Comprimido (GNC) Todo gás natural processado e condicionado para o transporte em ampolas ou cilindros, à temperatura ambiente e pressão próxima à condição de mínimo fator de compressibilidade.

Gás Natural Liquefeito (GNL) Gás natural resfriado a temperaturas inferiores a -160°C para fins de transferência e estocagem como líquido. É composto predominantemente de metano e pode conter quantidades mínimas de etano, propano, nitrogênio ou outros componentes normalmente encontrados no gás natural.

Gás Natural Não-Associado Gás natural produzido de jazida de gás seco ou de jazida de gás e condensado.

Gás Natural ou Gás Todo hidrocarboneto ou mistura de hidrocarbonetos que permaneça em estado gasoso ou dissolvido no óleo nas condições originais do reservatório, e que se mantenha no estado gasoso nas condições atmosféricas normais. É extraído diretamente de reservatórios petrolíferos ou gaseíferos, incluindo gases úmidos, secos, residuais e gases raros (gases nobres). Ao processar o gás natural úmido nas UPGNs, são obtidos os seguintes produtos: (i) o gás seco (também conhecido como gás residual); e (ii) o líquido de gás natural (LGN), que contém propano (C_3) e butano (C_4) (que formam o gás liquefeito de petróleo - GLP) e a gasolina natural (C_{5+}).

Gás Natural Veicular (GNV) Mistura combustível gasosa, tipicamente proveniente do gás natural e biogás, destinada ao uso veicular e cujo componente principal é o metano, observadas as especificações estabelecidas pela ANP.

Gás Seco Produto do processamento do gás úmido, o qual não contém líquidos comercialmente recuperáveis (LGN).

Gás Úmido Gás natural que entra nas UPGNs contendo hidrocarbonetos pesados e comercialmente recuperáveis sob a forma líquida (LGN).

Indústria de Petróleo Conjunto de atividades econômicas relacionadas com a exploração, desenvolvimento, produção, refino, processamento, transporte, importação e exportação de petróleo, gás natural, outros hidrocarbonetos aromáticos (metano, etano, propano, etileno, propileno, butenos, ciclohexanos, benzeno, tolueno, naftaleno etc) e seus derivados.

Transferência Movimentação de petróleo, seus derivados ou gás natural em meio ou percurso considerado de interesse específico e exclusivo do proprietário ou explorador das facilidades.

Transportador Pessoa jurídica operadora das Instalações de Transporte.

Transporte Movimentação de petróleo, seus derivados ou gás natural em meio ou percurso considerado de interesse geral.

Unidade de Processamento de Gás Natural (UPGN) Instalação industrial que objetiva realizar a separação das frações pesadas (propano e mais pesados), existentes no gás natural, do metano e do etano, gerando GLP e gasolina natural (C₅₊).