

Reginaldo Inojosa da Silva Filho

**Um modelo para sistemas complexos diagnosticáveis com
estudo de caso para diagnóstico em dispensadores
automáticos de cédulas.**

**São Paulo
2006**

Reginaldo Inojosa da Silva Filho

**Um modelo para sistemas complexos diagnosticáveis com
estudo de caso para diagnóstico em dispensadores
automáticos de cédulas.**

**Dissertação apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção do título de
Mestre.**

**Orientador
Prof. Dr. Zsolt Laszlo Kovacs**

**São Paulo
2006**

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, de julho de 2006.

Assinatura do autor _____

Assinatura do orientador _____

FICHA CATALOGRÁFICA

Silva Filho, Reginaldo Inojosa

Um modelo para sistemas complexos diagnosticáveis com estudo de caso para diagnóstico em dispensadores automáticos de cédulas / R.I. Silva Filho. -- ed.rev. -- São Paulo, 2006.

p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Sistemas Eletrônicos.

1.Falhas computacionais (Diagnóstico; Predição) 2.Inteligência artificial I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Sistemas Eletrônicos II.t.

Agradecimentos

Agradeço primeiramente a **Deus**, por me dar saúde e força para prosseguir e por iluminar meus caminhos, permitindo-me chegar até o final de mais essa etapa. Agradeço a meu orientador, Professor Doutor **Zsolt Laszlo Kovacs**, pelo incentivo e orientação dados ao longo deste mestrado. Agradeço também a minha querida esposa **Patrícia** e meu filho **Christian** pela paciência e ajuda durante todo esse tempo e durante todas as horas junto a eles que sacrifiquei em prol da conclusão desse trabalho, a meu pai **Reginaldo** e minha irmã **Patrícia Cristina** pelo apoio e pela confiança.

Não poderia deixar de agradecer também ao Doutor **João Kogler** e aos Professores Doutores **Henrique Schützer Del Nero** e **Marcio Lobo Netto**, pelas discussões que trouxeram muitas idéias e esclarecimentos ao meu trabalho. Agradeço também aos colegas engenheiros **Mariana Antônia Aguiar**, **Valnir Rogerio Trava Airoidi** e **Alex Lopes** pelas conversas acerca do tema dessa dissertação e pelas dicas e sugestões dadas.

Quero agradecer também à Diebold Procomp e à Atech, bem como aos Professores que fizeram parte de minha banca de qualificação, Prof. Doutor **João Batista Camargo Junior** e Prof. Doutor **Jorge Luis Risco Becerra** pelas correções de rumo e pelas observações sempre muito pertinentes.

Em memória de minha amada mãe **Carmem** (1933 - 2004).

Sumário

Agradecimentos	4
Sumário	5
Índice de Figuras	8
Índice de Tabelas	9
Glosário de Termos	10
Resumo	11
Abstract	12
1 Introdução	13
1.1 <i>Organização do trabalho</i>	14
2 Introdução ao conceito de diagnóstico	16
3 Analogia médica: o raciocínio clínico.....	18
3.1 <i>As teorias médicas: explicando as doenças</i>	18
3.2 <i>Os esquemas explanatórios</i>	19
3.3 <i>A relação entre os esquemas explanatórios e o diagnóstico clínico</i>	22
3.4 <i>As classes de conhecimento no raciocínio clínico</i>	23
3.5 <i>O processo de diagnóstico clínico e suas etapas</i>	24
3.5.1 <i>Formulação de um conjunto de hipóteses</i>	26
3.5.2 <i>O Teste das hipóteses</i>	26
3.5.3 <i>Rejeição de uma hipótese</i>	27
4 O Diagnóstico de falhas	29
5 A formalização da teoria diagnóstica	34
6 O Raciocínio baseado em modelos (RBM).....	35
6.1 <i>Descrição de um exemplo de ambiente de teste para a aplicação e comparação entre os modelos das comunidades DX e FDI</i>	39
6.2 <i>A Comunidade FDI</i>	41
6.2.1 <i>Aplicação da abordagem FDI no exemplo do polybox</i>	43
6.3 <i>A abordagem DX</i>	51

6.3.1	Aplicação da abordagem DX no exemplo do polybox	52
7	O Raciocínio Baseado em Casos (RBC)	61
7.1	<i>A etapa de recuperação.....</i>	62
7.1.1	Definição de caso.....	63
7.1.2	Definição de similaridade	64
7.2	<i>A etapa de reutilização.....</i>	68
7.3	<i>As etapas de revisão e retenção</i>	69
7.4	<i>Um exemplo de aplicação da abordagem RBC para diagnóstico.....</i>	70
7.5	<i>Comparação entre a opção do RBC para o diagnóstico em relação às outras abordagens para diagnóstico de falhas</i>	71
8	O Troubleshooting	72
9	Outras abordagens: O Diagnóstico baseado em testes de hipóteses e o modelo PMC	74
9.1.1	Detecção de sintomas	75
9.1.2	Geração de hipóteses	76
9.1.3	Discriminação de uma hipótese	76
	<i>O Modelo PMC.....</i>	77
10	Sistemas automáticos de diagnóstico.....	78
10.1	<i>Rapidez na detecção e no diagnóstico.....</i>	79
10.2	<i>Isolabilidade.....</i>	80
10.3	<i>Robustez</i>	80
10.4	<i>Capacidade de identificação para situações novas.....</i>	81
10.5	<i>Transparência nos resultados</i>	81
10.6	<i>Adaptabilidade</i>	81
10.7	<i>Clareza na explicação dos resultados.....</i>	82
11	A importância do diagnóstico de falhas em Engenharia	83
12	Os Sistemas complexos diagnosticáveis (SCD)	84
12.1	<i>Definindo os Sistemas complexos diagnosticáveis.....</i>	85
12.2	<i>As Conseqüências do modelo: o enunciado do problema diagnóstico.....</i>	89
12.3	<i>As diferentes Classes de diagnóstico.....</i>	91
12.3.1	Quanto à aquisição de dados: <i>troubleshooting</i> e diagnóstico auxiliado por sensores.	91

12.3.2	Quanto à interpretação: diagnóstico preditivo e diagnóstico corretivo	92
13	Estudo de Caso	93
13.1	<i>O Histórico do projeto SRD</i>	94
13.2	<i>A Arquitetura Global da Solução</i>	95
13.3	<i>Análise do S-Paciente: O Dispensador de Numerário</i>	96
13.3.1	<i>O presenter e a placa controladora</i>	98
13.4	<i>A aplicação do modelo SCD para o diagnóstico do MMD</i>	99
13.5	<i>O Histórico</i>	100
13.6	<i>A Sintaxe do arquivo de parametrização</i>	101
13.6.1	<i>Log</i>	101
13.6.2	<i>Variáveis</i>	102
13.6.3	<i>Sintomas</i>	103
13.6.4	<i>Diagnóstico</i>	105
13.7	<i>A Sintaxe das Mensurações implementadas</i>	106
13.8	<i>A Sintaxe do Arquivo de Mensagens</i>	110
13.9	<i>O Processo de aquisição do conhecimento</i>	112
13.10	<i>Validação</i>	116
14	Conclusões	117
15	Referências bibliográficas	118

Índice de Figuras

Figura 1 - O processo de tratamento clínico.....	16
Figura 2 - Diagrama de estrutura causal.....	20
Figura 3 - Exmplo de diagrama de estrutura causal para a teoria microbiana das doenças	21
Figura 4 - Estado decisório antes do diagnóstico: múltiplas causas possíveis.....	25
Figura 5 - Fluxograma do processo de diagnóstico.....	27
Figura 6 - O <i>Polybox</i>	39
Figura 7 - Matriz de <i>Matching</i>	46
Figura 8 - As relações de redundância	47
Figura 9 - O Ciclo RBC.....	62
Figura 10 - Um exemplo de formato de Caso	70
Figura 11 - organização hierárquica da biblioteca de casos	70
Figura 12 - Exemplo de diagrama de estratégia.	73
Figura 13 - Hierarquia das tarefas em sistemas baseado em teste de hipótese.....	75
Figura 14 - Os três estados possíveis de um Sistema no estudo do diagnóstico de falhas.....	78

Índice de Tabelas

Tabela 1 - Representação tabular da matriz de <i>SM-incidência</i>	46
Tabela 2 – Possíveis defeitos para o Sistema Diagnosticável trem de pouso.....	72
Tabela 3 – Ações e perguntas para o problema do trem de pouso	72

Glosário de Termos

ATM	Auto Teller Machine
FDI	Fault Detection Isolation
MMD	Multi Media Dispenser
RBM	Raciocínio Baseado em Modelos
RBC	Raciocínio Baseado em Casos
SDA	Sistema de Diagnóstico Automático
SGDA	Sistema de Diagnóstico Automático Genérico

Resumo

Este trabalho introduz a definição de Sistemas Genéricos para Diagnóstico de Falhas, os conceitos envolvidos, e a sua utilização na manutenção de equipamentos de aplicação crítica em importantes setores industriais tais como a aeronáutica e bancária. As noções de Sistema Complexo Diagnosticável, bem como o processo de diagnóstico e sua classificação, são mostrados, sempre mantendo um foco sobre as questões do custo computacional dos modelos apresentados. As principais abordagens para o diagnóstico de falhas serão apresentadas e, ao longo do trabalho, será possível observar as várias definições, conceitos e pontos de vistas concernentes ao estudo da formalização do processo diagnóstico. Para completar, um estudo de caso será apresentado, tendo como foco um módulo dispensador de numerários. Tal equipamento é encontrado nos ATMs Bancários (do inglês: Auto Teller Machines, mais conhecidos no Brasil como “Caixas Eletrônicos”) e será apresentado como componente possível de ser monitorado por um Sistema Genérico de Diagnóstico Automático (SGDA).

Abstract

This work presents a formal definition for Generic Fault Diagnostic System as well as the fundamental concepts involved, and describes its application to the maintenance processes of complex machinery in the aerospace and banking automation industries. The concept of Diagnosticable Complex System is explored as well as the associated diagnostic processes focusing primarily on the computational burdens involved. As the main diagnostic approaches are presented, the various definitions, concepts, strengths and weaknesses of the formalization efforts will be explored to some extent. Finally, a case study involving the fault diagnosis of the multimedia dispenser module of a automatic teller machine (ATM) by an Automatic Generic Diagnostic System is analyzed.

1 Introdução

Com o avanço técnico e científico, as aplicações de engenharia (máquinas e processos) em setores de alta criticidade tais como o aeroespacial, nuclear, bancário e automobilístico se tornam cada vez mais automatizadas e integradas, o que gera situações de risco de falhas por vezes extremamente complexas. Além do mais, a alta disponibilidade dos equipamentos torna-se importante diferencial competitivo para as empresas do setor, tendo em vista ser esta característica cada vez mais exigida pelo aumento da demanda sobre os serviços oferecidos.

Nesse ambiente, os sistemas de diagnóstico surgem como recurso que contribui para atacar estes dois fatores: queda de risco e aumento de disponibilidade. Sua importância está sendo cada vez mais reconhecida pelo mercado. Um exemplo claro vem da área automobilística: em modelos de automóveis com alto grau de automação digital, mais de cinquenta por cento do software é dedicado ao diagnóstico.

O diagnóstico de falhas é realizado em equipamentos e processos quando eles são críticos e complexos. A correção rápida de defeitos em tais circunstâncias tem relevante impacto, não só do ponto de vista acadêmico, mas também econômico, na medida em que as ferramentas de diagnóstico se transformam em ferramentas de otimização que podem ser incorporadas em produtos bancários, aeronáuticos e outros com o intuito de minimizar custos de manutenção.

Neste sentido, cabe salientar aqui as experiências trazidas pelo FLD (*Flight Line Diagnostic*) desenvolvido pela Embraer em parceria acadêmica com a Escola Politécnica da USP, bem como o projeto CADS (*Computer Automatic Diagnostic System*) desenvolvido com a Diebold e o SAT (Sistema Auxiliar de *Troubleshooting*), com o apoio da Fundação Aplicações de Tecnologias Críticas - *Atech*.

O diagnóstico de falhas é uma área de estudo singular. Conceitualmente, o diagnóstico é o inverso da simulação. Enquanto a segunda deriva a descrição do comportamento do sistema a partir da estrutura deste, o diagnóstico, por outro lado, define o estado para a estrutura do sistema diagnosticado a partir do comportamento apresentado por ele. Filosoficamente, o diagnóstico de falhas tem uma visão fatalista: as falhas em Sistemas de Engenharia são inevitáveis e os defeitos vão acontecer, pode ser em curto, médio ou longo prazo, porém vão acontecer. Cabe então criar e estudar métodos eficientes para tratar sua correção.

Ao longo dos anos, esses métodos têm surgido nos meios acadêmico e industrial em grande número e nas mais diversas áreas. Infelizmente, esses métodos são, na maioria dos casos, conjunto de técnicas *ad hoc*. Mesmo as sistematizações do estudo pertinentes ao diagnóstico de falhas têm recaído em várias teorias gerais do diagnóstico, diferentes entre si em vários pontos e com o entendimento sobre suas semelhanças ainda incompletamente compreendido.

Será apresentada, portanto, a visão sistemática das abordagens. Logo em seguida, define-se o modelo central do trabalho: o **Sistema Complexo Diagnosticável**. Os conceitos básicos como **tipos de diagnósticos, sintomas, falhas e defeito** serão explicados em seguida e, posteriormente, a análise de dois exemplos de aplicação será realizada.

1.1 Organização do trabalho

Os capítulos desta dissertação estão organizados conforme segue:

- **Capítulo 2:** Apresentação dos conceitos intuitivos do diagnóstico. Neste capítulo será mostrada uma visão geral do que é diagnóstico, apresentando, sob uma perspectiva macro, os conceitos que serão desenvolvidos e discutidos nesta dissertação.
- **Capítulo 3:** Descrição das atividades da pesquisa médica e do raciocínio clínico como base para uma analogia com o estudo do diagnóstico de falhas. Neste capítulo serão exploradas mais detalhadamente as características do diagnóstico ainda sob o ponto de vista clínico. Destas características será derivado um conjunto de definições ligados ao conceito de diagnóstico que se tornarão recorrentes ao longo da dissertação.
- **Capítulo 4:** Apresentação do conceito de diagnóstico de falhas. Aqui, o foco passa para a área da engenharia, onde se analisará a multiplicidade de definições do que seja o diagnóstico, multiplicidade, esta, influenciada pelos autores e pelas abordagens adotadas. Será apresentado, também, um breve histórico da evolução do estudo e da pesquisa na área de diagnóstico de falhas.
- **Capítulo 5:** Explanação sobre a necessidade da formalização dos conceitos envolvidos no estudo do diagnóstico de falhas e as dificuldades dessa formalização. Neste capítulo é apresentado breve histórico da evolução no estudo de diagnóstico de falhas. É apresentado modelo a ser utilizado para demonstrar a aplicação de algumas abordagens.
- **Capítulo 6:** Apresentação da teoria de diagnóstico baseado em modelos.
- **Capítulo 7:** Apresentação da teoria de diagnóstico baseado em casos.
- **Capítulo 8:** Apresentação do modelo de pesquisa de panes.

- **Capítulo 9:** Apresentação de outras abordagens.
- **Capítulo 10:** Definição de sistemas automáticos de diagnóstico.
- **Capítulo 11:** Apresenta alguns pontos econômicos e operacionais que tornam importante o diagnóstico de falhas.
- **Capítulo 12:** É apresentado o conceito central da dissertação: Os sistemas complexos diagnosticáveis.
- **Capítulo 13:** Estudo de caso para um sistema remoto de diagnóstico para caixas eletrônicos.

2 Introdução ao conceito de diagnóstico

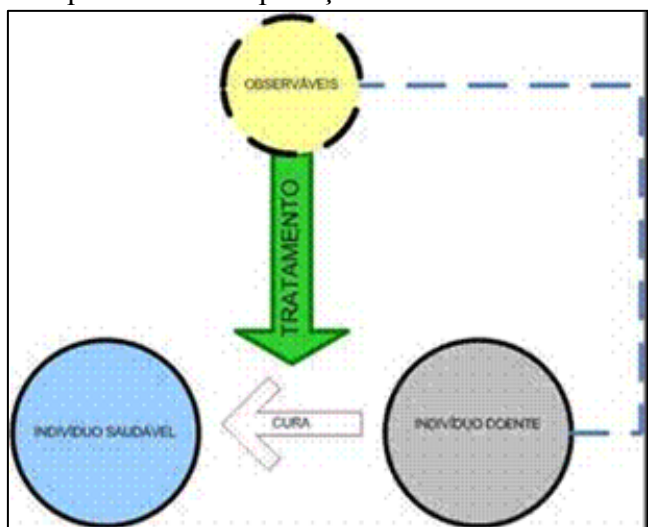
Em Medicina, temos as doenças e desordens do paciente. Já nos domínios que envolvem as tecnologias desenvolvidas pelo homem, encontramos os defeitos e falhas presentes nos dispositivos (LUCAS, 1997). Dessa forma, no momento da consulta, o médico busca solucionar o problema clínico do paciente, de modo a obter, como resultado do processo, determinada terapêutica ou tratamento que tenha resultado efetivo no sentido de resolver ou ao menos aliviar o estado do paciente, ou evitar que seu quadro progrida para estado crítico ou fatal.

Em Engenharia, analogamente, o técnico, seja ele um engenheiro ou não, busca eliminar o problema apresentado pelo equipamento durante a intervenção de manutenção. Obtém-se, como resultado do processo, o conjunto de ações corretivas que tenham resultado efetivo no sentido de restabelecer total ou parcialmente o funcionamento do equipamento, ou impede-se que ele atinja um estado irrecuperável de mau funcionamento, ficando incapacitado ou seriamente comprometido no cumprimento de suas funcionalidades.

Nas duas situações, os dois tipos de especialistas tiveram que trabalhar lançando mão de um conjunto de técnicas específicas (como testes ou análises) e competências próprias (como conhecimento formal e experiência) no objetivo de determinar as causas que levaram os elementos sob análise (no caso do médico, o seu paciente) ao estado indesejado.

Em seguida, o especialista deve propor ações que tenham implicações diretas ou indiretas nessas causas de forma a restabelecer seu estado normal. Para tanto, os dois dispõem somente de conjunto limitado de dados observáveis, sejam esses dados obtidos por entrevista, verificação visual, teste ou exame e é por meio deles que a relação causal deve ser estabelecida. Este processo está representado na figura (1).

Figura 1 - O processo de tratamento é baseado nas observações possíveis.



Desta forma, o especialista, seja ele da área de exatas ou de biologia, se encontra em situação semelhante à do desportista de *rafting* (corrida de botes em corretezas de rios): para cumprir a prova, ele tem que ter bom conhecimento sobre o tipo do rio que irá navegar e conhecer bem

suas características, sua trajetória e seu relevo. Durante a descida, ele deve ter a capacidade de reconhecer rapidamente sua posição relativa de modo a tomar decisões rápidas que o possibilitem realizar manobras precisas.

Porém, para conhecer bem o rio, o praticante de *rafting* deve ter muita experiência ou pelo menos ter estudado por tempo suficiente um bom mapa. Da mesma maneira, tratar um paciente ou consertar um equipamento exige do profissional a disponibilidade de uma base de conhecimento completa, detalhada e manuseável o bastante para lhe dar suporte em suas atividades. Desta forma, quando tiver que enfrentar um problema, esse especialista estará bem embasado e poderá dispor de um repertório de explicações possíveis grande o suficiente para lhe dar orientação segura ao analisar os dados colhidos de seu paciente ou equipamento.

3 Analogia médica: o raciocínio clínico

A pesquisa na área de diagnóstico de falhas é multidisciplinar na medida em que obtém subsídios de vários outros ramos da engenharia (tais como Teoria do Controle e Confiabilidade) e bebe em fontes de conhecimento dispares entre si, como a Medicina, a Ciência da Computação e a Filosofia da Ciência. Em sua raiz está a busca da aplicação de conceitos oriundos do raciocínio clínico na área de detecção, isolamento e tratamento de falhas. Esta dissertação defende a idéia de que é possível a analogia entre as atividades médicas e de engenharia via o conceito de diagnóstico e de áreas de estudo da medicina como a semiologia, a propedêutica e a sintomatologia.

3.1 As teorias médicas: explicando as doenças

Na medicina, construir a “biblioteca” das possíveis causas das doenças conhecidas se traduz em estudar, investigar e sugerir explicações para todos os possíveis distúrbios e doenças conhecidos. Tais atividades são de responsabilidade da pesquisa médica e científica que vêm se desenvolvendo em constante evolução desde a antiguidade. Tratar um paciente exige primeiramente que se determine qual é a causa do mal, suas razões de existir e sua descrição, de modo a explicar como e por que o problema ocorreu. Tais explicações são sintetizadas por meio das teorias médicas.

Tomemos como exemplo uma teoria médica muito significativa: a teoria microbiana das doenças (*the germ theory of disease*). Esta teoria surgiu em 1860 com Louis Pasteur e seu trabalho de analogia entre o processo de fermentação e as doenças. De forma brilhante, Pasteur observou que, se a fermentação era causada por bactérias, então as doenças também podiam ser causadas por microorganismos. Tal explicação revelou a causa microbiana de muitas doenças como a tuberculose e a cólera. Nas décadas que se seguiram, essa teoria foi estendida de modo a explicar o motivo de doenças causadas por vírus, protozoários e outros microorganismos.

A teoria microbiana revolucionou a história da Medicina e causou enorme impacto no desenvolvimento de políticas sanitárias. Para avaliar tal impacto, basta analisar o caso da tuberculose, doença infecciosa crônica temida há seis mil anos. A tuberculose tem conjunto típico de sintomas: fraqueza, perda de peso, tosse persistente (às vezes com presença de

sangue) e crescimento de tubérculos (nódulos) microscópicos nos pulmões, no cérebro (a neurotuberculose) e na pele (lupus) bem como em outras regiões. A evolução da doença, até o século vinte, terminava em óbito por exaustão ou por intoxicação respiratória em decorrência da falência das trocas gasosas realizadas pelos pulmões. Em 1882, Robert Koch descobriu, graças à teoria microbiana, que a causa da tuberculose era uma bactéria, chamada *Mycobacterium tuberculosis*, também conhecida como Bacilo de Koch. Em 1932 (portanto aproximadamente cinquenta anos depois da descoberta da causa da doença) surgiu o primeiro tratamento, quando Gerhard Domagk descobriu que esse tipo de micróbio podia ser morto pela droga Prontosil. O passo seguinte foi a descoberta da droga Streptomycina, em 1944, que reforçou o tratamento da doença. Hoje, apesar de ainda haver alta incidência ao redor do mundo, a tuberculose tem causa bem conhecida e conjunto variado de tratamentos possíveis.

A teoria microbiana surgiu como alternativa aos quase vinte e quatro séculos de teoria humoral de Hipócrates. Dessa forma, o desequilíbrio entre os quatro fluidos (humores) do corpo humano deixou de ser a principal causa do surgimento das doenças. Este fato mostra a dinâmica da pesquisa médica na construção de teorias que aumentem o universo de causas possíveis para doenças e disfunções: o surgimento de novas teorias pode complementar, conviver ou simplesmente substituir uma ou várias teorias aceitas anteriormente. E a importância de tal mudança pode ser medida pelo impacto que ela causa na ciência e na sociedade.

3.2 Os esquemas explanatórios

A experiência acumulada durante séculos de estudo demonstra que tais teorias fornecidas pela pesquisa médica recaem sistematicamente em padrão comum de raciocínio lógico. Segundo THAGARD (1999), este padrão é denominado Esquema Explanatório (*Explanation Schema*) e possui a seguinte sintaxe:

Esquema de Explanatório de Doenças

Pergunta Base

Por que o [PACIENTE] tem [DOENÇA] com [SINTOMAS] associados?

Padrão de Explicação

O [PACIENTE] está sujeito a [FATORES CAUSAIS]

Os [FATORES CAUSAIS] produzem a [DOENÇA]

A [DOENÇA] gera os [SINTOMAS]

Terapêutica

Os [AGENTES TERAPÊUTICOS] causam [REAÇÃO]

A [REAÇÃO] elimina/atenua [FATORES CAUSAIS]

O esquema explanatório é uma especificação semiformal. Seguindo a definição dada para as linguagens semiformais (MENDES e AGUIAR, 1989), o esquema explanatório é a cadeia de caracteres alfabéticos, numéricos ou especiais, possui sintaxe bem definida e semântica que não é precisa, porém é mais restrita que a das linguagens naturais.

Em esquema que represente caso real, os termos entre chaves são substituídos pelos elementos específicos da situação particular. Embora pareça muito simples (e até mesmo óbvio), muitos médicos e cientistas têm descrito seus trabalhos em termos de esquemas explanatórios (THAGARD, 1999). Na pesquisa médica, os Sintomas são manifestações observáveis da doença e se desenvolvem de diferentes maneiras ao longo de seu curso. Os Sintomas são decorrentes da causa ou causas da doença (etimologia) e o tratamento médico afeta os sintomas agindo em sua(s) causa(s).

Forma alternativa de representar o esquema explanatório é utilizando o chamado Diagrama de Estrutura Causal, cujo formato é o mostrado na figura (2).

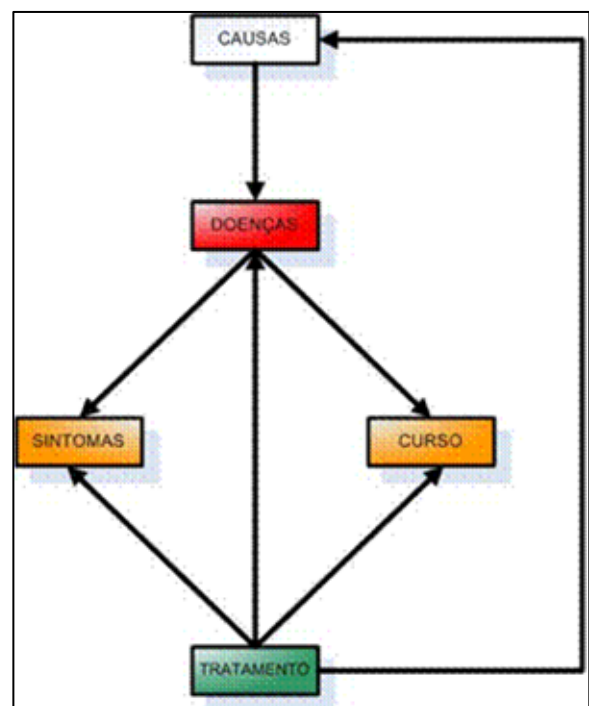


Figura 2 - Diagrama de estrutura causal

Com este diagrama, temos a representação gráfica completa das relações entre causa e efeito das doenças. Tal representação apresenta a vantagem de permitir que grande número de informações seja assimilado rapidamente. Esta representação utiliza simbologia gráfica acompanhada de cadeias de caracteres e símbolos alfanuméricos. Um esquema explanatório e um Diagrama de Estrutura Causal são representações equivalentes: enquanto o primeiro apresenta a explicação da doença em formato sintático, o segundo apresenta as mesmas informações em linguagem gráfica.

A título de exemplo, vamos ilustrar abaixo o esquema explanatório para o caso particular da cólera dentro da teoria microbiana.

Esquema de Explicação de Doenças

Pergunta Base

Por que o [PACIENTE] têm **cólera** com **diarréia, vômitos e febre** associados?

Padrão de Explicação

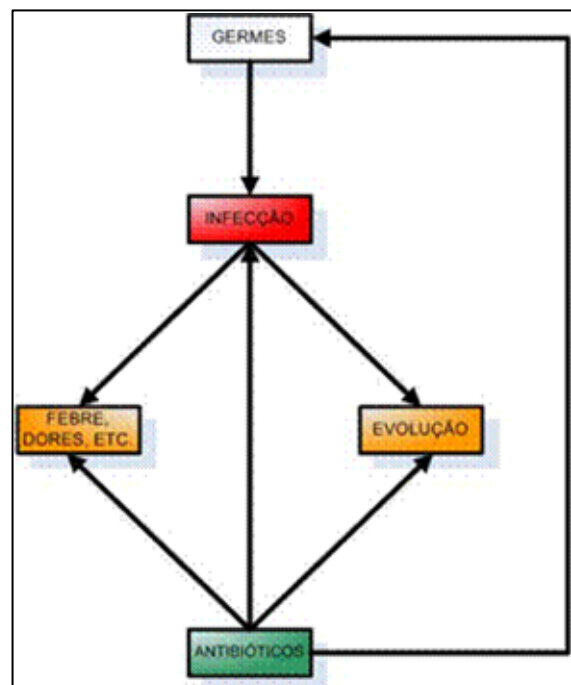
O [PACIENTE] está **infectado por micróbios** (*Vibrio cholerae*).

Os **micróbios** produzem **infestação microbiana intestinal e toxina que induz a secreção intestinal**.

A **infestação microbiana intestinal e a produção da toxina** causam a **diarréia, vômitos e febre**.

Podemos observar que o elemento que designa o paciente permanece indeterminado, enquanto os elementos que designam a doença e seus sintomas ganham suas instâncias para o caso particular da cólera. Na figura (3) temos o Diagrama de Estrutura Causal para a teoria microbiana.

Figura 3 - Exemplo de diagrama de estrutura causal para a teoria microbiana das doenças



3.3 A relação entre os esquemas explanatórios e o diagnóstico clínico

Se, por um lado a pesquisa médica visa descobrir as causas das doenças, por outro, durante o processo de tratamento, o médico tem que tomar uma decisão que responda à seguinte questão: qual doença ou conjunto de doenças fornece a melhor explicação para o comportamento do paciente? e complementar o esquema explanatório, agora para um paciente em particular. Ilustrando esse fato: o estado do esquema explanatório durante um exame assume então a forma apresentada abaixo para paciente imaginário de nome Doe que possui cólera, porém ainda não diagnosticada.

Esquema de Explicação de Doenças

Pergunta Base

Por que **Doe** tem [DOENÇA] com **diarréia, vômitos e febre** associados?

Padrão de Explicação

Doe está sujeito a [FATORES CAUSAIS]

Os [FATORES CAUSAIS] produzem [DOENÇA]

[DOENÇA] gera **diarréia, vômitos e febre**.

Ao processo de estabelecimento desses fatores causais e a explicação ao paciente dos motivos de seu mal-estar, damos o nome de diagnóstico, que vem do grego *diagnóstikós* – capaz de distinguir, de discernir, de mesma origem que *diagnose*, do qual se torna sinônimo, substantivado na locução grega *diagnóstikê (tékhne)* – arte de distinguir doenças, ou seja: explicar através do conhecimento.

Em geral, a complementação para o esquema explanatório pode ser obtida por um processo abduativo: de uma variedade de esquemas explanatórios possíveis dentro de uma ou mais teorias médicas que explicariam os sintomas do paciente, o médico escolhe aquele que caiba melhor com a situação. Às vezes, entretanto, um problema diagnóstico não admite solução abduativa. Nesse caso, as analogias podem ser úteis. A estrutura geral da analogia diagnóstica consiste em utilizar a solução de problema já resolvido para resolver novo problema.

3.4 As classes de conhecimento no raciocínio clínico

Os tipos de conhecimento empregados em diagnóstico são: conhecimento causal, conhecimento da estrutura e do comportamento funcional e associações empíricas.

Interações causais

Conseqüência direta da dos Esquemas Explanatórios, compreende também a informação taxonômica, que descreve as categorias de doenças e a inter-relação entre elas.

Considere a seguinte parte de conhecimento médico: “a gripe que se infiltra no corpo do paciente pelas vias aéreas, causa febre e infecção da traquéia e brônquios, que se reflete em uma dor de garganta; se o paciente sofrer de asma, a dificuldade da respiração ocorrerá também“. Usando a lógica como nossa língua da representação, a figura pode ser suposta para corresponder a:

$$D1 \rightarrow D2$$

$$D1 \rightarrow F1$$

$$D2 \rightarrow F2$$

$$D2 \wedge D3 \rightarrow F$$

Onde:

D1: Gripe

D2: Infecção dos Brônquios

D3: Asma

F1: Febre

F2: Dor de garganta

F3: Dificuldades respiratórias

Conhecimento empírico

A informação empírica representa o conhecimento derivado da experiência e da observação direta. São compostas de regras de classificação e geralmente significam conhecimento desestruturado. O conhecimento médico heurístico compreende os relacionamentos parcialmente compreendidos entre os sintomas e sinais do paciente, sejam obtidos por observações do médico, ou resultado de exames e testes de laboratório; as categorias gerais ou

específicas de diagnósticos ou tratamento são baseadas na experiência prática. Para o exemplo da gripe, a representação da informação heurística seria:

Para:

D1: Gripe

D2: Amidalite

F1: Dores articulares e musculares

F2: Dor de garganta

Temos:

F1 = D1

F2 = D2

Aqui, a implicação lógica é interpretada como relação de classificação. As heurísticas são essenciais para reduzir a necessidade de fazer muitas perguntas ou realizar testes supérfluos de laboratório e para tornar prática e eficiente a tarefa de analisar e sintetizar dados.

Conhecimento anatômico ou estrutural

Segundo (LUCAS, 1995), existe relacionamento forte entre a combinação de tipo particular de conhecimento para construir diagnóstico e a natureza do domínio subjacente do problema. Por exemplo, para determinação do Esquema Explanatório da Teoria Infecciosa, o conhecimento patológico e fisiológico do processo da doença pode ser usado. Mas em outros domínios médicos, como a neurologia, ortopedia e endocrinologia, resolver os problemas diagnósticos parte principalmente do conhecimento e descrição da função normal do sistema em questão, combinado ao conhecimento da estrutura anatômica do corpo humano como todo.

3.5 O processo de diagnóstico clínico e suas etapas

O processo de diagnóstico pode ser visto como o recolhimento e a interpretação seletivos de informações que trazem evidências da presença ou ausência de um ou mais defeitos no sistema diagnosticado. Esta definição informal revela dois aspectos que são de importância central para se resolver o problema diagnóstico: a coleta de dados relevantes e sua interpretação.

Em Medicina, a coleta de dados é processo realizado geralmente de forma sistemática e estruturada, pois há número muito grande de testes clínicos disponíveis. Além disso, alguns

testes causam desconforto ao paciente, ou mesmo carregam algum risco em sua aplicação, desta forma, os testes são executados somente quando necessários.

Embora o processo de coleta de dados como evidência seja uma característica importante do diagnóstico, a interpretação deles para o encontro da solução diagnóstica é o aspecto fundamental, em se tratando da teoria diagnóstica. O processo de interpretação pode ser descrito usando a noção científica do ciclo empírico, que descreve o fundamento da pesquisa científica (POPPER, 1974), com o intuito de selecionar entre um grande conjunto de possíveis causas, aquela mais representativa para explicar a situação do paciente (figura 4); essas etapas são:

- Formulação de um conjunto de hipóteses
- Teste das hipóteses
- Rejeição de uma hipótese

Cada uma das etapas será analisada.

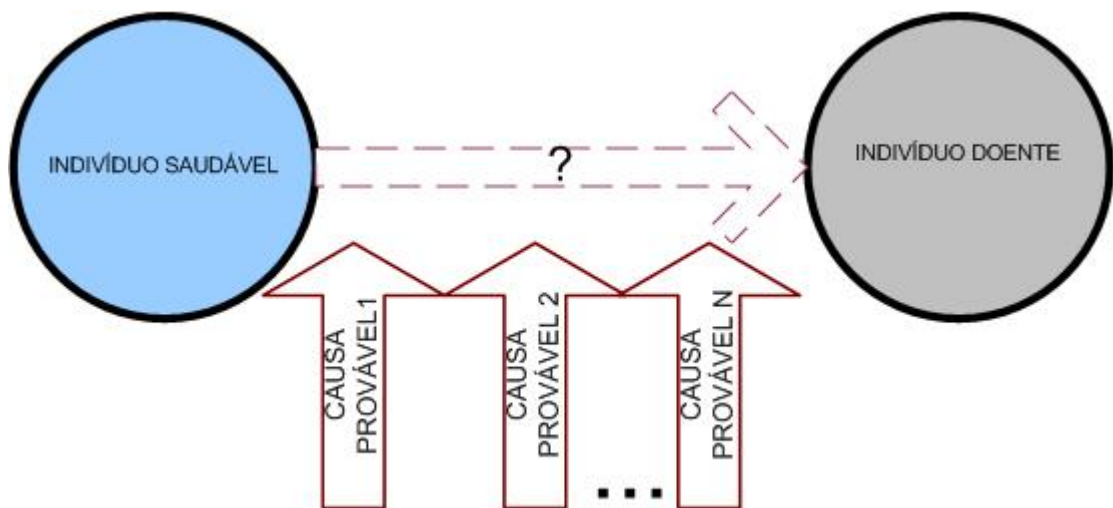


Figura 4 - Estado decisório antes do diagnóstico: múltiplas causas possíveis

3.5.1 Formulação de um conjunto de hipóteses

Devido à grande incerteza que caracteriza a fase inicial da primeira entrevista do médico com o paciente, as hipóteses iniciais têm uma função primordial: elas diminuem o conjunto das possíveis soluções e fornecem um contexto para a progressão do raciocínio clínico e da exploração diagnóstica (BARROWS e PICKELL, 1991). Cada hipótese diagnóstica evoca um possível esquema explanatório com o qual as manifestações clínicas do paciente podem ser comparadas (KASSIRER e KOPELMAN, 1991), cada uma representa a causa possível que explique o quadro presente.

Nessa etapa do processo de diagnóstico, o médico confronta o conjunto de dados clínicos obtidos com os conhecimentos que possui. Cada uma das hipóteses iniciais é selecionada a partir do conjunto de dados iniciais. Ele pode basear suas hipóteses nos três tipos de conhecimento descritos acima, entretanto, os médicos utilizam-se mais freqüentemente sua experiência (BORDAGE E ZACKS, 1984). Tanto a disponibilidade quanto recuperação do conhecimento baseado na experiência são variáveis críticas no processo de raciocínio clínico e se relacionam diretamente à qualidade da solução proposta pelo profissional.

3.5.2 O Teste das hipóteses

É necessário, após o médico ter construído seu conceito inicial e várias hipóteses serem levantadas, iniciar o processo de teste e avaliação das hipóteses. Nesta fase, cabe ao médico responder quais informações serão necessárias para estabelecer a hipótese e de qual maneira ele obterá essas informações. Informações clínicas englobam dados obtidos pela anamnese (entrevista com o paciente), exame físico, testes de laboratório e procedimentos.

Porém não são apenas quais os elementos (testes ou perguntas) que o médico deve definir, mas como e em qual ordem eles serão utilizados. Se a queixa inicial for muito vaga, o médico entrevistará o paciente com questões que esclareçam mais a natureza do problema dentro do conjunto das suas hipóteses iniciais. As perguntas iniciais são mais abrangentes, para obter mais informações, o médico pode utilizar perguntas mais específicas. Nesta situação, a ordem em que as perguntas são feitas ao paciente também é ponto importante, pois pode abreviar a sua quantidade no caso de uma pergunta bem colocada trazer informações que eliminem a necessidade de se realizar outro conjunto perguntas.

Os critérios que determinam a ordem dos testes e das perguntas são dependentes dos atributos e das informações que compõem esses testes, tais como: precisão, sensibilidade, especificidade, custos e riscos.

As propriedades inerentes das hipóteses, tais como a taxa de incidência ou a probabilidade da causa associada (FLETCHER *et al.*, 1989) também devem ser levadas em conta. Dessa forma, a maior utilidade dos testes diagnósticos ocorre nos pacientes com hipóteses que possuam alta probabilidade pré-teste referente a uma causa. Nesses pacientes, a probabilidade pós-teste da doença aumenta notavelmente com resultado positivo (principalmente se o teste for bem específico) e diminui na mesma proporção com o teste negativo (principalmente se o teste for bem sensível) (NETO, 1998).

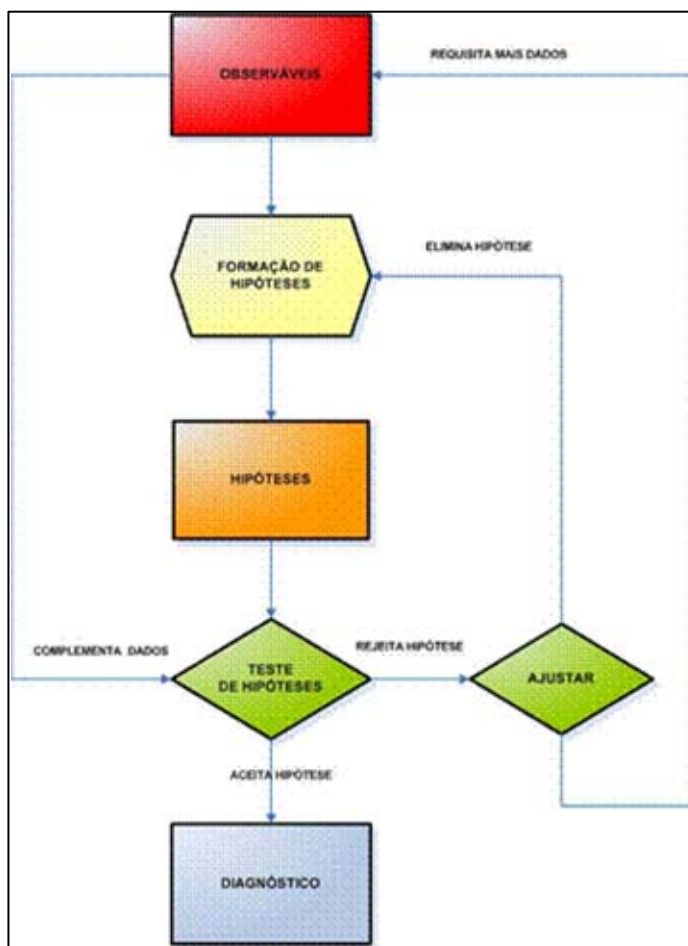
Para dar um exemplo simples, suponhamos que durante uma epidemia de meningite um paciente que dê entrada em um hospital com forte dor de cabeça. Tal paciente é candidato a uma bateria de testes que confirmem ou neguem a presença desta doença, pois, em função da alta taxa de incidência da mesma durante aquele período específico do ano, a probabilidade do quadro do paciente ser de contaminação pelo vírus da meningite é alta.

3.5.3 Rejeição de uma hipótese

Quando não passa nos testes, a hipótese é rejeitada. Esta etapa pode envolver a coleta de observações adicionais. O processo pode recomeçar com a inserção de uma nova hipótese ou, possivelmente, com o ajuste de uma hipótese já existente, modificando-a. O diagnóstico será, portanto, a hipótese (ou hipóteses) restante(s).

Figura 5 - Fluxograma do processo de diagnóstico

Na figura (5) é mostrado o processo de diagnóstico como variação do ciclo popperiano. Testar envolve verificar a verdade ou falsidade da hipótese



usando os fatores observáveis constituídos pelos dados obtidos na etapa da coleta. Em geral, a hipótese é uma seqüência de relações causais que têm comportamento anormal ou normal em uma ponta (quase sempre determinado por conjunto de fatores observáveis) e a identificação de parte do elemento diagnosticado na outra. Esta simplificação nem sempre pode ser justificada pois, por o exemplo, as hipóteses podem ser relacionadas entre si criando dependência. Para o exemplo, a hipótese A pode ser ou não uma causa para a existência da hipótese B. Não obstante, essa simplificação é desejada e aceitável para manter a eficiência e o controle das aplicações diagnósticas.

O objetivo desse capítulo foi ilustrar de forma intuitiva os principais conceitos do processo diagnóstico do ponto de vista clínico. As ilustrações médicas foram rigorosas até o ponto necessário para esclarecer tais conceitos. As idéias estudadas neste capítulo, embora não embasado pelas “ciências duras” (matemática, lógica, etc.), serão recorrentes ao longo de todo o trabalho e aparecerão na maioria dos modelos formais de diagnósticos aqui apresentados.

O resumo fundamental é que, do ponto de vista clínico, diagnósticos corretos são baseados em raciocínios adequados e em informações válidas (NETO, 1998).

4 O Diagnóstico de falhas

Não existe na literatura definição única ou consenso em relação aos conceitos e elementos que fazem parte do estudo do diagnóstico de falhas, sequer existe uma definição única do que seja o diagnóstico de falhas. Porém, apesar de existirem várias definições, variando de autor para autor, todas elas circundam idéias semelhantes.

Desta forma, não é possível enunciar uma única definição do que seja o diagnóstico de falhas, mas analisar as semelhanças e complementos entre as várias definições apresentadas e aceitas. Em KRYSANDER (2003) e NYBERG (2000) temos:

“Geralmente o diagnóstico é investigação ou análise da causa ou natureza de uma condição, situação ou problema como, por exemplo, o diagnóstico do problema em um motor. O diagnóstico pode também ser uma indicação ou a conclusão de tal análise”.

Esta definição, embora enfoque bem o diagnóstico como o estabelecimento de relação causal, é um tanto vago e mesmo o exemplo empregado na própria não traz maiores esclarecimentos.

Já para MCILRAITH (1999): “O objetivo do diagnóstico é tecer conjecturas sobre o que esta ocorrendo de errado com o Sistema, (quais os componentes do dispositivo estão se comportando anormalmente? De qual doença o paciente está sofrendo?)”.

De maneira semelhante, MOZETIC (1984) afirma que o diagnóstico de falhas “consiste em determinar, de acordo as observações para um Sistema que se comporta de maneira anormal, quais são os Subsistemas que geram este comportamento”.

Nesta definição nos deparamos com a relação hierárquica entre “Sistemas” (o todo) e “Subsistemas” (as partes) no processo de diagnóstico.

A mesma relação torna a surgir na definição dada em FIJANY, VATAN (*et al.*, 2002):

“O diagnóstico de um sistema é a tarefa de identificar os componentes defeituosos que fazem com que o sistema não funcione como esperado. O problema do diagnóstico surge quando um conjunto de sintomas é observado, isto é, quando o comportamento real de Sistema está em contradição com o comportamento previsto”.

Além da descrição do diagnóstico como processo analítico, é citado o conceito de “Sintoma” como a confrontação entre comportamento esperado e comportamento real e a conceituação de “componente” como sendo um subsistema.

Para LUCAS (1997):

“O Diagnóstico é visto como a interpretação da observação de uma situação específica diante do contexto representado pelo conhecimento referente ao domínio do problema, de modo a se obter uma indicação da presença ou ausência de falhas ou defeitos e determinar a natureza de suas origens”.

Aqui, autor usa os termos “falha” e “defeito” (embora não esclareça a semelhança ou diferença entre ambos), reportando ao conceito de anormalidade visto anteriormente, e detalha a confrontação entre o comportamento observado e o comportamento esperado em termos de evento temporal (“observação de uma situação específica”) versus conhecimento atemporal (“conhecimento do domínio do problema”).

Algumas definições enfocam o diagnóstico como processo e citam suas etapas, como em GASCA (*et al.*, 2003):

“O diagnóstico serve para identificar as peças que falham em um sistema. Integra geralmente o monitoramento (com sensores que, por definição, não falham) e o diagnóstico (detecção e identificação de falhas)... A explicação de um comportamento errôneo de uma observação determinada é a tarefa principal do diagnóstico”.

Ao enxergar o diagnóstico sob o ponto de vista do processo, o autor define a etapa de monitoramento como responsável por obter a observação do comportamento do sistema. Os sensores surgem como agentes responsáveis por essa aquisição de informações e têm como características, ainda segundo o autor, sua alta confiabilidade, de modo a tornar o conjunto de dados obtidos aderente ao conceito abstrato de observação.

Mesmo trazendo mais detalhes ao conceito de diagnóstico (dividido agora em um processo que compreende a aquisição de informação e sua interpretação) o texto não deixa claro os termos da segunda etapa que ele rotula genericamente como “diagnóstico” o que resulta na definição do diagnóstico em termos do próprio diagnóstico. A única informação dada pela definição sobre essa segunda etapa é que ela é composta de duas sub-etapas (detecção e identificação das falhas) sem maiores detalhes. Seria mais apropriado se a definição esclarecesse o significado dessas sub-etapas de detecção e identificação oferecendo uma descrição, por mais breve que fosse, de cada uma.

Como visto acima, a atividade diagnóstica possui dois grandes componentes: um conhecimento *a priori* do domínio do problema e a estratégia que permita ao diagnosticador chegar a uma conclusão pertinente.

As primeiras abordagens para a construção de sistemas de diagnóstico consistiam em capturar a experiência do especialista em um conjunto de regras empíricas. Para se tratar a questão do diagnóstico de falhas, estabelecia-se, por meio de um conjunto de regras, a relação imediata entre causa e efeito para todas as situações de falha passíveis de ocorrer no sistema em análise. Tal “tabela” de regras era elaborada de forma exaustiva (pois, em tese, deveria ser correta e completa) e abarcavam todas as situações possíveis pela qual se pudesse derivar tal conjunto de regras: sejam de dados experimentais (empíricos), históricos, para cenários levantados *a priori* (pelo uso de árvores de falha, por exemplo) ou mesmo pela experiência da equipe técnica envolvida na manutenção.

Um exemplo típico deste modelo pode ser visto abaixo, neste fragmento do conjunto de regras aplicado ao diagnóstico de automóveis em formato de sentenças condicionais semelhante ao ilustrado por BALAKRISHNAN e HONAVAR (1998).

Regra 1:

SE Faróis não funcionam

ENTÃO lâmpada queimada OU/E Bateria descarregada

Regra 2:

SE Faróis não funcionam E veículo não liga

ENTÃO Bateria descarregada

Regra 3:

SE Faróis não funcionam E veículo liga

ENTÃO lâmpada queimada

De fato, as primeiras ferramentas de diagnóstico desenvolvidas nos anos setenta foram baseadas nesse modelo. Porém, as limitações advindas desse modelo estático eram muitas, a principal era que a adoção do conjunto de regras fixas tornava essas ferramentas dependentes do equipamento ou processo que diagnosticavam, o que inviabilizava a utilização do mesmo sistema de diagnóstico em equipamentos diferentes, pois as regras derivadas eram fortemente dependentes do tipo do equipamento diagnosticado. O conjunto de regras construídas para um

sistema de diagnóstico utilizado em aviões eram brutalmente diferentes das regras geradas para diagnóstico em automóveis.

Os pesquisadores que estudavam o desenvolvimento de tais ferramentas perceberam então que desenvolver sistemas diagnósticos eficientes era mais difícil do que pensavam. Eles concluíram, então, que os princípios subjacentes ao processo de diagnóstico não eram, até o momento, bem compreendidos e que uma formalização mais substancial era necessária.

Dessa forma, em meados dos anos oitenta, CHANDRASEKARAN (1988) e sua equipe adotaram a abordagem baseada no modelo de resolução geral de problemas (*general problem-solving*) de Allen Newell e Herbert Simon para atacar a questão do diagnóstico de falhas. O modelo de Newell e Simon tinha como objetivo imitar os protocolos humanos de resolução de problemas (RUSSEL e NORVIG, 2003) e sua capacidade de atacar questões e obter soluções independente do domínio em questão. Ao usar esse modelo, um dos principais objetivos de Chandrasekaran era alcançar uma solução que tornasse o sistema de diagnóstico independente de uma aplicação específica. Desta forma, sua abordagem enxergava a solução diagnóstica como um processo onde as invariantes eram as etapas que compunham o processo, e não o equipamento a ser diagnosticado, este poderia mudar sim, de acordo com a aplicação do momento.

Desta forma, as etapas do diagnóstico deveriam ser necessariamente as mesmas, independentes do equipamento a ser diagnosticado, refletindo o modo como a mente humana tratava o problema da diagnose. Esta era uma visão bem diferente da simples busca monolítica em uma lista de regras ao qual consistiam os primeiros sistemas.

Mas como possibilitar essa generalidade? Para Chandrasekaran, o desacoplamento da aplicação era possível através de um processo independente de geração de hipóteses. Resolver o problema diagnóstico era visto, portanto, como um caso especial de raciocínio hipotético, em que múltiplas hipóteses são geradas. Diagnosticar consiste em selecionar a hipótese a respeito da presença de falhas ou de processos anormais em função de testes realizados, rendendo o diagnóstico (ou diagnósticos) que explicam a situação presente.

Outros grupos, por sua vez, voltaram seus focos na questão da representação do domínio do problema em sistemas diagnósticos, surgiu então o diagnóstico baseado em modelos. Um dos fatos curiosos sobre o diagnóstico baseado em modelos é que os primeiros sistemas desenvolvidos sob essa abordagem, o INTER (KLEER, 1976) e o CASNET (WEISS, *et al.*, 1978) foram desenvolvidos antes da formalização da teoria.

Segundo os pesquisadores desta vertente, o conhecimento *a priori* do domínio do problema pode ser desenvolvido a partir do entendimento das bases e fundamentos do comportamento do equipamento diagnosticado, ou seja: a partir do conhecimento do funcionamento de um determinado equipamento e possível antever seu comportamento de modo a comparar a previsão com a situação real vivida. A abordagem baseada em modelos defende a construção de sistemas diagnósticos onde os dados obtidos da observação são confrontados com o comportamento previsto com base no conhecimento formal e explícito fornecido pelo modelo, que descreve as interações estruturais e funcionais entre componentes do sistema diagnosticado, bem como as interações causais entre seus elementos.

5 A formalização da teoria diagnóstica

Como as aplicações diagnósticas são quase sempre críticas, a formalização torna seguro o processo de diagnóstico. Desta forma, a formalização de uma teoria diagnóstica é imprescindível.

De acordo com LUCAS (1995), há diversas vantagens oriundas da formalização da teoria diagnóstica: uma estrutura formal, tanto lógica como matemática, tem sintaxe e semântica bem definida, conseqüentemente apresentando significado claramente unico do conhecimento referente ao domínio do problema (diagnóstico); logo, toda informação derivada do conhecimento primário, se bem gerada, terá como características a estabilidade e a integralidade. Como resultado, toda conclusão diagnóstica derivada será confiável, relevante e sem ambigüidade, pois pode ser efetivamente verificada.

Porém, o processo de formalização do diagnóstico não é imediato. Embora esteja intuitivamente compreendido que o diagnóstico tem como objetivo explicar o comportamento anormal, é muito difícil definir de forma precisa e abrangente o conceito de diagnóstico. Por exemplo, já é difícil definir o que significa um comportamento anormal, pois existem diferentes formas de descrever o que é anormalidade (influenciadas pelas diferentes noções de causalidade), bem como existem diferentes graus de detalhamento possível para as definições do conceito de anormalidade, conseqüência do propósito ao qual a definição serve.

Segundo CORDIER (2004), dada a importância do problema de detecção e isolamento de falhas para os sistemas críticos, uma ampla gama de abordagens de diagnóstico de falhas tem sido desenvolvidas. Muitas dessas abordagens assumem óticas que vão desde a análise puramente lógica àquelas que utilizam ferramentas estatísticas como a teoria da estimação e a teoria da decisão.

Até o presente momento, neste trabalho, foi apresentado um sumário da evolução das idéias referentes à pesquisa sobre diagnóstico de falhas. Nos capítulos seguintes, será apresentada uma descrição de cada uma das abordagens mencionadas na retrospectiva histórica do capítulo quatro. Uma maior ênfase, do ponto de vista da formalização, será dada para a abordagem baseada em modelos, por tal abordagem o foco dessa dissertação. Apesar disso, a visão dada às outras abordagens não será de modo algum superficial: cada uma delas será descrita e discutida em seus aspectos mais importantes.

6 O Raciocínio baseado em modelos (RBM)

Na retrospectiva histórica presente no capítulo quatro, foi visto que os primeiros sistemas de diagnósticos desenvolvidos foram baseados em regras. Cabe fazer algumas observações sobre essa abordagem para acaminhar a discussão sobre o paradigma RBM de diagnóstico, que é o objetivo deste capítulo.

Em sistema baseado em regras, o conhecimento é tipicamente oriundo de um perito que tenha a experiência diagnóstica necessária com o dispositivo. Esses sistemas são construídos capturando tal experiência e transformando-a em uma seqüência de regras que associem anomalias às causas possíveis. Embora existam muitos sistemas de diagnóstico baseados nessa abordagem, ela é considerada ingênua, por apresentar limitações sérias (STEELS, 1988), algumas delas já citadas no sumário histórico do capítulo quatro.

Segundo GRESSE (2003), o processo de derivação das regras condicionais em tais sistemas pode ser trabalhoso e pouco confiável, sendo difícil para os especialistas expressar o conjunto de conhecimentos tácitos no formato das regras, e mais difícil ainda garantir sua correção.

Mesmo os exemplos simples de derivação e elaboração de regras condicionais, como o exemplo de BALAKRISHNAN e HONAVAR (1998), apresentado na página vinte e oito deste trabalho, podem ser problemáticos. Estes problemas têm origem em vários fatores, dentre os quais pode-se citar: a interpretação pessoal do técnico, não corroborada tecnicamente e transformada em regra, regras mal construídas (apresentando redundâncias ou encadeamentos muito longos) e má formulação na descrição do domínio do problema (uma descrição mal elaborada acerca dos componentes de um sistema e sua estrutura e comportamento pode gerar um conjunto de regras menos eficiente). Uma vez que o conjunto de regras foi estabelecido para uma aplicação específica, seu reuso é difícil para qualquer outra aplicação, pois tal conjunto de regras é fortemente dependente do dispositivo para o qual elas foram geradas.

Outra limitação está associada à interação entre sistema e usuário. Sistemas baseados em regras fornecem explicação muito limitada da solução alcançada. Desde que o conhecimento em tais sistemas seja representado por regras no formato CONDIÇÃO-AÇÃO, as explicações consistem em rastrear as regras que foram usadas durante o processo de diagnóstico. Tal rastreamento pode não ser muito compreensível para os técnicos envolvidos no processo de diagnóstico. Técnicos humanos tendem a trabalhar melhor com justificativas próximas à sua

linguagem do dia a dia. Simplesmente apresentar as regras utilizadas por um determinado sistema (mesmo que essas regras tenham sido construídas com o auxílio dos próprios técnicos) implica em apresentar uma justificativa em um formato artificial e lógico que exige um esforço de compreensão para pessoas não-treinadas, conseqüentemente, as soluções podem não ser facilmente aceitas.

O estudo dos sistemas de diagnóstico por RBM surgiu no intuito de libertar o processo de diagnóstico dessas limitações pela construção de um modelo que representasse, de maneira completa o suficiente, o equipamento diagnosticado e tornasse configurável a representação deste equipamento, bem como do seu comportamento. O básico do paradigma RBM reside nesse fato.

Construindo um modelo de estrutura e comportamento do equipamento, é possível definir o comportamento normal (ou anormal), e confrontá-lo a uma bateria de observações realizadas no equipamento real. O quão próximo esse modelo de estrutura e comportamento deve ser próximo do real? Tanto quanto a aplicação diagnóstica necessitar e a quantidade de informações e conhecimento acerca do equipamento permitir.

De certa forma, fazendo analogia com o raciocínio clínico, o paradigma RBM é a abordagem do processo diagnóstico com a ênfase dada ao conhecimento “anatômico”, a partir do momento em que usa o modelo explícito do equipamento diagnosticado. De um lado, temos o conhecimento da organização e do comportamento do equipamento; do outro, temos seu comportamento real observado. A diferença entre ambos é chamada discrepância. Quando não ocorre discrepância, há corroboração. Corroboração e discrepância são usadas para identificar as partes do equipamento que estão se comportando de forma incorreta. A localização e o tratamento de falhas se dão pelo entrecruzamento das informações referentes aos grupos de componentes que estão envolvidos em cada uma das discrepâncias detectadas.

O modelo do equipamento diagnosticado é derivado da informação do seu projeto, e consiste tipicamente no conhecimento sobre a estrutura (componentes e suas conexões) e comportamento (relações de entrada e saída dos componentes) do artefato. Isto facilita muito o processo de aquisição de conhecimento, se comparado aos sistemas baseados em regras, nos quais quase todo conhecimento tem que ser adquirido dos especialistas e peritos. Tal facilidade deriva do fato de ser possível exercitar, através do modelo, as possíveis situações

de falhas dos componentes que constituem o modelo, bem como observar a alteração do comportamento do equipamento como resultado dessas falhas.

Os sistemas de diagnóstico RBM manifestam comportamento mais robusto que os sistemas baseados em regras, pois podem tratar tanto os problemas mais comuns quanto problemas novos e inesperados, visto que os sistemas baseados em regras lidam somente com problemas pré-codificados. A base de conhecimento explícita (o modelo do dispositivo) dos sistemas de RBM pode, teoricamente, explicar todas as falhas possíveis, pois sua única limitação é o grau de detalhamento utilizado no modelo. Já o conhecimento contido nos sistemas baseado em regras foi extraído da experiência do especialista humano e não há nenhuma garantia de que tal conjunto de regras possa esclarecer todas as falhas possíveis.

Os sistemas RBM são mais reutilizáveis que os baseados em regras. O modelo do equipamento captura melhor o conhecimento geral que pode ser aplicado no processo de diagnóstico. Parte disso se deve ao fato de o comportamento global estar particionado no comportamento de cada componente, bem como pela própria separação entre estrutura e comportamento oriunda do modelo. Como os componentes podem ser usados em vários artefatos diferentes, os modelos de componentes também podem ser reutilizados em sistemas de diagnóstico que monitorem outros equipamentos que contenham os mesmos tipos de componentes em comum.

Porém, o paradigma RBM também tem suas limitações. Segundo DAVIS (*and* HAMSCHER, 1988), a questão crítica no diagnóstico baseado modelo é a construção do modelo do equipamento *a priori*. Tal questão tem um nome específico: *o problema da modelagem*. Segundo esse autor, sempre haverá um “teto” do que é possível ser modelado. Tal “teto” está relacionado com as informações disponíveis para se gerar o modelo. A questão é se este limite satisfaz totalmente ou não o grau de detalhamento que se deseja e o quão próximo do real o modelo precisa ser para que a análise de discrepância seja eficiente e informativa.

Obviamente, o desejável seria que o modelo reproduzisse integralmente o equipamento que ele representa nos mínimos detalhes, a ponto do modelo se confundir com o próprio equipamento, de maneira que não houvesse diferença entre ambos (ou que tal diferença fosse imperceptível), como isso jamais será possível, um bom termo deve ser alcançado com os recursos disponíveis.

Outra dificuldade, quando se fala em RBM, vem do fato que realizar um diagnóstico a partir desse paradigma pode ser caro nos termos computacionais. Segundo BYLANDER (1989) e RYMON (1991), diagnosticar pela abordagem RBM, no caso geral, é um problema NP.

O modelo é independente do domínio, mas depende da representação e do formalismo utilizado para expressá-lo. De fato, existem duas grandes correntes que pesquisam a área da teoria do diagnóstico de falhas pelo paradigma RBM: a comunidade DX, com forte influência da área de Inteligência Artificial (IA), e a comunidade de detecção e isolamento de falha ou *Fault Detection and Isolation (FDI) community*, proveniente de grupos oriundos da área de Engenharia de Automação e que baseia sua análise de RBM na Teoria de Controle (CORDIER, 2004). As duas abordagens diferem em suas maneiras de interpretar o que é o diagnóstico no contexto de raciocínio baseado em modelos: enquanto a comunidade DX entende o diagnóstico como processo de inferência lógica, os grupos pertencentes à corrente FDI vêem o diagnóstico como processo analítico. Na abordagem FDI, como na maior parte da teoria de controle, o diagnóstico é baseado em modelo quantitativo dinâmico do sistema sob o estudo, representado geralmente como conjunto de equações algébricas ou diferenciais, enquanto na abordagem DX, o modelo da estrutura e comportamento é representado por predicados lógicos.

Segundo CORDIER (*et al.*, 2000a, 2000b):

“Embora as abordagens sejam baseadas nos mesmos princípios, cada comunidade desenvolveu seus próprios conceitos, ferramentas e técnicas. Os modelos formais empregados em cada linha vão certamente para campos técnicos muito diferentes: modelos analíticos e algébricos de um lado e modelos simbólicos e qualitativos baseados na lógica do outro”.

De igual maneira, VENKATASUBRAMANIAN (*et al.*, 2003) afirma que:

“Na abordagem diagnóstica baseada em modelos, o conhecimento *a priori* pode ser classificado em quantitativo ou qualitativo, dependendo de como se escolhe representar o conhecimento à cerca do modelo [do sistema diagnosticado]. Nos modelos quantitativos, esse conhecimento é representado por meio de relações matemáticas funcionais entre as entradas e saídas do sistema. Em contraste, no qualitativo, o conhecimento é representado por funções lógicas centradas no comportamento dos diferentes componentes que fazem parte do sistema”.

As especificidades técnicas e terminologia própria, junto ao fato de que cada grupo tem seu próprio ambiente de conferências e publicações, resultaram durante muito tempo em um nível relativamente alto de isolamento entre as duas linhas de estudo RBM. Ambas as abordagens têm seus pontos fortes e serviram de inspiração para o modelo de sistemas complexos

diagnosticáveis, que será apresentado no capítulo onze deste trabalho. Desta forma, uma descrição e uma análise mais detalhadas sobre as duas vertentes é agora necessária.

6.1 Descrição de um exemplo de ambiente de teste para a aplicação e comparação entre os modelos das comunidades DX e FDI

Neste ponto do trabalho, pode surgir uma pergunta: se ambas as comunidades, FDI e DX, são casos especiais de uma mesma abordagem (a abordagem de RBM) o que as diferencia realmente? Como o fato já declarado de que uma comunidade tem uma visão baseada em expressões lógicas e outra adota uma visão analítica as tornam efetivamente diferentes dentro da mesma abordagem?

O sistema cujo diagrama está descrito logo abaixo é um exemplo já clássico e considerado padrão em publicações voltadas ao diagnóstico de falhas.

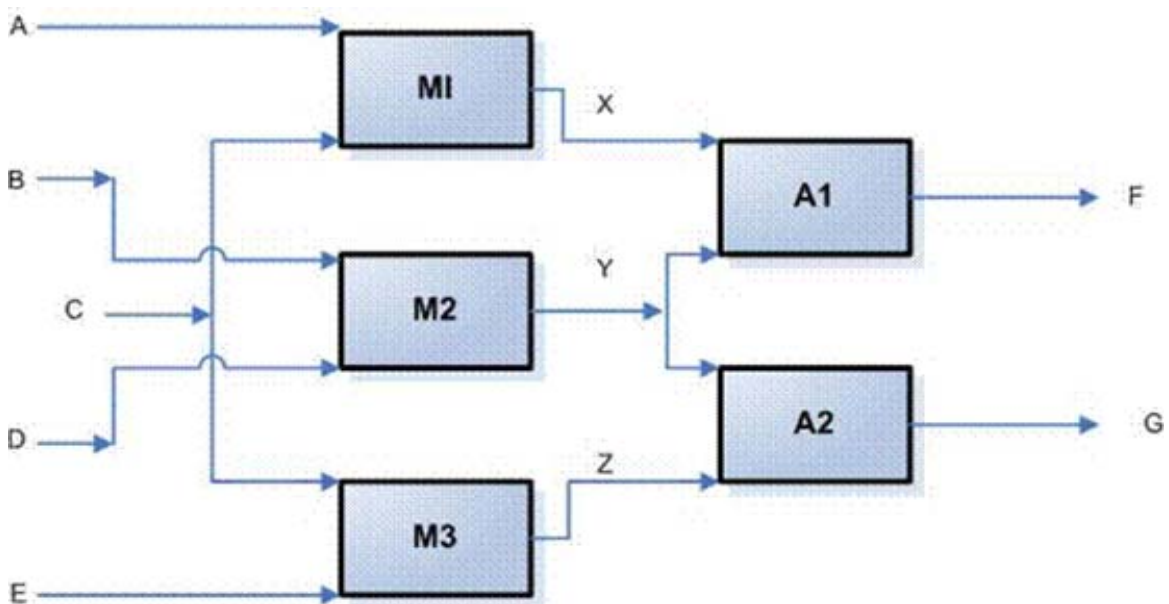


Figura 6

Tal sistema é chamado de *consistence-model expository example* em algumas vertentes de pesquisa diagnóstica, entretanto a maior parte dos trabalhos se refere a ele pelo nome mais simples de *polybox* ou *polycell*. Ele foi utilizado inicialmente em DAVIS (1984), GENESERETH (1984) e KLEER (1987). Desde então, a maioria dos trabalhos sobre diagnóstico automático de falhas o utiliza para a validação de novas idéias e modelos.

Sua utilidade neste trabalho consiste em evidenciar as “sutilezas” nas diferenças apresentadas pelas duas principais correntes de estudo do diagnóstico baseado em modelos. Será, portanto,

com o auxílio do exemplo do *polibox*, evidenciado o significado do caráter analítico da comunidade FDI, bem como o caráter lógico da comunidade DX, construindo, assim, uma comparação didática entre as duas comunidades. O *polybox* será utilizado ao longo do restante deste capítulo.

O sistema *polibox* possui um conjunto de pontos de entrada (a, b, c, d, e) bem como um conjunto de pontos de saída (f, g) mais um conjunto de pontos de prova (x, y, z).

Segundo (NYBERG, 2000): “O exemplo do *polybox* é construído para proporcionar uma expressão logicamente coerente do diagnóstico, isto é, uma listagem completa e correta das modalidades possíveis de falha”.

O sistema é formado por três multiplicadores genéricos M_1, M_2, M_3 e dois somadores A_1 e A_2 também genéricos conectados conforme a figura. Trata-se de sistema estático e discreto, de maneira que podemos ignorar comportamentos transitórios. Embora os modelos aqui descritos e analisados possam ser aplicados tanto em sistemas discretos quanto contínuos, a idéia proposta por esse exemplo consiste em manter a clareza e simplicidade que enfatizem apenas os conceitos diagnósticos dos modelos, sem sobrecarregá-los com detalhes irrelevantes.

Por definição, o ambiente em que o sistema opera é livre de ruídos e perturbações de qualquer natureza. O enunciado do problema é simples: determinar qual componente está em estado defeituoso tendo apenas o conjunto dos observáveis (a,b,c,d,e,f,g) como fonte de informação.

A descrição da utilização do *polibox* como exemplo de sistema diagnosticável e sua aplicação para as duas vertentes (DX e FDI) do paradigma de diagnóstico baseado em modelos teve como base os trabalhos de P. M. FRANK (1996) e CORDIER (2000, 2004).

6.2 A Comunidade FDI

Sendo a mais antiga das duas abordagens de RBM, a vertente FDI teve início com os trabalhos de CLARK (1978), HIMMELBLAU (1978) e PAU (1981). Historicamente, o estudo do diagnóstico de falhas por parte da comunidade de controle tem como objetivo automatizar a tarefa denominada de Gerência de Eventos Anormais (*Abnormal Event Management* ou AEM, em inglês) de modo a se melhorar a obtenção dos tempos necessários para a tomada do conjunto de ações e decisões (sejam eles a ativação de redundâncias, reconfiguração, manutenção ou reparo) que evitem desastres que possam resultar em perda de equipamentos caros e danos possíveis ao ambiente e às pessoas.

Dentro da área de estudo do controle automático, as funções supervisionais servem para indicar os estados indesejados ou proibidos do processo, de forma a se empreenderem as ações cabíveis de retirem o processo destes estados, a fim de manter a operação em andamento e evitar danos ou acidentes. Tais funções podem ser subdivididas em dois tipos: funções de monitoração e proteção automática

Função de monitoração

Nesta função de supervisão, dado um processo relevante e crítico, os valores das suas variáveis mensuráveis são verificados no que diz respeito aos limites de tolerância que os mesmos podem assumir. Caso tais valores não sejam respeitados, os alarmes pertinentes são gerados para o operador.

Função de proteção automática

No caso do processo atingir um estado perigoso, a função de proteção inicia automaticamente ações de reação apropriadas que implica em uma alteração controlada dos valores das variáveis mensuradas de forma a estabilizar a situação, levando o processo para uma “área” segura.

As duas funções de supervisão descritas acima são clássicas e as mais encontradas na área de supervisão de processos. Suas grandes vantagens são a simplicidade e a confiabilidade. Tão somente verificar o valor limite do conjunto das variáveis relevantes de um processo é a estratégia usada frequentemente quando se trabalha com processos que operem em um estado estacionário. Entretanto, tais soluções têm seus limites: elas somente são eficazes após uma alteração grande, intensa, notável e permanente das características do processo, ou seja, após

uma alteração catastrófica de um subconjunto das variáveis, de modo que essa alteração se mantenha no tempo, ou uma alteração crescente e gradual que transcorra por tempo relativamente longo. Logo, fica claro que tais funções não são eficientes para processos que mudem rapidamente e com grande frequência.

O desempenho do processo depende do estado dos dispositivos de controle, a saber, sensores ou instrumentos (para ler os dados que recebe do processo) e atuadores (para executar os comandos emitidos). Obviamente, toda situação anormal (ou falha), nestes dispositivos ou na própria planta, pode afetar o desempenho do processo. Se o sistema de controle eventualmente esconder um mau funcionamento ou apresentar um defeito abrupto, então a falha do controle será inevitável. Situações como a presença de uma válvula furada, escapamento nas tubulações ou leitura incorreta dos sensores conduz à perda da eficácia no controle. Com respeito aos diferentes elementos em que as falhas podem ocorrer, é possível estabelecer uma distinção entre a detecção de falhas nos instrumentos (*instrument fault detection* ou IFD), a detecção de falhas no atuador (*actuator fault detection* ou AFD) e a detecção de falhas nos componentes (*component fault detection* ou CFD).

Desta forma, surge uma terceira função de supervisão: A função de diagnóstico para supervisão de falhas. Tal função é necessária quando os processos têm as seguintes exigências:

- Detecção adiantada de falhas com evolução de comportamento incipiente no tempo.
- Detecção de falhas necessária também no atuador e não apenas nos componentes do processo.
- Supervisão dos processos em estados transientes.

A comunidade FDI lança mão da utilização de estimadores (*estimators*) para encontrar a estimativa do estado dos elementos que compõem o processo, mediante as medidas disponíveis. A diferença entre as medidas reais e os valores preditos é chamada de resíduo (*residuals*). Os resíduos grandes indicam discrepância entre o modelo representado pelo estimador e o processo real. Isto pode ser causado por alguma falha referente aos elementos que compõem o processo. Os métodos analíticos de redundância, descritos na seção seguinte, usam os resíduos para a detecção e o diagnóstico de falhas.

6.2.1 Aplicação da abordagem FDI no exemplo do *polybox*

A partir de agora será apresentado o formalismo da abordagem FDI, tendo como pano de fundo a utilização do *polybox*. Desta forma, as definições e conceitos do modelo da comunidade FDI serão apresentados, seguidos imediatamente pela sua aplicação no exemplo do *polybox*.

Definição 1.1: Um Modelo do Sistema SM (*System Model*) na abordagem FDI é definido pela dupla:

$$SM = (BM, OM) \quad (1)$$

Onde BM é o Modelo Comportamental do Sistema (*Behavioral Model*) composto por conjunto de relações que representam o comportamento do Sistema. Já OM é definido pela relação $OM \subset X \times O$ e representa o Modelo de Observações (*Observation Model*) do Sistema, o conjunto de observações passíveis de serem realizadas, onde X é o conjunto de todas as variáveis do Sistema e O é o conjunto das variáveis do Sistema que são observáveis por meio de sensores.

Exemplo: No *polybox*, os componentes elementares são os multiplicadores M_1, M_2, M_3 e os somadores A_1 e A_2 mais um conjunto de sensores. A comunidade FDI não declara nominalmente os sensores em seu modelo. Sua presença ocorre de maneira implícita pelo conjunto O. Para o *polybox*, os conjuntos O e X do exemplo assume a seguinte plástica:

$$X = \{a, b, c, d, e, f, g, x, y, z\}$$

$$O = \{a_{obs}, b_{obs}, c_{obs}, d_{obs}, e_{obs}, f_{obs}, g_{obs}\}$$

Temos, portanto, que o SM para o *polybox* é dado por:

$$SM_{polybox} = (BM_{polybox}, OM_{polybox})$$

Para:

Modelo de comportamental ($BM_{polybox}$)

RM1: $x = a * c$

RM2: $y = b * d$

RM3: $z = c * e$

RA1: $f = x + y$

$$\mathbf{RA2}: g = y + z$$

Modelo de observações ($OM_{polybox}$)

$$\mathbf{RSa}: a = a_{obs}$$

$$\mathbf{RSb}: b = b_{obs}$$

$$\mathbf{RSc}: c = c_{obs}$$

$$\mathbf{RSd}: d = d_{obs}$$

$$\mathbf{RSe}: e = e_{obs}$$

$$\mathbf{RSf}: f = f_{obs}$$

$$\mathbf{RSg}: g = g_{obs}$$

Dessa forma temos a descrição formal para o *polybox* a partir do paradigma da comunidade FDI.

Definição 1.2: O Problema Diagnóstico para a comunidade FDI é definido pela tripla:

$$DP = (SM, OBS, F) \quad (2)$$

Onde SM é o Modelo do Sistema, e OBS constitui-se no conjunto:

$$OBS = (v_{1obs} = val_1, v_{2obs} = val_2, \dots, v_{nobs} = val_n) \quad (3)$$

Denominado conjunto das observações (*Observation Set*) que atribuem valores às variáveis observáveis do Sistema, onde $v_{nobs} \in O$ e val_n pertence ao domínio de v_{nobs} .

F é o conjunto de falhas definido por $F = 2^{SF}$ sendo SF definido como o conjunto de falhas simples.

Exemplo: No *polybox*, o conjunto de falhas simples é composto por $SF_{polybox} = \{F_{M1}, F_{M2}, F_{M3}, F_{A1}, F_{A2}\}$.

Definição 1.3: Dada a aplicação binária

$$s: SM \times V \rightarrow \{0,1\} \quad (4)$$

Onde V é definido como o conjunto de todas as variáveis do Sistema, de maneira que:

$$V = X \cup O \text{ e } s(\text{rel}, v) = 1 \Leftrightarrow v \subset \text{rel}.$$

A *Estrutura do Sistema* é definida pelo conjunto de pares (rel, v) para o qual $s(rel, v) = 1$. Uma Estrutura do Sistema st pode ser representada por um grafo bipartido $G = (SM \cup V, A)$. Neste grafo, os nós são relações $rel_i \subset SM$ ou variáveis $v_j \subset V$. A é o conjunto das arestas do grafo que seguem a seguinte lei de formação:

$$\exists a_{ij} \in A \Leftrightarrow s(rel_i, v_j) = 1$$

Definição 1.4: Uma Relação Analítica de Redundância ARR (*Analytical Redundancy Relation*) é a restrição deduzida do Modelo do Sistema e que contém somente variáveis observáveis. ARRs são usadas para verificar a consistência das observações em relação ao SM. O âmago da abordagem FDI é que um ARR é a expressão simbólica em termos das variáveis do conjunto O, obtidas simbolicamente eliminando as variáveis inobserváveis de X presente nas relações definidas no Modelo do Sistema. Um SM pode derivar uma ou mais ARRs e o processo de derivação se dá a partir da Estrutura do Sistema como será visto a seguir.

Exemplo: Fica bem mais claro seguir o processo de derivação das ARRs a partir do exemplo. O primeiro passo é representar o grafo bipartido da Estrutura do Sistema do *polybox* pela matriz de *SM-incidência* (CARPANZANO, 1998).

A matriz de *SM-incidência* é definida como:

$$ISM[ism_{ij}] \Rightarrow v_j \in rel_i \Leftrightarrow ism_{ij} = 1$$

Onde os elementos de índice j representam as variáveis (observáveis ou inobserváveis) contidas no SM e os elementos possuidores de índice “i” representam as relações definidas no SM. A representação tabular da matriz de *SM-incidência* está representada na tabela (1) abaixo.

	a	b	c	d	e	f	g	x	y	z	a_{obs}	b_{obs}	c_{obs}	d_{obs}	e_{obs}	f_{obs}	g_{obs}	
RM1	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
RM2	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
RM3	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
RA1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0
RA2	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
RSa	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
RSb	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RSc	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
RSd	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0

RSe	0	0	0	0	1	0	0	0	0	00	0	0	0	0	1	0	0
RSf	0	0	0	0	0	1	0	00	0	0	0	0	0	0	0	1	0
RSg	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

Tabela 1 - Representação tabular da matriz de SM-incidência.

A etapa seguinte consiste em gerar a matriz de *matching* CM_x , matriz essa derivada a partir da matriz de SM-incidência, onde:

$$CM_x[cm_{ij}] \Rightarrow \{ v_j \subset rel_i \Leftrightarrow cm_{ij} = -1 \mid (v_j \in O \wedge rel_i \in OM) \vee (v_j \notin O \wedge rel_i \in BM) \}$$

Para o *polybox*, a matriz $CM_{x_{polybox}}$ resultante é mostrada na figura (7) em forma de tabela:

	a	b	c	d	e	f	g	x	y	z
RM1	0	0	0	0	0	0	0	-1	0	0
RM2	0	0	0	0	0	0	0	0	-1	0
RM3	0	0	0	0	0	0	0	0	0	-1
RA1	0	0	0	0	0	0	0	0	0	0
RA2	0	0	0	0	0	0	0	0	0	0
RSa	-1	0	0	0	0	0	0	0	0	0
RSb	0	-1	0	0	0	0	0	0	0	0
RSc	0	0	-1	0	0	0	0	0	0	0
RSd	0	0	0	-1	0	0	0	0	0	0
RSe	0	0	0	0	-1	0	0	0	0	0
RSf	0	0	0	0	0	-1	0	0	0	0
RSg	0	0	0	0	0	0	-1	0	0	0

Figura 7

A matriz de *matching* CM_x gerada é utilizada no estabelecimento do conjunto RD das relações redundantes de SM. O conjunto RD é definido como se segue:

$$RD = \{ rel_i \mid (rel_i \in (cm_{ij} \neq -1)) \wedge (rel_i \in BM) \}$$

De forma que para o Sistema *polybox* temos:

$$RD_{polybox} = (RA1, RA2)$$

Na figura (8) temos, destacado em cinza, as relações $rel_{i_{polybox}} \subset CM_{X_{polybox}}$ que são os elementos de $RD_{polybox}$:

	a	b	c	d	e	f	g	x	y	z
RM	0	0	0	0	0	0	0	-1	0	0
RM	0	0	0	0	0	0	0	0	-1	0
RM	0	0	0	0	0	0	0	0	0	-1
RA1	0	0	0	0	0	0	0	0	0	0
RA2	0	0	0	0	0	0	0	0	0	0
RSa	-1	0	0	0	0	0	0	0	0	0
RSb	0	-1	0	0	0	0	0	0	0	0
RSc	0	0	-	0	0	0	0	0	0	0
RSd	0	0	0	-1	0	0	0	0	0	0
RSe	0	0	0	0	-1	0	0	0	0	0
RSf	0	0	0	0	0	-1	0	0	0	0
RSg	0	0	0	0	0	0	-1	0	0	0

Figura 8

Tendo em mãos o conjunto RD derivado a partir do SM, obter o conjunto ARR das Relações Analíticas de Redundância se faz em duas subetapas. A primeira subetapa consiste em gerar um elemento em ARR para cada elemento de RD.

Finalmente, podemos ilustrar a determinação do conjunto, o conjunto de todas as ARRs para o exemplo do *polybox*.

Subetapa 1. Tomar os elementos do conjunto $RD_{polybox}$:

$$\mathbf{RA1: } f = x + y$$

$$\mathbf{RA2: } g = y + z$$

Onde:

$$x = a * c \quad \mathbf{(RM1)}$$

$$y = b * d \quad \mathbf{(RM2)}$$

$$z = c * e \quad \mathbf{(RM3)}$$

De forma que, assumindo que os sensores que fornecem o conjunto de observações não falham, teremos as seguintes expressões através da substituição de **(RM1)**, **(RM2)** e **(RM3)** em **RA1** e **RA2**:

$$\mathbf{RA1}: f = a * c + b * d$$

$$\mathbf{RA2}: g = b * d + c * e$$

Assim, derivamos os *ARR elementares* do conjunto ARR_{polibox} , pois:

$$ARR1_{\text{polibox}} : r1 = 0 \Rightarrow r1 \equiv f - a * c + b * d$$

$$ARR2_{\text{polibox}} : r2 = 0 \Rightarrow r2 \equiv g - b * d + c * e$$

Podemos derivar mais um elemento para o conjunto ARR_{polibox} a partir da combinação linear dos dois primeiros elementos. Portanto, tomando a seguinte combinação:

$$ARR3_{\text{polibox}} : r3 = 0 \Rightarrow r3 \equiv r1 - r2$$

Teremos:

$$ARR3_{\text{polibox}} : r3 = 0 \Rightarrow r3 \equiv r1 - r2$$

De forma trivial, fazemos as substituições:

$$ARR3_{\text{polibox}} : r3 = 0 \Rightarrow r3 \equiv (f - a * c + b * d) - (g - b * d + c * e)$$

$$ARR3_{\text{polibox}} : r3 = 0 \Rightarrow r3 \equiv f - g - a * c + c * e$$

Definição 1.5: Dado o conjunto ARR das Relação Analíticas de Redundância e o conjunto F de falhas citado na **Definição 2.2**, uma Assinatura de Falha FS_j é definida como se segue:

$$\forall F_j \in F \Rightarrow FS_j = (s_{1j}, \dots, s_{nj}), \text{ com } |ARR| = n$$

Onde:

$$s_{ij} = 1 \Leftrightarrow \text{Se o componente que apresenta a falha } F_j \text{ estiver presente em } ARR_i$$

Note-se que a declaração explícita dos componentes (A1,A2,M1,M2,M3) não faz parte, de maneira alguma, da definição de sistema na modelagem FDI. Desta forma, é necessário analisar mais detalhadamente a relação entre F_j e ARR_i

O fato de s_{ij} assumir valor diferente de zero significa que a ocorrência de F_j (dada, até aqui, de maneira descritiva e sem a utilização de elementos explícitos do modelo) afeta o resultado da ARR_i , da forma como segue:

$$ARR_i : r_i \neq 0 \Rightarrow F_j$$

Cabe perceber que esta interpretação supõe que a ocorrência de F_j seja observável implicitamente pelo resultado da ARR_i . Portanto, é possível incluir, a partir daí e de maneira indireta, os componentes que fazem parte do sistema diagnosticado pela associação das falhas com o conjunto F e os elementos do conjunto O das variáveis observáveis, contidos nas ARR's.

Exemplo: O *polybox* é constituído pelos seguintes componentes : A1,A2,M1,M2,M3. A associação dessa lista com os conjuntos $ARR_{polybox}$ e $F_{polybox}$ produz o conjunto $F_{polybox}$ rotulado em função dos componentes do Sistema, como segue:

$$ARR1_{polybox} : r1 \neq 0 \Rightarrow F_{A1_{polybox}}, F_{M1_{polybox}}, F_{M2_{polybox}}$$

$$ARR2_{polybox} : r2 \neq 0 \Rightarrow F_{A2_{polybox}}, F_{M2_{polybox}}, F_{M3_{polybox}}$$

$$ARR3_{polybox} : r3 \neq 0 \Rightarrow F_{A1_{polybox}}, F_{A2_{polybox}}, F_{M1_{polybox}}, F_{M3_{polybox}}$$

Definição 1.6: Ao conjunto FS de todas as Assinaturas de Falha, com suas respectivas falhas rotuladas em relação aos componentes do Sistema é dado o nome de Matriz de Assinatura (MA).

Exemplo: A Matriz $MS_{polybox}$ para o *polybox* é dada abaixo.

$$\begin{pmatrix} \times & F_{A1} & F_{A2} & F_{M1} & F_{M2} & F_{M3} \\ ARR1 & 1 & 0 & 1 & 1 & 0 \\ ARR2 & 0 & 1 & 0 & 1 & 1 \\ ARR3 & 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$

Definição 1.7: Uma assinatura de Observação OS é definida como se segue:

$$OS = (os_1, \dots, os_n) \text{ com } |ARR| = n$$

Onde:

$$os_i = 0 \Rightarrow V(r_i, OBS) = 0$$

$$os_i = 1 \Rightarrow V(r_i, OBS) = 1$$

De maneira que a função binária V é definida da seguinte forma:

$$OBS \Rightarrow r_i = 0 \Leftrightarrow V = 0$$

$$OBS \Rightarrow r_i \neq 0 \Leftrightarrow V = 1$$

Para os valores particulares do conjunto OBS em determinado instante, o valor assumido pela função V pode ser 0 ou o 1, dependendo de o valor assumido pelo resíduo de ARR ser diferente de zero.

Exemplo: Para duas observações hipotéticas, vamos analisar os possíveis estados assumidos por: $OS_{polybox}$

$$OBS(1)_{polybox} : \text{Para a situação em que } f = 10 \text{ e } g = 10, \text{ teremos } OS_{polybox} = (1, 1, 0)$$

$$OBS(2)_{polybox} : \text{Para a situação em que } f = 10 \text{ e } g = 14, \text{ teremos } OS_{polybox} = (1, 1, 1)$$

Definição 1.8: Uma *função de consistência* CS para a assinatura de Observação OS sujeita à observação OBS é definida como se segue:

$$CS(FS_j, OS(OBS)) = 1 \Leftrightarrow \forall i, \exists F_j : (os_i \in OS) = (s_{ij} \in FS_j) \text{ com } F_j \in F$$

$$CS(FS_j, OS(OBS)) = 0 \Leftrightarrow \text{Caso contrário}$$

Quando $CS(FS_j, OS(OBS)) = 1$ dizemos que $OS(OBS)$ está consistente com FS_j .

Definição 1.9: Para o caso de falhas simples, um Conjunto de Diagnóstico DS (*Diagnosis Set*) é composto pelas falhas cujas assinaturas são consistentes com a assinatura de Observação, de tal forma que, para determinado conjunto OBS de observações, teremos:

$$DS = (F_1, \dots, F_j) \Leftrightarrow \begin{cases} CS(FS_1, OS(OBS)) = 1 \\ \dots \\ CS(FS_j, OS(OBS)) = 1 \end{cases}$$

Exemplo: Daremos a seguir, para o *polybox*, um exemplo de quatro conjuntos de Assinaturas de Observação e seus respectivos conjuntos de Diagnósticos.

OS1: $(os_1, os_2, os_3) = (0,0,0)$; $\mathbf{DS} = \{ \}$ \Rightarrow (sem falhas)

OS2: $(os_1, os_2, os_3) = (0,1,1)$; $\mathbf{DS} = (F_{A2_{polybox}}, F_{M3_{polybox}})$ \Rightarrow (A2 falho ou M3 falho)

OS3: $(os_1, os_2, os_3) = (1,0,1)$; $\mathbf{DS} = (F_{A1_{polybox}}, F_{M1_{polybox}})$ \Rightarrow (A1 falho ou M1 falho)

OS4: $(os_1, os_2, os_3) = (1,1,0)$; $\mathbf{DS} = (F_{M2_{polybox}})$ \Rightarrow (M2 falho)

6.3 A abordagem DX

A comunidade DX tem fortes vínculos com o Departamento de Astronáutica Americano (National Aeronautics and Space Administration, ou NASA) e suas técnicas foram usadas no sistema embarcado LIVINGSTONE para diagnosticar veículos espaciais como a sonda espacial *Deep Space One* (WILLIAMS e NAYAK, 1996). O estudo do diagnóstico de falha, do ponto de vista da agência espacial americana, integra o esforço pela busca cada vez maior de autonomia das sondas espaciais interplanetárias (BERNARD *et al.*, 1998).

Em inteligência artificial, as primeiras pesquisas na área de diagnóstico tiveram origem no estudo da Engenharia Ontológica, em situações em que é inserido o processo de alteração estrutural como elemento de uma ontologia. Os primeiros trabalhos relacionados ao diagnóstico foram de DAVIS (1984) e GENESERETH (1984). Os primeiros estudos do diagnóstico baseado em modelos pela comunidade de IA foram aplicadas com sucesso no diagnóstico de falhas em circuitos eletrônicos.

Dos primeiros sistemas desenvolvidos dentro do paradigma DX, o GDE (*General Diagnosis Engine*) (KLEER, 1986) é até hoje considerada a implementação mais bem sucedida e conhecida (PULIDO e GONZALEZ, 2004).

O estudo de circuitos eletrônicos rendeu muita reflexão acerca da natureza do processo diagnóstico na comunidade de IA, mas, o mais importante, que foi dessa pesquisa que surgiu uma das primeiras formalizações da teoria diagnóstica: a Teoria do Diagnóstico Baseado em Consistência (*Consistency Based Diagnosis Theory*) como proposto por (REITER, 1987). Tal modelo da foi aplicado com sucesso em grande variedade de domínios diferentes, tais como a Medicina (LAFON *et al.*, 1998), indústria automobilística (SACHENBACHER *et al.*, 2000) (STRUSS, *et al.*, 2000) e nuclear (MOSTERMAN, 1997). O impacto deste trabalho foi tão grande

que, até os dias de hoje, muitos trabalhos dentro da comunidade DX usam o termo Diagnóstico Baseado em Consistência no lugar de RBM.

Neste trabalho, é descrito um algoritmo que, dado um conflito entre as observações e o modelo do equipamento diagnosticado, calcula o conjunto mínimo de diagnóstico, isto é um conjunto dos componentes falhos. Nos trabalhos posteriores de (KLEER, 1992) e (DRESSLER, 1996) essa formalização foi refinada.

6.3.1 Aplicação da abordagem DX no exemplo do polybox

Definição 2.1: Um modelo do Sistema (*System Model*) é definido como se segue:

$$SM = (SD, COMPS)$$

Onde COMPS é o conjunto finito de componentes (*COMPonentS*) que constituem o Sistema em análise. SD consiste na descrição do Sistema (*System Description*) e é composto pelo conjunto de proposições em lógica de primeira ordem. COMPS é o conjunto finito de componentes que compõem o Sistema em análise. Baseado. A descrição do SD é dividida em três tópicos, a saber:

Regras do domínio

O objetivo das Regras do Domínio é apresentar as classes de “objetos” constituintes do equipamento diagnosticado e descrever seus comportamentos em função dos elementos que constituem cada objeto.

Uma Regra do Domínio tem o seguinte formato:

$$COMP - FUNC(x) \wedge COMP - ESTR(x) \Rightarrow COMP - CONST(x)$$

Onde:

$$x \in COMPS.$$

COMP-FUNC(x): É o Comportamento Funcional e especifica a classe taxonômica para o componente genérico x em função da função desempenhada por x descrita por COMP-CONST(x).

COMP-ESTR(x): É o Comportamento Estrutural, representa todos os estados que o componente x pode assumir. O conceito de estados não é formalmente definido para a comunidade DX, sendo expresso nas regras do Domínio como mais um predicado.

COMP-CONST(x): É o Comportamento das Partes Constituintes e tem como função declarar os principais elementos do ponto de vista diagnóstico que formam o componente x bem como descrever o relacionamento entre essas partes para gerar o Comportamento Funcional.

Especificação do sistema

A especificação do sistema enumera as instâncias dos “objetos” cujas classes foram definidas nas Regras do Domínio. Neste ponto, é dado nome a cada instância de “objeto” criado e definida classe a que o objeto pertence. Tais instâncias constituem todos os componentes visíveis pela abstração concernente ao modelo dos componentes do equipamento diagnosticado passíveis de sofrerem falhas.

Relacionamentos

Segundo STRUSS (*et al.*, 2000), “[Em um Sistema] as instâncias das classes de objeto geram n-uplas de relacionamento”. O papel dos Relacionamentos é justamente descrever como as instâncias dos objetos interagem.

STRUSS (*et al.*, 2000) menciona ainda um quarto tópico denominado “Leis Básicas”. Este tópico não foi listado neste trabalho. Ele foi considerado alternativo, por ter função complementar à Especificação do Sistema em situações nas quais os comportamentos das classes de objetos necessitem de definições (leis físicas, por exemplo) que não sejam passíveis de serem comportadas pela estrutura definida nas regras do domínio, seja por sua extensão ou complexidade. As “Leis Básicas” serviriam, portanto, como espécie de “apêndice” da Especificação do Sistema.

Exemplo: A partir deste exemplo, analisaremos o *polybox* do ponto de vista da modelagem DX. Iniciamos com a declaração de SM para o *polybox* feita a seguir:

$$\begin{aligned}
SD_{\text{polybox}} = \{ & \\
& \text{ADD}(x) \wedge \neg \text{AB}(x) \Rightarrow \text{Output}(x) = \text{Input1}(x) + \text{Input2}(x), \\
& \text{MULT}(x) \wedge \neg \text{AB}(x) \Rightarrow \text{Output}(x) = \text{Input1}(x) \times \text{Input2}(x), \\
& \text{ADD}(A1), \\
& \text{ADD}(A2), \\
& \text{MULT}(M1), \\
& \text{MULT}(M2), \\
& \text{MULT}(M3), \\
& \text{Output}(M1)_{\text{polybox}} = \text{Input1}(A1)_{\text{polybox}}, \\
& \text{Output}(M2)_{\text{polybox}} = \text{Input2}(A1)_{\text{polybox}}, \\
& \text{Output}(M2)_{\text{polybox}} = \text{Input1}(A2)_{\text{polybox}}, \\
& \text{Output}(M3)_{\text{polybox}} = \text{Input2}(A2)_{\text{polybox}}, \\
& \left. \text{Input2}(M1)_{\text{polybox}} = \text{Input1}(M3)_{\text{polybox}} \right\}
\end{aligned}$$

$$\text{COMPS}_{\text{polybox}} = \{A1, A2, M1, M2, M3\}$$

Logo:

$$SM_{\text{polybox}} = (SD_{\text{polybox}}, \text{COMPS}_{\text{polybox}})$$

Logo:

Desta forma, dentro do paradigma da comunidade DX, temos a descrição formal do *polybox*. Olhando para o exemplo acima, podemos discriminar os elementos que compõem a descrição do SD. Primeiro, vamos enumerar as linhas na declaração do SD_{polybox} :

$SD_{polybox} = \{$	
$ADD(x) \wedge \neg AB(x) \Rightarrow Output(x) = Input1(x) + Input2(x),$	LINHA1
$MULT(x) \wedge \neg AB(x) \Rightarrow Output(x) = Input1(x) \times Input2(x),$	LINHA2
$ADD(A1),$	LINHA3
$ADD(A2),$	LINHA4
$MULT(M1),$	LINHA5
$MULT(M2),$	LINHA6
$MULT(M3),$	LINHA7
$Output(M1)_{polybox} = Input1(A1)_{polybox},$	LINHA8
$Output(M2)_{polybox} = Input2(A1)_{polybox},$	LINHA9
$Output(M2)_{polybox} = Input1(A2)_{polybox},$	LINHA10
$Output(M3)_{polybox} = Input2(A2)_{polybox},$	LINHA11
$Input2(M1)_{polybox} = Input1(M3)_{polybox}$	LINHA12
$\}$	

Agora vamos discriminar cada linha em função dos tópicos que constituem um SD:

Regras do domínio:	LINHA1 e LINHA2
Especificação do sistema	LINHA3, LINHA4, LINHA5, LINHA6 e LINHA7
Relacionamentos	LINHA8, LINHA9, LINHA10, LINHA11 e LINHA12

$AB(x)$ utilizado na declaração de SM é o predicado do Comportamento Estrutural que significa “comportamento anormal (*abnormal*) do componente x ”, onde $x \in COMPS$.

Definição 2.2: Uma observação OBS (*OBServation*) é o conjunto de predicados de lógica de primeira ordem resultantes da série de mensurações realizadas no Sistema em análise.

Exemplo: Suponha-se que o *polybox* esteja fornecendo as seguintes mensurações para seu conjunto de observações:

Conjunto de valores para as entradas:

$$a = 2$$

$$b = 2$$

$$c = 3$$

$$d = 3$$

$$e = 2$$

Conjunto de valores para as saídas:

$$f = 10$$

$$g = 12$$

O conjunto das observações é representado por:

$$\begin{aligned} \text{OBS}_{\text{polybox}} = \{ \\ \text{Input1(M1)}_{\text{polybox}} = 2, \\ \text{Input2(M1)}_{\text{polybox}} = 3, \\ \text{Input1(M2)}_{\text{polybox}} = 2, \\ \text{Input2(M2)}_{\text{polybox}} = 3, \\ \text{Input2(M3)}_{\text{polybox}} = 2, \\ \text{Output(A1)}_{\text{polybox}} = 10, \\ \text{Output(A2)}_{\text{polybox}} = 12 \\ \} \end{aligned}$$

Definição 2.3: O Problema Diagnóstico para a comunidade DX é definido pela dupla:

$$DP = (SM, OBS) \quad (5)$$

Onde é o Modelo do Sistema e é uma observação.

Definição 2.4: O diagnóstico para DP é o conjunto de componentes $\Delta \subseteq COMPS$ de forma que:

$$SD \cup OBS \cup \{AB(x) \mid x \in \Delta\} \cup \{\neg AB(y) \mid y \in COMPS - \Delta\}$$

é consistente.

Definição 2.5: O conjunto diagnóstico mínimo é o conjunto Δ tal que:

$$\forall \Delta' \subset \Delta \Rightarrow \Delta' \text{ Não é diagnóstico.}$$

O diagnóstico é o subconjunto dos componentes do equipamento diagnosticado em que se verifica, por observação, a presença de um comportamento anormal (através do predicado $AB(x)$). Na determinação de Δ , sempre se procurará o conjunto diagnóstico mínimo em respeito ao princípio da parcimônia. Uma maneira computacional ingênua de se obter os diagnósticos baseados na definição acima é através de um algoritmo que gerasse os teste de

consistência, nos quais os subconjuntos dos componentes fossem selecionados, gerando os conjuntos diagnósticos mínimos.

O obstáculo para essa abordagem consiste no fato de que, segundo VATAN (*et al.*, 2002), ela resolve o problema em tempo não-polinomial sendo, portanto, ineficiente. Entretanto, a comunidade DX mostra que, modelando a solução do problema diagnóstico com auxílio dos *hitting set* e conjuntos R-conflito, algoritmos com melhor desempenho podem ser alcançados.

Definição 2.6: O R-conflito para (SM, OBS) é o conjunto de componentes $C \subseteq COMPS$ tal que:

$$SD \cup OBS \Rightarrow \exists c \in C \mid AB(c)$$

No conjunto R-conflito, pelo menos dois componentes apresentam comportamento anormal (indicado pelo predicado AB) a fim de corroborar com as observações. A principal função dos conjuntos R-Conflito é obter o conjunto diagnóstico mínimo para o S-Paciente em situação de anormalidade.

Exemplo: A tabela apresentada na figura (8) mostra, de forma semelhante ao último exemplo apresentado, uma série de três mensurações contidas no conjunto $OBS_{polybox}$ em situações a qual o *polybox* apresente-se em situação de falha. Na segunda coluna, temos os conjuntos R-Conflito associados a essas observações:

$OBS_{polybox}$	R-Conflito
Entrada: a = 2, b = 2, c = 3, d = 3, e = 2 Saída: f = 10, g = 12	$C1_{polybox} = \{A1, M1, M2\}$ $C2_{polybox} = \{A1, A2, M1, M3\}$
Entrada: a = 2, b = 2, c = 3, d = 3, e = 2 Saída: f = 10, g = 10	$C1_{polybox} = \{A1, M1, M2\}$ $C2_{polybox} = \{A2, M2, M3\}$
Entrada: a = 2, b = 2, c = 3, d = 3, e = 2 Saída: f = 10, g = 14	$C1_{polybox} = \{A2, M2, M3\}$ $C2_{polybox} = \{A1, M1, M2\}$ $C3_{polybox} = \{A1, A2, M1, M3\}$

Figura 8

Definição 2.7: Um R-conflito mínimo é o conjunto R-conflito que não possui como elemento qualquer outro conjunto R-conflito.

Com o uso da definição de conjunto R-Conflito mínimo é possível caracterizar os conjuntos que proporcionem base computacional adequada para obtermos processo eficiente para a

geração de diagnósticos mínimos. Esta caracterização é baseada na definição de corte transversal mínimo (*hitting set*) dada a seguir:

Definição 2.8: Dado um conjunto finito E , e uma coleção finita C de forma que:

$$C = \{S_1, \dots, S_j\}; \text{ onde } S_w \text{ é um subconjunto de } E, \text{ para } 1 \leq w \leq j$$

o corte transversal mínimo é o subconjunto A de tamanho mínimo com as seguintes propriedades:

$$(A \subseteq E) \wedge (A \cap S_w \neq \{\}) \text{ para } 1 \leq w \leq j \quad (6)$$

De modo que o corte transversal intercepte todos os conjuntos da coleção. Obviamente, para o corte transversal ser mínimo é necessário e suficiente que nenhum dos subconjuntos próprios (ou seja, qualquer subconjunto do próprio corte transversal) seja um corte transversal para C .

Exemplo: Será fornecido aqui exemplo conceitual à parte da seqüência adotada até agora do *polybox*. Embora não tenha nenhuma relação com o *polybox* e consista em exemplo genérico, sua simplicidade ajuda a esclarecer o conceito de corte transversal mínimo, de forma a cooperar quando tal conceito for aplicado no exemplo do *polybox* e no contexto do estudo de diagnósticos de falha.

Considere o conjunto E_{exemplo} declarado abaixo:

$$E_{\text{exemplo}} = \{1,2,3,4,5,6,7,8,9\}$$

E os seguintes subconjuntos dele derivados:

$$S1_{\text{exemplo}} = \{7,8\}$$

$$S2_{\text{exemplo}} = \{5,6\}$$

$$S3_{\text{exemplo}} = \{4,5\}$$

$$S4_{\text{exemplo}} = \{8,9\}$$

$$\text{De forma que: } C_{\text{exemplo}} = \{S1_{\text{exemplo}}, S2_{\text{exemplo}}, S3_{\text{exemplo}}, S4_{\text{exemplo}}\}$$

Para o conjunto C_{exemplo} definido acima temos que $A_{\text{exemplo}} = \{5,8\}$ é corte transversal mínimo de acordo com (6).

O estudo de cortes transversais mínimos é dos problemas-chave em combinatória segundo VATAN (*et al.*, 2002) e BERGE (1989). Já a sua conexão com a modelagem da comunidade DX e sua importância no processo de determinação dos conjuntos diagnósticos mínimos será mostrada na definição a seguir.

Definição 2.9: Δ é diagnóstico mínimo para (SD, COMPS, OBS) se e somente se é corte transversal mínimo para a coleção de R-Conflitos mínimos de (SD, COMPS, OBS).

Exemplo: A tabela apresentada na figura (9) é complementação da tabela da figura (8). De forma semelhante à tabela anterior, temos série de três mensurações contidas no conjunto $OBS_{polybox}$ em situações nas quais o *polybox* apresente-se em situação de falha. Na segunda coluna, temos os conjuntos R-Conflito associados a essas observações e na terceira coluna temos os conjuntos $\Delta_{polybox}$ mínimos.

A associação entre as três colunas segue a seguinte lógica: os diagnósticos mínimos da terceira coluna são obtidos computando os cortes transversais mínimos para a coleção de R-Conflitos mínimo da segunda coluna, coleção essa gerada pelas observações em $OBS_{polybox}$ (primeira coluna).

OBS_{polybox}	R-Conflito	Δ_{polibox} mínimos
Entrada: a=2 b=2 c=3 d=3 e=2 Saída: f=10 g=12	$C1_{\text{polybox}} = \{A1, M1, M2\}$ $C2_{\text{polybox}} = \{A1, A2, M1, M3\}$	$\Delta1_{\text{polibox}} = \{A1\}$ $\Delta2_{\text{polibox}} = \{M1\}$ $\Delta3_{\text{polibox}} = \{A2, M2\}$ $\Delta4_{\text{polibox}} = \{M2, M3\}$
Entrada: a=2 b=2 c=3 d=3 e=2 Saída: f=10 g=10	$C1_{\text{polybox}} = \{A1, M1, M2\}$ $C2_{\text{polybox}} = \{A2, M2, M3\}$	$\Delta1_{\text{polibox}} = \{M2\}$ $\Delta2_{\text{polibox}} = \{A1, A2\}$ $\Delta3_{\text{polibox}} = \{A1, M3\}$ $\Delta4_{\text{polibox}} = \{A2, M1\}$ $\Delta5_{\text{polibox}} = \{M1, M3\}$
Entrada: a=2 b=2 c=3 d=3 e=2 Saída: f=10 g=14	$C1_{\text{polybox}} = \{A2, M2, M3\}$ $C2_{\text{polybox}} = \{A1, M1, M2\}$ $C3_{\text{polybox}} = \{A1, A2, M1, M3\}$	$\Delta1_{\text{polibox}} = \{A1, A2\}$ $\Delta2_{\text{polibox}} = \{A1, M2\}$ $\Delta3_{\text{polibox}} = \{A1, M3\}$ $\Delta4_{\text{polibox}} = \{A2, M1\}$ $\Delta5_{\text{polibox}} = \{A2, M2\}$ $\Delta6_{\text{polibox}} = \{M1, M2\}$ $\Delta7_{\text{polibox}} = \{M1, M3\}$ $\Delta8_{\text{polibox}} = \{M2, M3\}$

Figura 9

7 O Raciocínio Baseado em Casos (RBC)

A abordagem de Raciocínio Baseado em Casos não é uma teoria de diagnóstico de falhas em si. Ela é definida como um paradigma de aprendizado e resolução de problemas. Porém, sua aplicabilidade se mostrou tão flexível, que sistemas baseados em casos foram desenvolvidos e validados para os mais diferentes domínios: negociações diplomáticas, arquitetura, direito e diagnóstico, não só clínico, mas também de falhas.

Os estudos na área de Raciocínio Baseado em Casos (RBC ou CBR, do inglês *Case Based Reasoning*) começaram no final da década de setenta. Seu ponto de partida foi justamente o estudo do raciocínio por analogia e por experiência, iniciado pelo trabalho de GICK e HOLYOAK (1980). Paralelamente ao estudo do raciocínio por analogia, o desenvolvimento do RBC teve seu avanço incrementado pelos estudos relacionados à memória, formação de conceitos e aprendizagem realizados por SCHANK e ABELSON (1977) bem como outros pesquisadores, como SMITH e ADAMS (1978). Tais estudos tinham forte caráter multidisciplinar e envolviam profissionais de áreas como Psicologia, Ciência da Computação e Filosofia. Para ilustrar bem o tom multidisciplinar destas pesquisas, foi exatamente uma pesquisadora pertencente ao grupo de Schank, Janet Kolodner, que desenvolveu o primeiro software a utilizar o conceito de RBC, o sistema chamado CYRUS (KOLODNER, 1983).

Conforme foi apresentado no capítulo três, muitas vezes não é possível derivar um diagnóstico por métodos abduativos. Desta forma, a abordagem RBC retoma uma característica interessante no modo como os médicos elaboram seus diagnósticos: a analogia e a utilização de experiências passadas. Daí seu sucesso no que diz respeito a aplicações de cunho diagnóstico.

Como foi visto no capítulo dois, o principal critério usado pelo médico para a seleção de hipóteses válidas em meio ao conjunto grande de hipóteses iniciais possíveis é sua experiência. Dessa forma, soluções encontradas e aplicadas em outros pacientes podem ser reaproveitadas com pequenas modificações. Sistemas que se utilizam da abordagem RBC usam exatamente esta estratégia: eles guardam histórico de todas as situações passadas, e o utilizam para tratar novas situações.

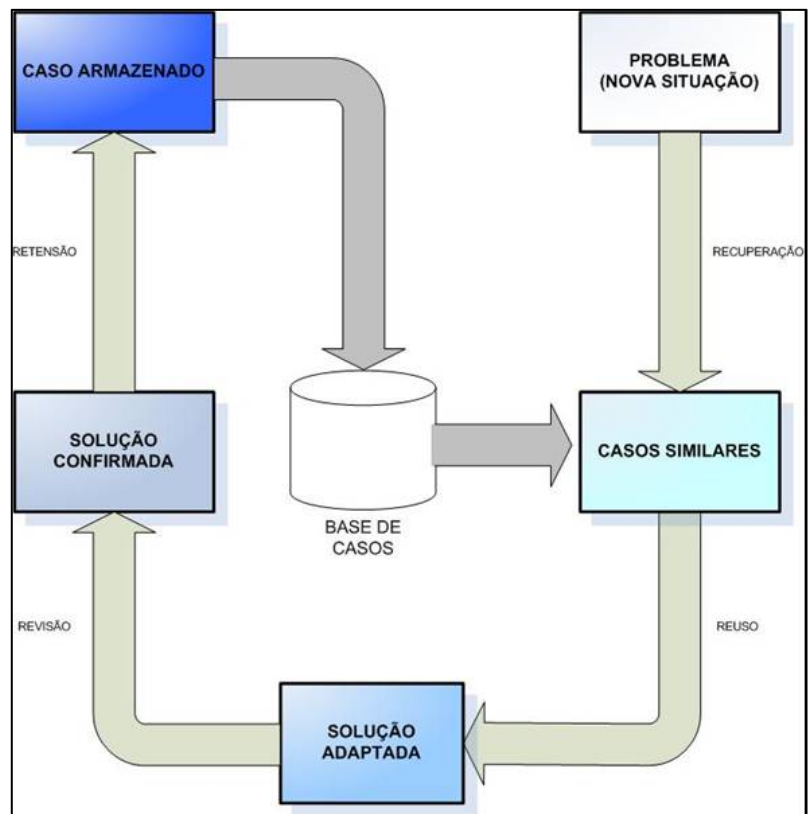
Para operacionalizar tal estratégia, utiliza-se o que foi definido como ciclo RBC, proposto e descrito por AAMONDT e PLAZA (1994). O ciclo RBC é composto de quatro etapas que são:

- Recuperação
- Reutilização
- Revisão
- Retenção

A dinâmica entre essas etapas está ilustrada na figura (9).

Figura 9 - O Ciclo RBC

Pelos termos empregados, e pela dinâmica, fica claro a semelhança entre o ciclo RBC e o raciocínio analógico. Os próximos itens descreverão cada uma das etapas do ciclo RBC em função de seus objetivos e características.



7.1 A etapa de recuperação

A etapa de recuperação pode ser entendida como um processo de busca, e provavelmente é o componente mais crítico do ciclo RBC. Ele determina como o sistema trabalha com sua “memória” e possui o custo computacional mais elevado para ser realizado.

O objetivo dessa etapa é encontrar, em meio a sua base de conhecimentos (histórico das situações), a solução mais útil para resolver o problema atual. Desta forma, o conjunto de hipóteses iniciais será composto de experiências passadas, descrevendo situações já resolvidas que representem as possíveis causas que levem à solução viável para a questão enfrentada.

O próximo passo é elaborar o mecanismo que utilize analogias para selecionar, entre todas as experiências armazenadas, aquela (ou aquelas) que pode ser reutilizada para resolver o problema atual. Para recuperar-se uma solução, é necessário, primeiramente, responder à pergunta: qual das experiências passadas mais se assemelha ao problema atual? Comparar o quadro atual com as experiências armazenadas é o processo para responder a essa pergunta; porém, tal comparação só é possível se as experiências forem formatadas e armazenadas de forma organizada. No paradigma RBC, o conhecimento é armazenado em forma de casos.

7.1.1 Definição de caso

Segundo KOLODNER (1993), um caso é definido como o fragmento de conhecimento contextualizado que representa a experiência que contém a lição passada fundamental para se conseguir alcançar determinado objetivo. Os casos são armazenados em uma biblioteca, indexada e organizada apropriadamente para facilitar a sua recuperação de forma eficiente. Tal biblioteca é chamada de Banco de Casos ou Base de Casos.

As bases de casos podem ser construídas usando critérios dependentes do domínio para estabelecer a estrutura mais eficiente de agrupamento de casos. Por exemplo, no sistema RBC de diagnóstico de falhas, o critério pode ser a divisão do equipamento diagnosticado em seus subsistemas, de forma de cada divisão tenha um conjunto de casos cujos diagnóstico estejam relacionados ao subsistema em particular. Dessa forma, vislumbrando a aplicação, para um sistema de diagnóstico de falhas em aeronaves teríamos como resultado da organização hierárquica da base de casos, um conjunto de casos relacionado aos aviônicos, outro conjunto de casos relacionados à parte hidráulica e assim por diante.

Segundo KOLODNER (1993), um caso representa tipicamente a descrição da situação (problema) com as experiências adquiridas (solução) durante sua resolução, sendo visto como essa associação dos dois conjuntos de informações: descrição do problema e respectiva solução. Desta definição, podem-se derivar os elementos que constituem o caso e que são, segundo WATSON e MARIR (1994):

- Descrição do Problema
- Descrição da solução e de seus resultados

Cada um desses elementos será descrito a seguir.

Descrição do problema

A descrição do problema consiste em uma declaração do estado do domínio do problema quando o caso ocorreu e os elementos que descrevem esse estado, além de conter a descrição do objetivo que deve ser alcançado para que o problema seja resolvido. A descrição dos elementos deve levar em conta a interação que possa existir entre eles, bem como um histórico de cada elemento.

Descrição da solução e dos seus resultados

A solução e os resultados gerados por sua aplicação podem ser descritos de várias formas dependendo da natureza do problema. Enquanto a solução pode ser apresentada na forma de algoritmo, um conjunto de justificativas, os resultados podem ser apresentados na forma de narrativa livre, ou usando informações quantitativas, como tabelas. De qualquer forma, os resultados sempre descreverão como ficou o estado do domínio do problema após a aplicação da solução. Em sistema de diagnóstico médico, por exemplo, poderia ser informado qual foi a reação do paciente e a alteração de seu quadro clínico em função da aplicação do tratamento (solução) referente ao caso, bem como reações alérgicas e outras que podem também ser apresentadas utilizando-se elementos gráficos ou referências a outros documentos.

Os elementos que compõem a descrição do problema e da solução são denominados atributos do caso. Olhando atentamente, pode-se perceber que um caso possui estrutura muito semelhante ao esquema explanatório visto no capítulo três. De forma semelhante ao caso, o esquema explanatório possui estrutura análoga à descrição do problema, constituída pela pergunta base. Possui também a descrição da solução, na combinação do padrão explanatório com a terapêutica.

7.1.2 Definição de similaridade

Ao elaborar a descrição do problema, o projetista de casos tem que se preocupar com as questões referentes à representação do caso, como ele será estruturado, implementado e qual será sua forma (por exemplo, se será representado através de uma classe, dentro do paradigma de orientação a objetos). Igualmente, o projetista tem que cuidar de questões relacionadas ao modo pelo qual o caso possa ser identificado como útil, e a maneira pela qual ele pode ser reutilizado. Essas decisões são extremamente dependentes do contexto do problema, porém um objetivo é sempre claro: os aspectos mais importantes do domínio do problema devem ser

destacados, rotulados e ordenados de maneira a criar um critério de escolha e recuperação dentro da biblioteca de casos. Dentro da abordagem RBC, esse critério é chamado de medida de similaridade.

A similaridade define como será verificada a analogia entre a situação atual e as experiências anteriores, sendo aplicado entre a descrição do problema atual e cada uma das experiências passadas que compõem o conjunto de hipóteses iniciais contidas na base de casos. Dependendo do critério adotado, a pesquisa pode retornar um resultado ou um conjunto de resultados, ordenados por grau de similaridade: da hipótese mais provável para explicar a situação atual até a menos provável. A escolha do critério depende do tipo do problema abordado, bem como a maneira de calcular a medida de similaridade depende dos elementos do caso que queremos verificar análogos.

Similaridade também significa utilidade. De acordo com GRESSE (2003), um caso armazenado na base de casos é similar ao problema atual se ele cumprir as seguintes metas:

- Permitir a solução correta e eficiente do problema atual de alguma forma.
- Evitar erros cometidos anteriormente.
- A solução deve ser a melhor possível dentro de uma meta de otimização. Tal meta é o cerne do critério de similaridade, e mede o quão similar é o caso selecionado com a situação presente.
- A solução, além de ser compreensível, deve ser reproduzível pelo usuário.

O conceito de similaridade é implementado pela já mencionada medida de similaridade. A medida de similaridade é nosso “motor analógico”, implementado por meio de um modelo matemático e/ou heurístico, definido para uma particular aplicação (diagnóstica, forense, etc.), e que efetivamente realiza o “julgamento” referente à semelhança entre dois casos. Para calcular determinada medida de similaridade definida, dispõe-se tão somente dos atributos do caso, que nesta situação, relembrando o capítulo os nossos observáveis. Quando utilizados para o cálculo da medida de similaridade, os atributos são chamados de índice.

É muito grande a importância que tem a escolha dos índices dentro do caso. O índice é o campo do caso que deve ser transformado em dado quantitativo e ser usado como elemento que compõe o cálculo da medida de similaridade usada pelo procedimento de recuperação para determinar o grau de similaridade entre a situação da entrada e os casos armazenados. De maneira geral, deve-se escolher como índices àqueles atributos que têm maior impacto em

carregar informações pertinentes ao caso, bem como ajudar no processo de discriminação entre casos diferentes. Os índices têm que ser escolhidos com cuidado pois, quanto maior o número utilizado, maior é o número de operações realizadas e, portanto, mais lenta a recuperação. Se os índices forem menos que os necessários, a precisão da recuperação irá se deteriorar.

Se os índices escolhidos forem modelados como dados booleanos (ligado ou desligado, existe ou não existe, etc.) então são chamados de índices simétricos, e a medida de similaridade é dita invariante. GRESSE (2003) esclarece que dados booleanos implicam em índices simétricos porque o fato da não - observação de um valor implica imediatamente na observação de seu oposto. Existem situações em isso não ocorre. Por exemplo, considere o seguinte:

“[Avião] fabricado pela empresa X.”

Tal índice é assimétrico. Para verificar esse fato, basta assumir a medida de similaridade que seja a comparação entre os índices. Caso os índices dos dois casos tenham o mesmo valor, esse valor é analisado.

Para duas instâncias do caso (Caso 1 e Caso 2) em que esse índice está definindo a situação:

Índice Caso 1: (Avião A fabricado pela empresa X) = 1

Índice Caso 2: (Avião B fabricado pela empresa X) = 1

implica necessariamente na similaridade dos dois casos, descontados todos os outros índices que possam existir para esses dois casos. Porém a seguinte situação:

Índice Caso 1: (Avião A fabricado pela empresa X) = 0

Índice Caso 2: (Avião B fabricado pela empresa X) = 0

não torna os dois casos similares, pois o fato dos aviões A e B de não possuírem a empresa X como fabricante, não implica que o fabricante dos dois aviões, qualquer que seja ele, vá ser o mesmo. Ocorre, portanto quebra de simetria, o valor ([Avião] fabricado pela empresa X) = 1 assume peso maior no que diz respeito à informação transmitida em relação à situação ([Avião] fabricado pela empresa X) = 0. Nesse caso definimos que a medida de similaridade é assimétrica. Quando a medida de similaridade é calculada entre os índices (como no caso do exemplo acima), ela é chamada de similaridade local. Quando uma medida de similaridade é

calculada mediante a utilização de todas as similaridades locais definidas, temos uma medida de similaridade global.

Existem várias heurísticas e modelos para se definir e calcular a medida de similaridade. Enumerar e descrever todas é tarefa que está além do escopo desse trabalho.

Após a definição do que seja caso, biblioteca, índice e similaridade, é possível passar para a descrição da etapa de recuperação. A etapa de recuperação, segundo GRESSE, (2003), é composta de três subtarefas:

- Elaboração de uma consulta
- Casamento
- Seleção

A seguir, será descrita cada uma delas.

Elaboração de uma consulta

Uma consulta é o conjunto por meio do qual os dados relevantes referentes ao problema atual são inseridos. Estes dados podem ser os próprios índices, atributos de casos ou outras informações pertinentes que, depois de tratadas, serão utilizadas no cálculo da medida de similaridade.

Casamento

O casamento é a seleção, pelo critério de similaridade, de um caso da Base de Casos em função dos dados inseridos na consulta.

Seleção

Aqui é selecionado o melhor casamento ou casamentos que satisfaçam totalmente o critério de similaridade em relação aos dados da consulta.

7.2 A etapa de reutilização

A primeira tarefa a ser realizada com a solução contida no caso recuperado dentre os melhores casamentos é verificar sua aplicabilidade ao problema novo. Se a situação nova for semelhante apenas parcialmente à solução encontrada, um processo conhecido como adaptação deve ser realizado para modificar o caso selecionado de maneira que esse possa ser reutilizado nova situação.

A reutilização é processo que requer cuidado. É necessário saber a priori o que pode ser adaptado e como. No exemplo da base de casos para diagnóstico em automóvel, dado acima, tome a seguinte situação: ocorreu novo problema descrito como “farol de milha não acende”. Todos os dados inseridos na consulta levaram à recuperação do caso “farol não acende” com a solução “troque o farol”. A utilização desta solução é pertinente ao novo caso, desde que seja adaptada em seu elemento principal (farol), tanto na descrição do problema, quanto na descrição da solução. Desta forma, após ser adaptado o enunciado do caso é modificado para “farol de milha não acende” com a solução “troque o farol de milha”.

Apesar de simples, a ilustração dada permite comentar algumas das principais características da reutilização: a adaptação não pode invalidar a solução do caso recuperado ou gerar situações inconsistentes. A adaptação permite a otimização do uso de espaço na base de dados, pois a base de casos de exemplo poderia perfeitamente possuir um caso enunciado como “farol de milha não acende”. Existem diferentes maneiras de adaptar um caso, que vão da simples inclusão, ou substituição, de dados nos atributos dos casos (o que foi feito no exemplo) até a alteração da solução contida no caso recuperado, de forma que a escolha de tais opções seja ditada pelo grau de entendimento e descrição do domínio do problema. Existem inúmeras estratégias de reutilização possíveis, desde a adaptação puramente manual (realizada pelo usuário) até as automáticas, que vão da simples aplicação de um conjunto de regras heurísticas até outras mais sofisticadas. Enumerar todas estaria fora do escopo desse trabalho (para se ter uma idéia das abordagens possíveis, é recomendável consultar LEWIS (1995)).

7.3 As etapas de revisão e retenção

Os sistemas de RBC têm capacidade de adaptarem-se às mudanças oriundas da aquisição de novos conhecimentos acerca do domínio do problema. Tais mudanças refletem em uma alteração nos casos armazenados ou na geração de novos casos, ou mesmo na alteração do critério de similaridade.

Quando o sistema RBC encontra uma situação para a qual nenhuma solução pertencente aos casos armazenados em sua base seja válida, têm-se início a um processo de avaliação da situação, pertencente às etapas de revisão e retenção. Existem várias causas que podem levar a essa situação: os casos cadastrados podem não cobrir todo o espectro de possíveis situações, falta de conhecimento disponível *a priori* (seja ele empírico, causal ou estrutural) ou por falha na transposição desse conhecimento quando da construção da base de casos.

Porém, para os sistemas de RBC esta situação não é intratável. O paradigma RBC foi pensado para trabalhar essas situações. Problemas cuja solução não conste da base de dados são vistos como potenciais portadores de novas informações sobre o domínio do problema, ou, na pior das hipóteses, elemento para testar a robustez da métrica de similaridade utilizada. O filtro utilizado na recuperação dos casos candidatos é a similaridade. Logo, um critério incompleto ou indevidamente ajustado pode barrar candidatos válidos, ou deixar passar candidatos incorretos.

Durante a etapa de revisão, o seguinte processo tem andamento: o problema candidato a possível novo caso é avaliado em relação ao critério de similaridade. Caso não seja a similaridade a causa que impossibilitou a localização da solução, então um conjunto de especialistas deverá analisar o problema, propor solução e decidir se o problema, agora solucionado, será definido como novo caso e como ele se encaixará na hierarquia da biblioteca de casos. Se for a similaridade a causa da não se localizar a solução, ela estando presente na base de casos, então a medida de similaridade deverá ser avaliada com o intuito de se definir pelas alterações (ou substituição) pela qual ela deverá passar. A retenção é o processo de aprendizagem definido para o paradigma RBC.

7.4 Um exemplo de aplicação da abordagem RBC para diagnóstico

Como exemplo, considere novamente o problema do diagnóstico de automóveis agora sob a ótica da modelagem de CBR, em que cada caso seja situação diagnóstica conhecida e experimentada. Por exemplo, o caso 1 na Figura (10) indica situação na qual o diagnóstico da ignição defeituosa seja apresentado quando o motor não ligou, e nenhuma informação estava disponível em relação ao status dos faróis e pára-brisas.

CASO 1	
-Motor	Não
-Farol	-
-Pára - Brisa	-
- Falha na Ignição 0	

CASO 2	
-Motor	Não
-Farol	Não
-Pára - Brisa	-
- Falha na Bateria 0	

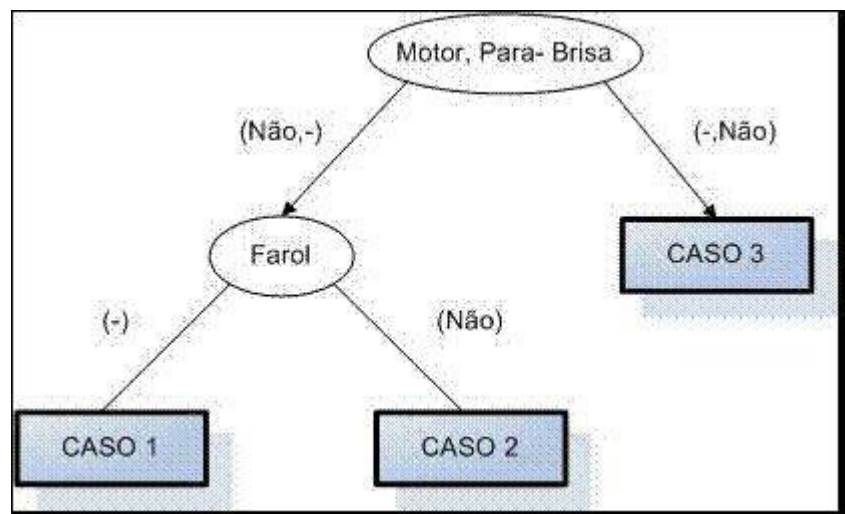
CASO 3	
-Motor	-
-Farol	Sim
-Pára - Brisa	Não
- Falha no Motor do Para - Brisa 0	

Figura 10

Os casos em nosso diagnóstico de automóveis têm três campos, que correspondem ao status do motor, dos faróis e dos pára-brisas. A figura (11) mostra a organização hierárquica possível da biblioteca de casos. Tais organizações da memória têm como objetivo otimizar o processo de recuperação dos casos.

Figura 11 - organização hierárquica da biblioteca de casos para o exemplo do diagnóstico automobilístico.

Agora cabe, através da heurística, definir a medida de similaridade. Para nosso problema diagnóstico, nós pudemos usar a heurística que, em circunstâncias



desconhecidas, produza fraco reconhecimento. Para valores idênticos de sim ou não produzam um reconhecimento forte e para valores opostos entre sim e não o reconhecimento não seja gerado. Desta forma, os valores 0, 1, e 2 são associados as situações de desconhecimento, reconhecimento fraco e reconhecimento forte, respectivamente.

Por exemplo, suponha-se que uma entrada “sim” corresponda ao campo do caso referente ao motor e o Sistema de Diagnóstico esteja verificando um caso na biblioteca que tenha entrada “não” no campo correspondente do motor, o grau de reconhecimento entre os dois casos para

o campo do motor assumirá então valor 0. O grau total de reconhecimento entre dois casos é definido pela soma simples de seus graus de reconhecimento para cada campo individual.

7.5 Comparação entre a opção do RBC para o diagnóstico em relação às outras abordagens para diagnóstico de falhas

Dentre tantas outras aplicações encontradas, o paradigma RBC tem sido aplicado em vários sistemas de diagnóstico de falhas. Apenas como exemplos, podemos citar o SPOTLIGHT da *CaseBank* utilizado no processo de manutenção dos aviões da empresa canadense Bombardier, o FLD (*Flight Line Diagnostic*), sistema de diagnóstico desenvolvido para a família dos aviões EB-145 da Embraer, e CASCADE (SIMOUDIS, 1992). Um dos motivos da boa aceitação de sistemas RBC, por parte das equipes técnicas, se deve ao fato de que sua dinâmica é muito próxima à maneira como as equipes de manutenção pesquisam e tratam as falhas em determinado campo técnico. Soma-se a isso o fato de que a forma pela qual os sistemas RBC armazenam o conhecimento (casos) é muito mais natural, fácil de ser compreendida e avaliada por seres humanos.

Os sistemas de RBC são diferentes dos sistemas baseados em regras e dos sistemas baseados em modelo, descritos anteriormente. Em comparação aos sistemas de diagnóstico baseados na abordagem RBM, o diagnóstico em um sistema RBC não é derivado de um modelo genérico do equipamento. Em relação aos sistemas de diagnóstico baseados em regras, os sistemas RBC apresentam o conhecimento acerca do domínio do problema explicitamente pela descrição de situações singulares armazenadas em sua base de casos, um formato bem mais estruturado que um simples conjunto de regras, por mais elaboradas que sejam. De igual maneira, a dificuldade de atualização das regras verificada nos primeiros sistemas de diagnóstico é tratada, nos sistemas RBC, pela adição de uma etapa de aprendizagem.

Enquanto os sistemas de RBM se fundamentam no conhecimento estrutural e causal, os sistemas baseados em RBC podem utilizar também o conhecimento empírico ao enfatizar as instâncias das ocorrências de determinadas situações. Não que os sistemas de RBC não possam ter ser casos construídos em cima de conhecimento estrutural e causal. Muito pelo contrário, conhecimento estrutural e causal pode ser utilizando na construção da base de casos inicial.

8 O Troubleshooting

Vimos, no capítulo referente ao diagnóstico clínico, que a ordem em que são realizados os testes laboratoriais é importante em função dos custos e dos riscos. No campo da manutenção ocorre fenômeno análogo. Às vezes, não é aconselhável obter informações por meio de testes que têm alto custo, em função de restrições de tempo (o teste é muito demorado) ou financeiras (o teste é muito caro).

Abaixo é descrita, como exemplo, uma situação hipotética referente aos defeitos que podem ocorrer nas partes constituintes do trem de pouso e sua probabilidade de ocorrência.

Tabela 2 – Possíveis defeitos para o Sistema Diagnosticável trem de pouso

Defeito	Descrição	Probabilidade
Def ₁	Placa controladora queimou	0.4
Def ₂	Cabeamento solto	0.3
Def ₃	Vazamento hidráulico	0.2
Def ₄	Servomecanismo obstruído	0.1

Existe um conjunto, chamado de conjunto A, de ações que eliminam esses defeitos (veja tabela 4) e assim como os defeitos têm suas probabilidades de ocorrerem, as ações têm um custo associado à sua execução. Temos também o conjunto de perguntas P, resultantes do conjunto T de testes que têm como objetivo verificar e registrar quais Sintomas estão presentes. Esses testes, assim como as ações em A, também possuem custo associado. Fica assumida a condição de independência entre o conjunto A e P.

Tabela 3 – Ações e perguntas para o problema do trem de pouso

Defeito	Descrição	Custo
A ₁	Troque a placa	2
A ₂	Troque os cabos	5
A ₃	Troque o módulo hidráulico	20
P ₁	Existe vazamento de óleo?	10

O diagnóstico por teste seqüenciado ou *troubleshooting* consiste na construção da seqüência de perguntas e execução de ações que definam a síndrome e corrijam o defeito presente. O *troubleshooting* parte do pressuposto de que é necessária a intervenção humana pelos testes (perguntas) para se isolar a síndrome, porém tendo como fator limitante o custo associado às ações e perguntas. Portanto, uma sessão de *troubleshooting* deve isolar o defeito de forma a **minimizar o custo**.

Definição 3.1: Uma *estratégia de troubleshooting* é um grafo que descreve as realizações das ações e das perguntas com o objetivo de diagnosticar e corrigir o defeito.

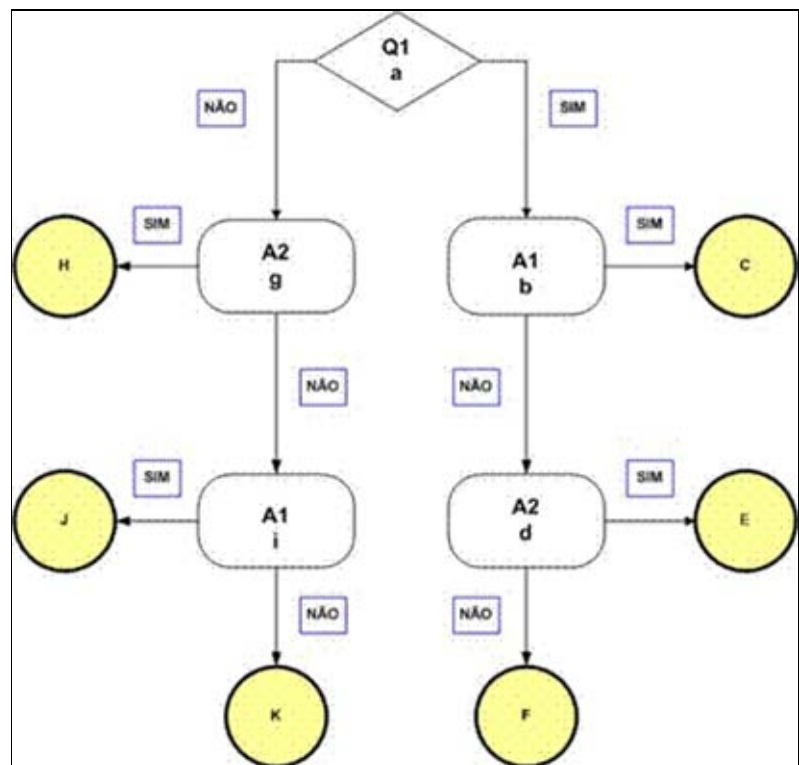
Existem muitas possíveis estratégias para o mesmo conjunto de P e A, tratando-se, pois, de problema combinatório. Seguindo nosso exemplo, se à pergunta P_1 “Existe vazamento de óleo?” corresponder resposta positiva, os defeitos Def_1 , Def_2 e Def_4 são eliminados.

Poderíamos também tentar A_1 e conseguirmos corrigir o defeito. Outra estratégia seria realizar A_1 e, em caso de insucesso, realizar A_2 . A terceira estratégia seria responder à questão P_1 e, em caso de resposta positiva, realizar A_3 . Não detalharemos aqui, é necessário manter em mente que, de todas as estratégias possíveis, só nos interessa aquelas de menor custo.

Definição 3.2: Um *diagrama de estratégia* é a representação gráfica da estratégia de *troubleshooting*. Na figura (12) temos exemplo de diagrama de estratégia.

Figura 12 - Exemplo de diagrama de estratégia.

Os nós são rotulados com a seqüência (a, b, c, ... k). O losango simboliza os elementos do conjunto P de questões, os retângulos de bordas arredondadas representam as ações intermediárias, enquanto que os círculos cinzas são as ações finais, aquelas ações que matam as panes.



Em SOCHOROV (2000) é provado que o problema de teste sequenciado é NP-difícil. Existem vários algoritmos sub-ótimos desenvolvidos para a geração de seqüência de testes, os mais conhecidos são: *Multistep information Heuristic approach* em (RAGHAVAN *et al.*, 1999) e *Rollout strategies*, descrito em (TU, PATTIPATI, 2002).

9 Outras abordagens: O Diagnóstico baseado em testes de hipóteses e o modelo PMC

A abordagem de diagnóstico baseado em teste de hipóteses, também conhecida como a teoria diagnóstica da competência (BENJAMINS, *et al.*, 1994), segue as idéias de CHANDRASEKARAN (1988), já discutidas nesse trabalho. Nesta abordagem, o diagnóstico é visto como um problema que consiste em identificar a causa da falha que se manifeste por algum comportamento observado. É necessário, portanto, definir a combinação de métodos de resolução geral de problemas que conduzam a uma solução geral de diagnóstico. A esta combinação de métodos, os pesquisadores desta abordagem dão o nome de estratégia diagnóstica.

Um método de resolução geral de problemas toma o problema em questão e o descreve em termos de tarefa a serem realizadas. A tarefa é uma especificação do conjunto de ações ou estados que devam ser realizados ou percorridos. O método, então, define a maneira como o objetivo desta tarefa pode ser alcançado. O método estabelece ainda as entradas e saídas e decompõe a tarefa em subtarefas com objetivos próprios. Como a própria subtarefa pode consistir de subtarefas diferentes, esta decomposição se torna recursiva até que as subtarefas atômicas, também chamadas de inferências primitivas, sejam encontradas. Além disso, o método especifica o controle e a ordem da execução e das iterações entre as subtarefas.

A entrada do método que implementa a estratégia diagnóstica é o conjunto de observações iniciais, que são também a entrada da tarefa da detecção do sintoma. Nessa tarefa decide-se se as observações do equipamento são observações de anormalidade ou de normalidade. As observações de anormalidade e de normalidade são entradas para a tarefa da geração da hipótese. A tarefa gera o conjunto de hipóteses, que é a entrada para a tarefa da discriminação da hipótese. Aqui, as observações adicionais são feitas se necessárias. A saída da tarefa da discriminação, que é também a saída do pdm, é o conjunto de diagnósticos que explica as observações realizadas.

Desta forma, a tarefa do diagnóstico pode ser dividida em três subtarefas: detecção de sintomas, geração de hipóteses e discriminação de hipóteses ilustradas na figura (13).

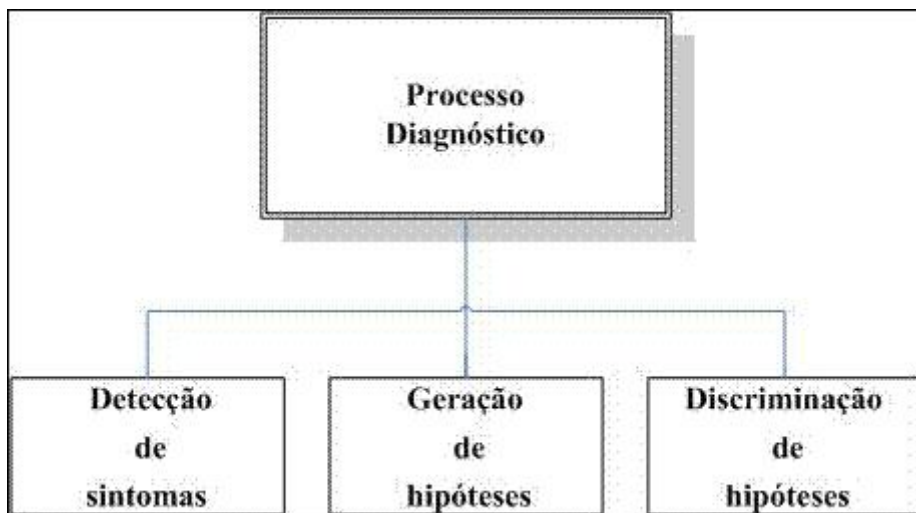


Figura 13 - Hierarquia das tarefas em sistemas baseado em teste de hipótese.

9.1.1 Detecção de sintomas

Esta subtarefa consiste em verificar se, em meio aos dados pertinentes às observações realizadas no o equipamento diagnosticado, existe realmente um conjunto de sintomas. Um sintoma é definido como uma observação que se desvia de sua expectativa.

Existem vários métodos que podem ser utilizados para a subtarefa de detecção de sintomas. Todos eles podem ser agrupados em três categorias: comparativo, classificatório e por configuração do usuário.

O método por comparação gera um valor previsto e esperado para uma determinada observação e, então, o compara com a observação realizada. Desta comparação, resulta em dois resultados possíveis: ou os valores são iguais ou são diferentes. Se a observação e a expectativa forem iguais, a observação será considerada normal, caso contrário a observação é considerada anormal (portanto, um sintoma é assinalado).

O método por classificação supõe a disponibilidade de um critério, seja ele analítico ou heurístico (como as redes neurais, por exemplo) ou estatístico, de classificação do domínio dos possíveis resultados da observação; com este critério a observação inicial é classificada como normal ou anormal.

O método de configuração pelo usuário se apóia no conhecimento extraído da experiência do usuário para determinar a presença ou ausência do sintoma. Este método foi utilizado em um dos primeiros sistemas de diagnósticos, porém de aplicação médica, o INTERNIST I (MILLER *et al.*, 1982).

9.1.2 Geração de hipóteses

Esta tarefa gera um conjunto de hipóteses candidatas que justificam o conjunto anormal de observações iniciais. As observações normais podem render um critério auxiliar ao processo de geração da hipótese. O método por compilação é o método clássico para a geração de hipóteses, ele explora as associações diretas entre sintomas e causas e é seguido, às vezes, por um filtro de probabilidade. Este método é usado frequentemente nos domínios nos quais a compreensão subjacente do inter-relacionamento dos componentes do equipamento seja somente parcial.

Muitos esforços foram realizados no sentido de coibir que essa etapa tornasse todo o Sistema de Diagnóstico dependente da aplicação, recaindo no mesmo problema dos primeiros Sistemas Diagnosticadores. Um desses esforços levou um ramo da comunidade Solving-Problem a adotar abordagem baseada em modelos na etapa de geração de hipóteses, permitindo que a associação entre sintomas e causas passasse por estágio intermediário, no qual o modelo do equipamento diagnosticado seja consultado e as hipóteses construídas em função da comparação realizada entre o modelo e os sintomas.

9.1.3 Discriminação de uma hipótese

Esta última etapa consiste em discriminar entre as hipóteses geradas, usando observações adicionais quando convier. Cada hipótese tem que ser avaliada e existem diferentes métodos para implementar esse processo.

O método seleção aleatória é aproximação simples e é útil quando as relações entre hipóteses não são modeladas. A estratégia “estabelecer e refinar” (*establish and refine strategy*) (CHANDRASEKARAN, 1988) avança no sentido em criar critério que defina o relacionamento entre as hipóteses e ordena a fila de hipóteses a ser avaliada em função dessa relação.

Outro método mais refinado consiste em selecionar a hipótese de acordo com algum outro critério dependente do conhecimento do domínio. Por exemplo, cada hipótese, no conjunto, é associada ao estimador de custo para testar a hipótese em relação ao conjunto inteiro. Este estimador pode ser baseado em tipos diferentes de conhecimento dependendo do domínio. Pode-se compor o custo em relação à frequência e à importância das observações com respeito ao conjunto de hipótese, ou usar método que minimize o número dos testes extras a serem executados (quando eles forem necessários). As hipóteses são requisitadas de acordo

com o seu custo estimado. Se a quantidade de dados for muito grande, ou o custo dos testes divergir, pode-se criar subetapas.

O Modelo PMC

A sigla PMC é formada pela composição da primeira letra do nome pertencente a cada um dos três pesquisadores que criaram o modelo: Preparata, Metze e Chien (PREPARATA, METZE, CHEN, 1967). O modelo PMC foi um dos primeiros a fugir do modelo de diagnóstico baseado em regras, antes mesmo dos modelos baseados em resolução geral dos problemas. Neste modelo, o sistema é decomposto em unidades não necessariamente idênticas, denotadas pelo conjunto:

$$P = (u_1, u_2, \dots, u_n)$$

onde cada unidade é partição atômica (que não pode ser mais decomposta) para o propósitos de diagnóstico. Considera-se que estas unidades são ou “defeituosas” ou “não-defeituosas”, e que esse status não muda durante o diagnóstico. As unidades devem ser capazes de testar outras unidades no sistema e determinar se elas estão funcionando corretamente.

No modelo PMC, cada unidade que pertence a P possui um subconjunto de P para ser testado (nenhuma unidade testa a si mesma). O conjunto completo de testes é representado pelo grafo G, em que cada nó representa um elemento do conjunto U e uma aresta (u_i, u_j) existe em G se, e somente se, o nó u_i testa o nó u_j .

Uma saída a_{ij} é associada a cada teste (u_i, u_j) . Se u_i é livre de falhas, então a_{ij} é 0, caso u_j for não defeituoso, e 1 se u_j for defeituoso. Se u_i for defeituoso, o resultado não é confiável e pode ser qualquer valor. Um sistema S é dito t-falhas-diagnosticável se todas as unidades defeituosas em S podem ser identificadas, visto que o número de unidades defeituosas não pode exceder t.

O teste envolve a aplicação controlada de algum estímulo a alguma unidade e a posterior observação das respostas. A saída do teste pode assumir dois estados: falhou ou não falhou. Supõe-se que o teste é **completo**, isto é, se o teste for aplicado por um nó não-defeituoso, então ele sempre retorna o status correto do nó testado e ele reflete precisamente as condições da unidade testada.

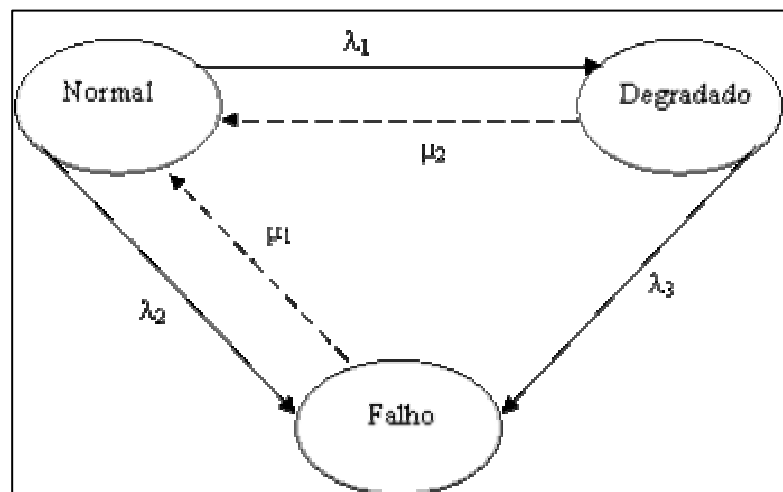
10 Sistemas automáticos de diagnóstico

Cabe definir, a partir de agora, o conceito de sistema Paciente (S-Paciente). S-Paciente é um sistema de engenharia (sistema estruturado que possui finalidade), uma entidade projetada e utilizada para determinado conjunto de objetivos, construída a partir de conceitos e técnicas, sejam elas empíricas ou não, provenientes de um ou mais ramos da Física e da Engenharia.

O sistema S-Paciente, tal qual um avião, um automóvel ou mesmo uma ponte, pode apresentar-se em duas situações cruciais do ponto de vista da segurança, confiabilidade e qualidade: com a presença ou ausência de falhas. Esta é a primeira visão para a construção do modelo aqui apresentado. Como primeira idéia para o conceito de falha, considera-se falha uma anormalidade no funcionamento do sistema, que o incapacita de executar o conjunto de funções ao qual se destina. Do desmembramento desse fato, verifica-se que o S-Paciente pode apresentar três estados possíveis: **Normal** (sem falhas), **Degradado** (falhas parciais ou em parcela de suas funções) e **Inoperante** (totalmente falho).

A inter-relação entre estes três estados é descrita na Figura (14).

Figura 14 - Diagrama ilustrando a interrelação entre os três estados possíveis do Sistema de Engenharia do ponto de vista do estudo do diagnóstico de falhas. Os valores λ_1 , λ_2 , λ_3 são as taxas de falhas de levam à situação de anormalidade. Já μ_1 e μ_2 são as taxas de correção.



A taxa de correção pode ser estimada pela expressão:

$$\mu_i = \frac{\sum_j C}{\sum_j T^M} \quad (6)$$

onde C é Número de falhas corrigidas pelas ações de manutenção em T^M (intervalo de tempo gasto na manutenção ou substituição do sistema de engenharia falho). Este tempo inclui as atividades de inspeções, checagens, serviços e reparos.

Um Sistema de Diagnóstico Automático (SDA) é uma ferramenta que implementa um processo de diagnóstico cujo objetivo é maximizar a taxa μ_i , de maneira mais automatizada

possível, de forma a minimizar a intervenção humana. A automatização do processo de diagnóstico não é um mero capricho, mas necessidade econômica. De acordo com OLIVE (2003): (...) “Geralmente, o diagnóstico é realizado por especialistas humanos resultando freqüentemente em erros devido à complexidade dos sistemas. Daí, portanto, é imperativa a redução da intervenção humana na geração de árvores de diagnóstico”.

Cabe salientar que a automatização aqui descrita não se restringe na transcrição de um Manual de Manutenção em alguma mídia eletrônica. Um SDA pode sim fazer uso dos Métodos contidos em um Manual como base de seu conhecimento.

As características do SDA, de acordo com LAFORTUNE e SAMPATH (2000) são:

- O SDA deve ser fácil de implementar;
- O SDA deve ser simples de integrar com o equipamento;
- O SDA deve usar de maneira eficiente os sensores do Sistema S-Paciente.

Já VENKATASUBRAMANIAN *et al.* (2003) dão conjunto um pouco diferente para as características desejáveis para um SDA, são elas:

- Detecção e diagnóstico rápidos
- Isolabilidade
- Adaptabilidade
- Robustez

10.1 Rapidez na detecção e no diagnóstico

O sistema diagnóstico deve responder rapidamente em detectar e diagnosticar maus funcionamentos. Entretanto, uma análise acurada deve ser realizada para que o SDA não confunda ocorrência de falha com ruídos inerentes ao processo, o que pode conduzir à ocorrência freqüente de falsos alarmes durante operação normal. De acordo com AGUIAR (2001) podemos definir um falso alarme como o evento em que o SDA indica o diagnóstico de falha que efetivamente não ocorreu, ou quando a falha diagnosticada não condiz com a que de fato aconteceu.

De uma maneira geral, o sistema de diagnóstico implementado deve responder rapidamente ao detectar os maus funcionamentos no processo diagnosticado. Portanto, fica claro que,

independentemente do processo ou equipamento diagnosticado, o SDA implementado deve ter por características ser **eficiente e eficaz**.

Pode-se definir o tempo gasto pelo SDA pela expressão:

$$\Delta T^D = \Delta T^M - \Delta T^C \quad \text{Onde:} \quad (7)$$

ΔT^M : Tempo total gasto durante a manutenção

ΔT^D : Tempo para o SDA para realizar o processo de diagnóstico de falha.

ΔT^C : Tempo efetivamente gasto para corrigir as falhas.

10.2 Isolabilidade

Isolabilidade é a capacidade de o sistema diagnóstico distinguir entre diferentes falhas. Sob condições ideais, livres de ruído e das incertezas do modelo, o sistema diagnóstico deve ser capaz de gerar saída ortogonal às falhas que não ocorreram. Naturalmente, existe conflito entre a isolabilidade e a presença de incertezas no modelo. Em abordagens como o Teste de Hipótese, este conflito pode ser minimizado pela aquisição de dados adicionais.

A eficiência do SDA se traduz em quanto o sistema maximiza o índice médio de rendimento:

$$I = \frac{D}{T^D} \quad (8)$$

onde o termo D é o conjunto dos Diagnósticos corretos fornecidos pelo SDA. A situação de perda se apresenta nos casos em que o SDA deveria detectar falha e não o fez. A razão entre a quantidade de diagnósticos corretos e incorretos corresponde à medida de eficácia média do SDA, e pode ser expressa por:

$$E = \frac{D}{F} \quad (9)$$

onde F é o total de diagnósticos incorretos.

10.3 Robustez

É desejável que o sistema diagnóstico seja robusto em relação ao vários ruídos e às incertezas inerentes ao processo e equipamento diagnosticado. Baixa robustez pode levar a situações de

perda de diagnóstico que é verificada quando situação real de falha não é percebida pelo SDA, de forma que não seja gerado diagnóstico para a situação.

10.4 Capacidade de identificação para situações novas

Em uma situação ideal, o modelo do domínio do problema pode dispor de dados suficientes para modelar o comportamento do equipamento de forma propiciar visibilidade completa das regiões de funcionamento anormais. Entretanto, em situação real, não é possível obter os dados completos (seja dados históricos, modelos analíticos ou mesmo empíricos) de forma a modelar as regiões anormais de funcionamento satisfatoriamente. Somente alguns sintomas podem ser isolados, de forma a cobrir apenas parcelas da anormalidade. Mesmo sob estas circunstâncias difíceis, é desejável que o sistema diagnóstico possa reconhecer a ocorrência de falhas não identificadas e possibilitar algum mecanismo de rastreamento para se identificar as causas que as geraram. Os sistemas baseados em CBR têm, parcialmente, essas habilidades.

10.5 Transparência nos resultados

Requisito importante para o sistema diagnóstico é a capacidade de gerar credibilidade do usuário em sua confiabilidade. Uma maneira de se obter isso seria fornecer a estimativa *a posteriori* da exatidão do resultado obtido. Tais medidas seriam úteis para projetar a confiabilidade nas decisões diagnósticas dadas, o que leva o usuário à sensação de maior segurança nas recomendações dadas pelo sistema.

10.6 Adaptabilidade

O processo de adaptabilidade pode ser dividido em dois pontos. No primeiro, a adaptabilidade diz respeito à capacidade do sistema de diagnóstico respeitar as mudanças gerais de comportamento do equipamento diagnosticado que não caracterizem incidência de falhas. As circunstâncias que levam o equipamento a sofrer tais mudanças dissociadas de falhas podem ser várias: vão desde alterações nas condições ambientais – como clima, ou até mesmo o volume da produção com variação na demandas, mudanças estruturais – como retirada ou inserção de módulos funcionais – como a instalação de leitor de código de barras em caixa eletrônico.

O outro ponto, tendo em vista a adaptabilidade, diz respeito à capacidade de o sistema incorporar, sempre que necessário, novos diagnósticos a medida que novas situações, imprevistas *a priori*, emergem. Sob esta perspectiva, a adaptabilidade é complementar à capacidade de o sistema detectar novos casos.

10.7 Clareza na explicação dos resultados

Além da capacidade de identificar a fonte do mau funcionamento, o sistema diagnóstico deve também fornecer explicações da origem da falha e sua propagação até a situação atual. Este fator é muito importante em sistemas de diagnóstico como ferramentas de apoio à decisão. Isso requer habilidade de articular sobre a causa e os relacionamentos com os efeitos. O sistema diagnóstico tem que justificar suas recomendações de modo a que o operador possa confortavelmente avaliar e agir usando sua experiência.

Do ponto de vista do diagnóstico de falhas, o Sistema de Engenharia pode estar em dois estados principais em relação à sua utilização: ou estará operacional ou em manutenção, o ΔT^M está definido em período de tempo em que o Sistema de Engenharia estará em seu estado não-operacional. Disso, conclui-se que a utilização do SDA seja assumida aqui quando das manutenções, vistorias ou inspeções, sejam elas preventivas, corretivas ou periódicas (anuais, semestrais...).

A escolha do período de manutenção para a utilização do SDA não é limitação, mas decisão do momento mais adequado para sua utilização, pois segundo DA COSTA (2004):

“A Manutenção Aeronáutica é uma atividade que produz grande número de erros... Mesmo com os métodos de segurança e Sistemas de Gerenciamento de Qualidade sendo arduamente estudados e implementados, o risco em manutenção nunca pode ser inteiramente eliminado” (...).

Este exemplo, advindo do contexto aeronáutico, pode ser generalizado; a utilização dos SDAs durante a manutenção tem por objetivo minimizar os erros que porventura possam ocorrer durante essa importante atividade.

11 A importância do diagnóstico de falhas em Engenharia

O resultado do mau funcionamento em um sistema S-Paciente crítico pode ser catastrófico. Os engenheiros envolvidos com tais sistemas, assim como os médicos, não colecionam apenas falhas, mas também podem colecionar óbitos. Em cenários mais suaves, as falhas produzidas em tais sistemas e nos seus respectivos processos podem causar paradas indesejáveis que resultem em prejuízos e atrasos na produção, além de produzir impacto negativo no meio ambiente (GASCA, *et al.*, 2003).

O diagnóstico eficiente é importante, pois aumenta a segurança e a confiabilidade, ele também é importante para a proteção de ambiente e para melhorar a manutenção. O diagnóstico consome muito tempo e é muito sofisticado para ser executado manualmente. Os métodos conseqüentemente automatizados são necessários.

As razões de sua importância podem ser divididas em duas partes, as necessidades de diagnóstico aumentaram e a disponibilidade de diagnóstico aumentou na medida em que aumentou a disponibilidade de computadores nos sistemas.

A medida em que os equipamentos trabalhem cada vez com maior número de funcionalidades, o diagnóstico manual irá se tornar cada vez mais trabalhoso e dispendioso. Sistemas de diagnóstico realizam ou auxiliam na obtenção de conclusões corretas. O custo de manutenção de um sistema complexo pode exceder o custo de compra do equipamento por fatores de três a dez vezes (PATTIPATI e TU, 2002).

Plantas químicas. As estatísticas industriais apontam para um fato singular: mesmo que os e os desastres mais catastróficos relacionados com a detecção e diagnóstico de falhas (como a explosão da refinaria petroquímica Mina *Al-Ahmedhi-Ahmedhi*, no Kuwait em 2000, e a explosão da plataforma P-36, da Petrobras em 2001) sejam menos freqüentes, os acidentes menores são muito comuns, tendo por resultado muitos ferimentos ocupacionais, ferimentos esses que causam impactos sérios na saúde dos funcionários. Fatos como esse geram custos de bilhões de dólares a cada ano. Como exemplo, estima-se que apenas a indústria petroquímica dos Estados Unidos incorra em aproximadamente 20 bilhões de dólares de perdas anuais devido a manutenções incorretas (VENKATASUBRAMANIAN *et al.*, 2003). Situações similares custam à economia britânica por volta de 27 bilhões de dólares cada ano (LASER, 2000). Custos muito maiores ocorrem em outras indústrias, como a farmacêutica.

12 Os Sistemas complexos diagnosticáveis (SCD)

“Todo bom modelo começa com uma pergunta. O arquiteto de um modelo deve sempre escolher o nível correto do detalhe que responde à pergunta feita”. (GOLDENFELD, 1999).

A frase acima é a base de algumas questões referentes à formalização do conceito de diagnóstico: como o objeto de estudo do diagnóstico deve ser visto e como analisar seu comportamento para que se possa levantar o conjunto completo de todas as relações de causa e efeito no que tange às situações de falha? Existe maneira eficiente de estabelecer essas relações, bem como verificar em qual caso (ou possivelmente casos) a ocorrência se encaixa? Qual deve ser a estrutura prática que dê base para essas relações de causa e efeito?

A maioria das teorias de diagnóstico de falhas das duas correntes atuais trabalha na modelagem do processo diagnóstico *per se*, no algoritmo ou na modelagem dos eventos que levam o Sistema de Engenharia ao estado de falha ou degradação. A terminologia dos conceitos aplicados não é consenso na literatura, de maneira que definições de pontos fundamentais envolvidos na questão do processo de diagnóstico têm tratamento diversificado e que varia de trabalho para trabalho até mesmo entre publicações da mesma linha, seja ela DX, FDI. Exemplo disso é o conceito de defeito em componente: Ora temos a definição de defeito associada à idéia de falha, ora temos o mesmo conceito tratado como Sintoma.

Neste trabalho, será proposta uma estrutura que hierarquize os conceitos fundamentais do processo de diagnóstico, a saber: Defeito, Sintoma, Síndrome, Diagnóstico, de forma a encadear de maneira lógica os conceitos posteriores e evitar ambigüidades. Desta maneira será definida, de maneira construtiva, uma terminologia consistente e bem justificada, fundamentada em bases sólidas.

Para isso, assumiu-se a premissa, já adotada tanto pelo grupo FDI quanto DX, de que se deva ter como ponto de partida a modelagem do objeto de estudo de forma orientada a responder as questões colocadas no início desse tópico, de forma a possibilitar que os SDAs cumpram seus objetivos. Tal modelo é calcado no conhecimento que possuímos acerca do objeto e quanto formalmente ele seja estruturado (GASCA, *et al.*, 2003). Para tanto, este trabalho lança mão do modelo de Sistemas Complexos Diagnosticáveis.

12.1 Definindo os Sistemas complexos diagnosticáveis

O Modelo dos Sistemas Complexos Diagnosticáveis surge da seguinte questão: seria possível pensar um modelo substitutivo para sistemas diagnosticáveis que abarcasse a vasta gama de Sistemas de Engenharia, aparentemente heterogêneos entre si, de modo a que o processo de diagnóstico acompanhasse essa generalização? Se a resposta for sim, então é possível construir SDAs que sejam adequados, senão a todos, pelo menos a grande número de situações heterogêneas e, assim, construindo um modelo que possibilite a criação do Sistema de Diagnóstico Automático e Genérico, um SDAG.

O conceito de Sistema Complexo Diagnosticável surge, portanto, como ponto central para o estudo da Engenharia Diagnóstica. Para STOREY (1996):

“Sistemas complexos são mais difíceis de serem projetados, portanto, mais susceptíveis a conter erros de projeto. Também são mais difíceis de testar, com maior chance, portanto, de conter erros não detectados. São mais vulneráveis a falhas humanas na instalação ou uso.”

Segundo FRANK (1996):

“Os sistemas de automação estão se tornando mais e mais complexos e os algoritmos de controle mais e mais sofisticados. Conseqüentemente, há uma demanda pelo crescimento da tolerância de falha, obtida não somente melhorando a confiabilidade individual das unidades funcionais que compõe o Sistema, mas também por uma detecção, isolamento e tratamento de falhas eficiente.”

Já para VASANDANI VIJAY, GOVINDARAJ (1990):

“A tarefa diagnóstica é dificultada freqüentemente pelo tamanho, pelas interações e pela dinâmica do Sistema. O tamanho diz respeito ao número de componentes que fazem parte do sistema. Um aumento no número desses componentes aumenta a probabilidade de falha no sistema e torna mais difícil a pesquisa da causa da mesma, aumentando as alternativas que podem explicar o comportamento anormal observado do sistema. A interação entre partes do sistema e da propagação do comportamento anormal no sistema faz do diagnóstico de falhas um problema difícil”.

Encontramos uma explicação um pouco mais estruturada em KRYSANDER (2003):

“Para produzir produtos competitivos, questões como segurança e desempenho, por exemplo, têm que continuamente ser melhorados. Freqüentemente quando os produtos técnicos são melhorados os sistemas são divididos em um número maior de componentes onde cada componente tem uma funcionalidade cada vez mais especializada. Aumentar o número dos componentes e aumentar o grau de especialização (*specialization mode*) conduzirá a uma complexidade maior do sistema. Muitos sistemas técnicos têm alcançado hoje o limite, onde nenhum ser humano consegue ter uma visão completa e detalhada do sistema como um todo”.

A base teórica dos Sistemas Complexos Diagnosticáveis (SCD) tem origem na Teoria Geral dos Sistemas desenvolvida por Ludwig von BERTALANFFY (1969).

Se for possível chegar-se ao denominador comum formal em relação a todos os Sistemas de Engenharia passíveis de serem diagnosticados, então os SDAG podem ser desenvolvidos e não estarão presos à determinada aplicação ou tecnologia em particular. Esta busca da generalização é a raiz do trabalho de Bertalanffy, apesar de seu trabalho inicial ter fortes influências da Biologia.

O termo complexidade vem do latim, *complexus*, que significa entrelaçado. Embora haja muitas abordagens para definir o que é um Sistema Complexo, seja ela do ponto de vista da Física, Economia ou Psicologia, aqui o interesse recai apenas no conceito de Sistemas Complexos **Diagnosticáveis**. Em razão disso, a complexidade será tratada dentro do escopo dos problemas de diagnósticos de falhas. Para tanto, este trabalho assume os quatro postulados abaixo, que serão a base da modelagem aqui apresentada:

Postulado 0: O SCD é Sistema da forma como o IEEE *electronics terms* define a palavra: “uma combinação de componentes (elementos) que agem juntos para exercer função que não seria possível ser exercida pelas partes (elementos) individualmente. Além disso, o SCD é ente entrópico e não-conservativo.

Postulado 1: O SCD deve ser composto por um conjunto com mais de um elemento. Deve ser possível criar mais de um conjunto nos quais os elementos difiram, quanto a sua natureza, de um conjunto para outro.

Postulado 2: Cada um dos elementos que compõem o SCD deve ter pelo menos uma interligação com outro elemento diferente do Sistema. Tais interligações compõem a estrutura do sistema de tal forma que ela seja estável (HEYLIGHEN, 1988). Entende-se por estabilidade a propriedade de a estrutura permanecer constante no tempo, completamente definida e organizada. Estas ligações devem tornar o SCD sinérgico e simbiótico, interna e externamente (GOLDBARG, LUNA, 2000).

Postulado 3: Deve haver um conjunto mínimo de elementos do SCD em que seja possível fornecer um conjunto não-vazio de observações de suas variáveis características. Entende-se por variáveis características as medições de um subconjunto não-vazio do total das dimensões que definam um elemento que compõe o SCD, sejam elas estáticas ou dinâmicas.

Postulado 4: O emprego de métodos reducionistas e abstrações para a análise do SCD tem um limite físico. Em determinado momento, não se poderá mais subdividir partes do sistema em sub - elementos menores.

Agora é possível, capitaneado por Bertalanffy, definir o que é um Sistema Complexo Diagnosticável R.

Definição 4.1: Um Sistema Complexo Diagnosticável (SCD) R é definido como uma quádrupla:

$$R \in \mathfrak{S} \Leftrightarrow R = (S, E, L, G) \quad (10)$$

Onde \mathfrak{S} : É o Conjunto dos Sistemas Complexos Diagnosticáveis.

$$\mathfrak{S} = (R_1, R_2, \dots, R_n) \quad (11)$$

S compõe o conjunto dos Subsistemas de R. A relação com $S \in R$ é a *hierarquia* de R, de forma que:

$$S = (K_1, K_2, \dots, K_n) \text{ com } S \subset \mathfrak{S} \text{ e } K_i \neq R \quad (12)$$

Definição 4.2: O índice W constitui-se no grau de hierarquias em R. Quando o sistema R possui $S = \{\}$, é dito que $W = 0$. Para uma hierarquia do tipo $\exists S \Rightarrow S \in R$, temos que $H = 1$. Quando temos subsistemas compostos de outros subsistemas em R, cada hierarquia entre os mesmos incrementa o valor de W. W também é denominado de *complexidade* da estrutura em R.

E: é o conjunto dos componentes do Sistema Diagnosticável de forma que:

$$E = (C_1, C_2, \dots, C_n) \text{ com } E \subset \mathfrak{S} \quad (13)$$

Onde \mathfrak{S} é o conjunto dos componente de \mathfrak{R} e é descrito pelo par ordenado:

$$C = (V^C, H^C) \quad (14)$$

Onde V é o conjunto de variáveis características do componente C de tal forma que:

$$V^C = (v^C_1, v^C_2, \dots, v^C_n) \text{ com } v^C_i \in \mathfrak{R} \text{ (conjunto dos Reais)} \quad (15)$$

H: é o vetor de serviços de C de tal forma que:

$$H^C = (h^{C_1}, h^{C_2}, \dots, h^{C_n}) \quad (16)$$

De forma que h é uma função do tipo : $\Psi \rightarrow Y$ com $\Psi: T \rightarrow I$ e $Y: T \rightarrow U$

Onde:

T é conjunto ordenado

I representa o conjunto de entradas do componente C

U representa o conjunto de saídas do componente C

Um serviço h é a transformação de um subconjunto de V , onde cada elemento v é alterado em v' seja h linear ou não. Para a nossa definição de R , basta que os elementos de V sejam descritivos e não-controláveis (GOLDBARG, LUNA, 2000).

Ocasionalmente, o conjunto V pode ser vazio, porém, deve existir pelo menos um elemento no conjunto dos componentes em que tal fato não ocorra. Cabe salientar que as variáveis características pertinentes ao diagnóstico são definidas no domínio do usuário de manutenção ou monitoração técnica, tal seleção pode se resumir a um subconjunto de V para um determinado C ou consistir no próprio conjunto V . De modo geral, as variáveis escolhidas devem consistir em todos os dados representativos do bom ou mau funcionamento do componente.

Dando continuidade à descrição do enunciando de R :

L : é chamado de **conjunto estrutura**, e será definido definido como:

$$L = (l_1, l_2, \dots, l_q) \text{ com } l_p = (C_m^{iK}, C_n^{jK'}, \varpi) \quad C \in E; p \leq |S| \times |E|; K, K' \subset S \quad (17)$$

Onde $\varpi \in \mathbb{N}$ é o grau de conectividade entre os dois componentes e indica o número de conexões entre os mesmos. Temos também que i e j com $i < j$ é a ordenação das conexões entre pares de componentes onde $\varpi > 1$.

G é denominado de conjunto funcionalidade de R , de forma que:

$$G = (G_I, G_E) \text{ é um mapeamento } V^{iK}_m \rightarrow V^{jK'}_n \text{ pertencentes a } C_m^{iK} \text{ e } C_n^{jK'} \text{ respectivamente} \quad (18)$$

Onde:

G_I é o conjunto das Funcionalidades Internas.

G_E é o conjunto das Funcionalidades Externas.

Definidas por $G_I : L \rightarrow H^C$ com $l_p \neq (C_m^{i^0}, C_n^{j^{K'}})$ e $l_p \neq (C_m^{i^K}, C_n^{j^0})$

$$G_E : L \rightarrow H^C \text{ com } l_p = (C_m^{i^0}, C_n^{j^{K'}}) \text{ ou } l_p = (C_m^{i^K}, C_n^{j^0}) \quad (19)$$

Com $C_m^{i^0}$ representando a entrada do Sistema R, e

$C_n^{j^0}$ representando uma saída do Sistema R.

12.2 As Conseqüências do modelo: o enunciado do problema diagnóstico

Definição 4.3: *Defeito* é a transformação $X : C_i \rightarrow C_i'$, de modo que para todos os elementos de L que contenham C_i , temos alteração no conjunto G. Logo, defeito é a alteração estrutural que modifique ou incapacite a execução de um subconjunto de G, de modo que se apresente a situação em que a forma afeta a função. Esta transformação pode ocorrer por influência de fatores externos ou internos. Operar um equipamento além das suas restrições ambientais e operacionais pode ser o agente causador do defeito.

Definição 4.4: A alteração $G \rightarrow G'$ em determinado instante t constitui *Falha*.

Definição 4.5: A *observação* $O^C_{(tipo)}$ é o conjunto gerado em determinado instante t a partir dos conjuntos V^C e H^C para o componente C com $O^C_{(tipo)} \subseteq V^C \cup H^C$ e $C \in E$ para dado R. Onde *tipo* pode ser externo ou interno dependendo de envolver componentes com entradas ou saídas do Sistema.

Definição 4.6: *Histórico* Ω^C é o conjunto de todas as observações $O^C_{(tipo)}$ geradas no intervalo de tempo Δt .

Definição 4.7: *Mensuração* é a função $M : \Omega^C \rightarrow \mathfrak{R}^N$.

Definição 4.8: *Restrição Operacional Direta* são os intervalos para cada valor em $V^C \cup H^C$ para os quais não se verifica a ocorrência de um ou mais defeitos no componente C. Dessa

forma, a restrição será o conjunto P^C dos valores $P^C = (P_{V_1}^C, P_{V_2}^C, \dots, P_{V_n}^C, \dots, P_{H_1}^C, \dots, P_{H_n}^C)$ onde $P_{V_i}^C \subset \mathfrak{R}$ e $P_{H_j}^C \subset \mathfrak{R}^m \rightarrow \mathfrak{R}^n$ para os quais não se verifica $G \rightarrow G'$.

Definição 4.9: *Restrição Operacional Indireta* são os intervalos da mensuração para os quais não se verificam a ocorrência de um ou mais defeitos no componente C. Dessa forma, a restrição Operacional Indireta será o conjunto $Q^C \subset \Omega^C \rightarrow \mathfrak{R}^N$.

Definição 4.10: *Sintoma Direto* é a máquina de Turing Γ que decide se $O^{C(\text{tipo})} \notin P^C$ para determinado componente. Caso a observação não seja restrição operacional, é dito que foi verificado um sintoma.

Definição 4.11: *Sintoma Indireto* é a máquina de Turing Λ que decide se o conjunto de entrada $O^C \notin Q^C$ para determinado componente.

Definição 4.12: **Síndrome** é o conjunto Σ de sintomas diretos e indiretos verificados em mesmo R. Para mesmo componente, pode ter vários sintomas.

Definição 4.13: Ao conjunto de Síndromes distintas para dado componente, dá-se o nome D_i de **Diagnósticos de C** em R. Diagnóstico é o conjunto de Sintomas que definem o mesmo defeito em C.

Definição 4.14: O problema diagnóstico é o **processo de se identificar** D_i associado a X (defeito) a partir do conjunto M e Ω^C obedecendo aos princípios de eficiência e eficácia.

Definição 4.15: A **aplicabilidade** é a validade de um conjunto de diagnósticos para um determinado SCD. Diferentes SCD podem ter o mesmo conjunto de Síndrome, porém possuir diagnósticos completamente diferentes. Tal situação é ilustrada por um exemplo médico: Um paciente com 50 anos de idade, obeso, queixando-se de dor precordial poderá ter o seguinte diagnóstico: “dor precordial por provável insuficiência coronariana”. Já uma paciente com 32 anos de idade, com pouca expressão facial, voz vagarosa, queixando-se de uma dor precordial que se iniciou há uma hora poderá ter como diagnóstico “uma mulher jovem, deprimida, com uma provável dor precordial de origem psicossomática”. Embora as queixas sejam as mesmas, os diagnósticos foram muito diferentes para ambos os pacientes.

12.3 As diferentes Classes de diagnóstico

O intervalo de eventos que cobre desde a geração do histórico até a identificação de um diagnóstico pode ser organizado nas seguintes etapas:

- Aquisição
- Tratamento
- Interpretação

Tal divisão em etapas no processamento dos observáveis de um sistema S-Paciente não é novidade. Algo parecido já foi visto no capítulo nove, e, na verdade, pode parecer que estes elementos sejam comuns à vários processos outros que não o diagnóstico. Porém, será observado que a decomposição do processo do diagnóstico nestas etapas é de suma importância para se definirem as classes de diagnósticos bem como seu alcance.

A razão disto reside no fato que as diferenças presentes em cada etapa geram classes de processos diagnósticos diferentes. Os dois próximos tópicos a seguir discutem tal fato.

12.3.1 Quanto à aquisição de dados: *troubleshooting* e diagnóstico auxiliado por sensores.

A etapa da aquisição engloba a obtenção dos dados descritos nas definições 4, 5 e 6. Segundo SOCHOROV (2000) O diagnóstico por seqüência de teste ou *troubleshooting* é a construção, por parte da equipe de manutenção, da seqüência de testes *in loco* e execução de ações que definam a Síndrome e corrijam o defeito presente. O *troubleshooting* parte do pressuposto de que é necessária a intervenção humana para a realização dos testes de modo a se isolar a síndrome. O critério para o estabelecimento da ordem em que os testes serão realizados é o custo associado às ações necessárias. Portanto a sessão de *troubleshooting* deve isolar o defeito de forma a minimizar o custo.

Já no Diagnóstico auxiliado por sensores, as etapas definidas acima têm suas execuções realizadas, ou auxiliadas, por uma rede de sensores presentes nos componentes, e que realizam as observações e, dependendo da capacidade de processamento desta rede, o processamento dos sintomas. Essa abordagem tem a vantagem de propiciar o pré-tratamento

dos dados, permitindo, quando da manutenção, que grande volume de processamento já tenha sido realizado.

12.3.2 Quanto à interpretação: diagnóstico preditivo e diagnóstico corretivo

No diagnóstico preditivo, a verificação do diagnóstico para um determinado componente é estabelecida deslocando-se o instante de tempo a ser analisado para um intervalo futuro desejado, mediante função de predição a partir do histórico representativo. Já no diagnóstico corretivo, o instante de tempo que determina o processo de decisão é o presente, são os componentes falhos quando são detectados.

13 Estudo de Caso

O estudo de caso desenvolvido visa mostrar a aplicabilidade da Modelagem de Sistemas Complexos Diagnosticáveis em uma situação concreta de projeto. A área de aplicação, neste estudo, foi o setor bancário, mais especificamente a utilização dos conceitos desenvolvidos neste trabalho na questão da manutenção de ATMs bancários.

O processo de manutenção de um parque de ATMs é um processo extremamente trabalhoso para qualquer fornecedor de serviços de manutenção para esse tipo de equipamento: de um lado, existe a demanda do cliente por uma alta taxa de disponibilidade dos “caixas eletrônicos”, do outro está o esforço do grupo responsável pela manutenção de lidar, com os recursos disponíveis, de um parque heterogêneo (diferentes tipos de máquinas) servindo à clientes também heterogêneos (diferentes bancos).

Por outro lado, dependendo do problema apresentado, a manutenção de um ATM em si já representa uma operação trabalhosa, não apenas pela questão de solucionar o problema técnico do equipamento, mas pela própria logística envolvida para a realização desta atividade.

A razão disso reside no fato de que ATMs são, sob uma ótica bem simples, pequenos cofres em áreas públicas; logo, advém daí todas as implicações possíveis referentes à segurança, não só do equipamento, mas também da própria equipe envolvida na manutenção.

Juntando-se a esse fato, temos que a evolução natural dos ATMs é embutir um número cada vez maior e diferente de funcionalidades. Em vista disso, é possível concluir que um ATM entra na categoria dos equipamentos onde a utilização dos princípios de diagnóstico tem relevância.

13.1 O Histórico do projeto SRD

A questão básica passa a ser, portanto, a seguinte: é possível diagnosticar, ao menos em parte, um ATM bancário com uma solução genérica, ou seja, que tenha como característica ser independente do equipamento diagnosticado?

A origem do projeto veio na forma de uma demanda da Empresa Diebold Procomp por um sistema que trabalhasse em paralelo com seu sistema de monitoração, que tem como função administrar a alocação das equipes técnicas de manutenção.

O SRD (Sistema Remoto de Diagnóstico) é uma aplicação *stand-alone* de auxílio à tomada de decisões de manutenção via diagnóstico remoto, no qual é realizada uma análise das possibilidades de falha em um conjunto de ATMs (Auto Teller Machines) monitorados. Para responder a pergunta inicial deste capítulo, o sub-sistema do ATM inicialmente escolhido para ser diagnosticado será o dispensador de numerário do equipamento.

Os objetivos do SRD são:

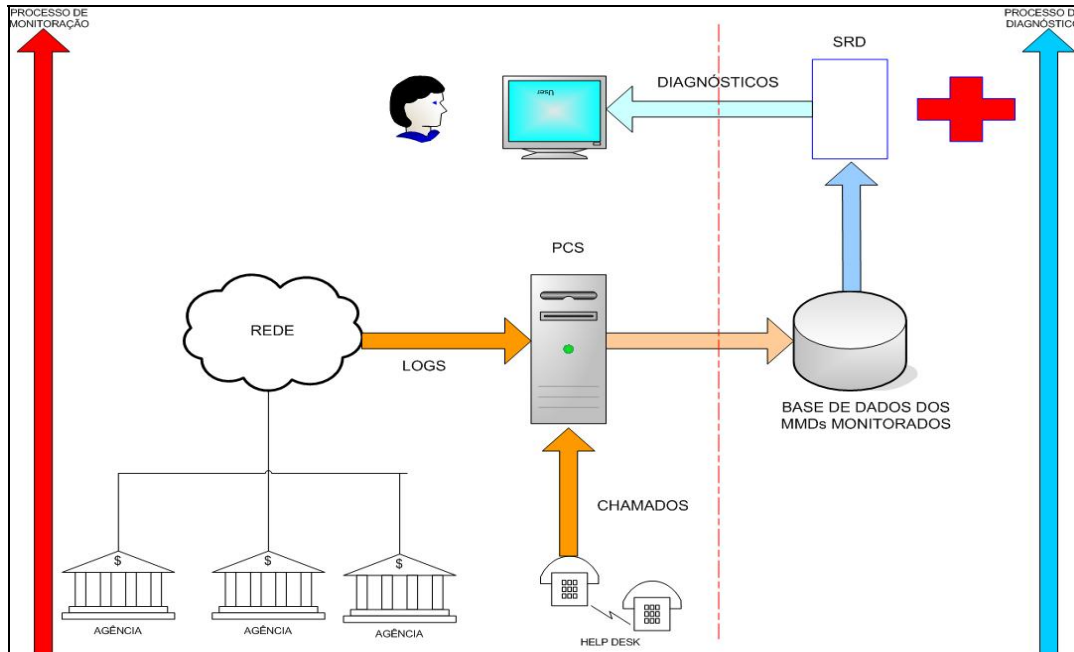
- Proporcionar diagnóstico para o módulo dispensador de numerários que compõem o ATM.
- Auxiliar a equipe de monitoração quanto às tomadas de decisões que dizem respeito ao processo de manutenção.

Para cumprir essas metas, o SRD providenciará as seguintes informações:

- Estado dos dispensadores monitorados (OK, Falho).
- Causa Provável.
- Relatórios de acompanhamento.

13.2 A Arquitetura Global da Solução

Segue o diagrama que apresenta de maneira global a solução SRD, bem como o contexto de integração do sistema.



13.3 Análise do S-Paciente: O Dispensador de Numerário

O Dispensador de cédulas é o módulo do ATM que tem por objetivo disponibilizar ao cliente do “caixa eletrônico” o valor solicitado. O dispensador guarda, em seu interior, as cédulas a serem disponibilizadas através das transações entre os clientes e o ATM. A função do dispensador, em resumo, é retirar, do total de cédulas disponíveis, o valor pedido, e entregá-lo ao usuário do ATM.

Retirar as cédulas do seu “depósito” interno, fazer a contagem das cédulas para o valor solicitado, transportá-las e, por fim, entregá-las: cada uma dessas etapas realizadas pelo dispensador precisam ser verificadas e confirmadas de modo que todas sejam realizadas sem falhas. Falha em alguma das etapas descritas no início do parágrafo resultaria desde a não entrega pura e simples do dinheiro ao cliente no final da transação até a entrega de uma quantia incorreta, gerando prejuízos e insatisfação da parte do cliente do ATM.

O dispensador analisado neste trabalho consiste no dispensador MMD de *Multi Media Dispenser*, pois, além de cédulas de dinheiro, ele também pode trabalhar com outros elementos, tais como envelopes e *tickets*. Deve-se salientar que esse trabalho não é o primeiro a utilizar este equipamento na pesquisa de diagnóstico de falhas, outros trabalhos, como em AGUIAR (2001) também fizeram uso deste módulo em particular.

Cabe agora responder as seguintes perguntas: um dispensador MMD pode ser modelado como um sistema complexo diagnosticável? Quais, em uma primeira análise, podem ser as falhas passíveis de ocorrer nesse módulo que prejudiquem o desempenho do ATM?

Em uma primeira análise, a resposta parece ser sim. O Dispensador de Cédulas é um equipamento que contém cinco motores, nove correias, vários eixos móveis, *drivers* de potência para os motores, sensores e uma placa controladora. Enfim, são mais de cem componentes distribuído em mais de cinco blocos.

O próximo passo consiste, portanto, em elaborar uma modelagem do dispensador MMD como um sistema complexo diagnosticável. Para isso, primeiro é necessário uma determinação dos componentes e subsistemas do modelo e, conseqüentemente, montar os graus de hierarquização.

A divisão estrutural do MMD é dada a seguir:

- Conjunto de cassetes
- Alimentador
- Correia
- Sensores
- *Presenter*
- Placa controladora

Desses seis itens, os quatro primeiros são componentes e os dois últimos, subsistemas. Uma descrição de cada item será realizada nos próximos parágrafos. Entre outras coisas, tal descrição dará mais detalhes sobre o funcionamento do depositário.

Conjunto de Cassetes

Cada cassete é composto por um estojo onde são armazenados os numerários a serem entregues. Um dispensador possui no máximo quatro cassetes, discriminados por identificadores lógicos. Os quatro cassetes são associados aos alimentadores. Existe ainda mais um cassete, denominado de cassete de rejeição. Ele fica localizado entre o Dispensador e o Presenter e é para este cassete que vão parar as cédulas que apresentam problemas ou cujo processo de contagem apresentou erro, ocasionando o cancelamento da mesma.

Alimentador

O alimentador é o canal físico por onde se encaixam os cassetes. Cada cassete é colocado sobre um módulo alimentadore. Cada dispensador possui no máximo quatro alimentadores, também identificados logicamente, e que são associados aos cassetes. É através do alimentador que o contador tem acesso aos numerários.

Correias e sensores

As correias são responsáveis pelo transporte das cédulas até o presenter. Os sensores são responsáveis por monitorar as operações de contagem das cédulas que saem dos cassetes.

Os dois últimos itens, o presenter e a placa controladora são, por definição, subsistemas, portanto podem ser mais elaborados em sua modelagem. A seguir será apresentada uma descrição desses dois subsistemas.

13.3.1 O *presenter* e a placa controladora

O *presenter* tem a função de apresentar as cédulas contadas para o usuário do ATM. Sua divisão estrutural é apresentada abaixo:

- Cam
- *Push Plate*
- *Shutter*
- Sensores
- Correias

Para esta modelagem, os cinco itens acima são definidos como componentes do subsistema *presenter*.

O cam

O cam consiste em um conjunto eletro-eletrônico, responsável pelas funções de empilhamento, transporte (com o auxílio das correias) e entrega das cédulas contadas.

O push plate e o shutter

O Push plate é o mecanismo que “empurra” as cédulas empilhadas até o local de saída. Já o *shutter* é composto por um acesso composto por uma porta controlada: quando acionado, libera a saída das cédulas para o usuário, porém evitando a introdução de qualquer objeto externo vindo em sentido contrário.

Os sensores e as correias do *presenter*

Os sensores do *Presenter* têm dupla finalidade: detectar a passagem das cédulas pelo *presenter* e verificar se as cédulas foram entregues ao usuário.

A Placa Controladora utiliza um microcontrolador 8051 da Intel e tem como função controlar o MMD.

13.4 A aplicação do modelo SCD para o diagnóstico do MMD

O objetivo do nosso estudo de caso é derivar um SGDA, ou seja, um sistema de diagnóstico que seja genérico. Para isso, dispomos do modelo de Sistemas Complexos Diagnosticáveis que aplicaremos com o intuito de obtermos o nosso “diagnosticador” flexível.

A solução para capacitar o sistema de diagnóstico de tal flexibilidade é torná-lo programável. Para tanto, o modelo SCD servirá de base para o desenvolvimento de uma linguagem *script* possível de ser editada pelo usuário e um formato de arquivo de dados, onde estarão contidas informações relevantes de cada diagnóstico descrito (ações corretivas, por exemplo). Desta forma, o usuário terá de criar dois elementos para o seu sistema de diagnóstico: o seu *script* de diagnóstico, interpretável pelo sistema e sua base de dados, tais elementos são denominados respectivamente, **arquivo de parâmetros** e **arquivo de mensagens**.

A grande vantagem em gerarmos uma linguagem interpretável, é que sua utilização isola o usuário responsável pela elaboração do script dos aspectos formais do modelo SCD, tornando muito mais intuitivo, pela característica computacional do arquivo de parametrização, a geração de um conjunto de diagnósticos.

Para tanto, temos que ter em mãos todos os elementos que fazem parte do modelo. Derivar os elementos do modelo a partir de uma situação real (no caso o SRD) é a prioridade e o início está justamente em determinar se é possível obter o elemento básico para a determinação da presença ou não de um sintoma: o histórico.

13.5 O Histórico

O histórico para o processamento de sintomas é obtido do próprio MMD. O dispensador possui sensores que registram uma dezena de informações relativas ao estado da máquina. Essas informações são gerenciadas por um servidor, chamado de servidor de log, que as grava em um arquivo log.

Muito dos dados gravados pelo servidor dizem respeito à informações que não são pertinentes ao diagnóstico, como, por exemplo, a versão do sistema operacional. Desta forma, um conjunto de filtros varre o log disponível e gera um segundo tipo de arquivo, denominado arquivo de filtro, que é baseado nos dados do arquivo de log.

Tal arquivo de filtro consiste em nosso histórico. Como visto na definição 4.6, um histórico é composto por um conjunto de observações. Existem cinco tipos diferentes de observações passíveis de serem geradas, presentes em um arquivo de filtro. Cada uma delas pode surgir com maior ou menor frequência em função das diferentes etapas do funcionamento do dispensador, bem como dos eventos que podem possivelmente acontecer. Cada observação gera uma linha no arquivo de filtro (AGUIAR, 2001). As classes de observação são dadas abaixo:

Classe STADISP e classe STAPRES

Registra o estado do equipamento como um todo. Possui informações como o estado do contador, informação sobre rejeição de cédulas, etc. A classe STAPRES registra o estado do presenter.

Classe STINT e classe CONFCAS

A classe STINT registra os estados internos do contador, mostrando diversas estatísticas sobre seu estado de operação. A classe CONFCAS registra a configuração dos cassetes e seus identificadores.

Classe LOGLINE

A classe LOGLINE registra as operações efetuadas no equipamento. Este registro armazena um ciclo completo de dispensa de numerários, ou seja, *contagem e entrega* ou *contagem e rejeição* das cédulas. Esta linha registra informações de códigos de erros retornados pelos subsistemas do dispensador.

13.6 A Sintaxe do arquivo de parametrização

Um Arquivo de parâmetros, pelo princípio da aplicabilidade descrito no capítulo dez, pode ser gerado para cada sistema diagnosticado. O Arquivo de parâmetros é constituído de quatro partes:

1. Log
2. Variáveis
3. Sintomas
4. Diagnósticos

13.6.1 Log

O bloco do arquivo referente ao Log tem como finalidade informar quais as informações do arquivo de filtro do MMD serão utilizadas no diagnóstico e qual o tamanho do histórico que será obtido deste filtro.

13.6.1.1 Sintaxe do bloco Log

Esta parte do arquivo de parametrização começa com o discriminador-título do bloco cujo formato é:

[LOG]

Abaixo deste discriminador, é declarado quais informações serão extraídas do arquivo de filtro. Tal especificação é feita declarando-se um rótulo para um dado não tipado que corresponderá a uma matriz composta por um subconjunto do histórico, e que será nomeada com o rótulo da classe das linhas do arquivo de filtro que se deseja selecionar. O número de linhas selecionáveis definidas para este rótulo, bem como as colunas destas linhas que serão relevantes para a análise, deverão estar do lado direito da atribuição. Para ilustrar esta definição temos o seguinte exemplo:

[LOG]

LOGLINE=15,99

LOGLINE	15	99
Variável declarada com um nome idêntico ao rótulo do arquivo de log.	Número das últimas linhas a serem selecionadas dentro do histórico	Número de colunas que este o dado LOGLINE deverá conter.

O lado direito da atribuição indica que serão utilizadas as últimas 15 linhas de LOGLINE, e cada linha contém 99 colunas. Vale ressaltar que cada coluna tem um significado previamente definido pelo filtro.

13.6.2 Variáveis

As variáveis são a implementação do resultado de uma mensuração do modelo SCD.

Nesta etapa são definidas as variáveis referentes às mensurações a partir das informações extraídas do arquivo de filtro (histórico) da etapa anterior. As mensurações são realizadas através de funções disponíveis na API (Application Interface) do Engine da solução. Esta etapa começa com a chave:

[VARIABLES]

Abaixo desta chave é informado o número total de variáveis a serem calculadas e depois nas próximas linhas são definidos o nome da variável e a respectiva função.

Cada final de linha deve vir acompanhado com um finalizador “|” e, após este, poderá ser inserido comentários identificados por “//”.

Exemplo:

[VARIABLES]

```
var=5|                                     // tem 5 variáveis

LErroEn= SEA LSV ERROS(15)|

LErroCd= SEA LSV ERROS(16)|

TxErroCCA= SEA TAX [50,0] LOGLINE(15)=196735|      //taxa de erros CCA

TxErroAI1= SEA TAX [50,0] LOGLINE(15)=12648575| //taxa de erros Alimentador 1

TxErroAI2= SEA TAX [50,0] LOGLINE(15)=127|        //taxa de erros Alimentador 2
```

TxErroAI2=	SEA TAX [50,0] LOGLINE(15)=127	//taxa de erros Alimentador 2
Nome da variável	Função utilizada	Linha de comentário

13.6.3 Sintomas

Nesta parte do arquivo são definidos os sintomas que definem um diagnóstico. Esta etapa começa com a chave:

[SINTOMAS]

Abaixo desta chave é informado o número total de sintomas a serem avaliados.

Nas próximas linhas são declarados o sintoma, a variável e o limiar adotado. Cada limiar corresponde a uma restrição operacional

Cada final de linha deve vir acompanhado com um finalizador “|”, e ser seguido de comentários identificados por “//”.

Exemplo:

[SINTOMAS]

sint=11| // tem 11 sintomas

S1:Tx1>0.02|

S2:Tx2>0.02|

S3:Tx3>0.02|

S4:Tx4>0.02|

S5:Tx5>0.02|

S6:LErroEn=196735|

S7-1:LErroEn=12648575|

S7-2:LErroEn=12714111|

S7-3:LErroEn=12779647|

S7-4:LErroEn=12845183|

S7-5:LErroEn=12910719| // verifica se a variável ErroEn é igual a 12910719.

S7-5:	LErroEn=12910719	// verifica se a variável ErroEn é igual a 12910719
Nome do sintoma	Variável utilizada = valor	Linha de comentário

13.6.4 Diagnóstico

Nesta parte do script é definido cada diagnóstico através da associação de sintomas. Esta etapa começa com a chave:

```
[DIAGNOSTICO]
```

Na primeira linha indica o número de linhas que contém descrição dos diagnósticos. Abaixo desta é descrito a opção do tipo de busca diagnóstica a ser realizada, que pode ser:

- Unico: devolve apenas o primeiro diagnóstico encontrado.
- Completo: devolve todos os diagnósticos.

Cada final de linha deve vir acompanhado com um finalizador “|” e, após este, poderá ser inserido comentários identificados por “//”.

Segue um exemplo:

```
[DIAGNOSTICOS]
```

```
diag=4| // quatro linhas de diagnostico
```

```
unico // tipo de diagnóstico
```

```
D1=S1|
```

```
D2=S2;S6| // o D2 consiste do sintoma S2 e S3
```

```
D3=S3| // o D3 consiste do sintoma S3 ou S4
```

```
D3=S4|
```

Associação “E” lógica na declaração do diagnóstico:

No campo de diagnóstico basta declarar o diagnóstico e especificar os sintomas separando-os por “;”.

Exemplo:

```
[DIAGNOSTICOS]
```

```
diag=2
```

```
unico
```

```
D1=S1|           // o diagnóstico D1 consiste apenas do sintoma S1
```

```
D2=S2;S1|       // o diagnóstico D2 consiste do sintoma S1 e S2
```

Associação “ OU” lógica na declaração do diagnóstico:

No campo de diagnóstico deve-se declarar o diagnóstico e especificar um sintoma e em outra linha declarar o mesmo diagnóstico associado ao outro sintoma. Exemplo:

```
[DIAGNOSTICOS]
```

```
diag=2
```

```
unico
```

```
D1=S1|           // o diagnóstico D1 consiste apenas do sintoma S1
```

```
D2=S2|
```

```
D2=S3|           // o diagnóstico D2 consiste do sintoma S3 ou S2
```

13.7 A Sintaxe das Mensurações implementadas

A sintaxe de implementação das mensurações segue um padrão já estabelecido para o projeto de pesquisa e desenvolvimento SAD (Sistema Automático de Diagnóstico) feito em uma parceria acadêmica com a empresa Diebold Procomp. Tal padrão foi primeiramente implementado pela equipe de pesquisa em Diagnóstico da Escola Politécnica da USP.

A definição das mensurações a serem utilizadas na elaboração dos scripts segue a seguinte sintaxe:

```
<NOME> = <CLASSE> <SUB_CLASSE> <PARÂMETROS> <EXPRESSÕES>
```

Onde NOME é o rótulo do dado não tipado que receberá o resultado do processamento da mensuração sobre o subconjunto do histórico em análise. CLASSE corresponde à classe de mensuração realizada. Por exemplo, podemos ter uma mensuração que corresponda à uma comparação entre dois valores. Porém, uma comparação pode ser realizadas de diversas maneiras. O detalhamento de uma classe de mensuração é dada pela SUB_CLASSE. Desta forma, uma comparação para determinar o menor valor presente em alguma observação do histórico, especifica a mensuração de comparação, e constitui um exemplo de SUB_CLASSE. O bloco PARÂMETROS especifica os parâmetros de entrada e de referência para a mensuração, e o bloco EXPRESSÕES contém todas as configurações lógicas e aritméticas necessárias para a execução da mensuração.

A seguir, serão apresentadas as mensurações implementadas até o momento para tratar os possíveis diagnósticos.

Mensuração de Estimação de Taxas

Dados dois vetores R e C, a taxa de R em C é definida como:

$$\frac{\sum_{i=1}^m R(i)}{\sum_{i=1}^m C(i)}$$

Desse modo, para se calcular uma taxa, é necessário especificar os vetores R e C, seus respectivos valores de m e o passo que deve ser seguido para obter o vetor de saída. Caso o passo seja zero, teremos apenas a taxa das operações mais recentes na janela.

Classe	EST
Sub-classe	TAX
Parâmetros	Janela e Passo, no formato: [Janela,Passo] Obs.: quando o passo for zero, será calculada apenas a janela mais recente.
Expressões necessárias	Vetor R e Vetor C, no formato: R,C
Modelo	EST TAX [Jan,Passo] R,C

Exemplo: Parametrizar a taxa de rejeição mais recente do alimentador 1 de um dispensador de cédulas, sobre uma janela de quarenta operações. Sabe-se que a quantidade de cédulas contadas está armazenada na primeira coluna da linha DISPENSA e a de rejeições na quinta coluna da mesma linha.

Tx1= EST TAX [40,0] DISPENSA(4),DISPENSA(0)|

Mensuração de Busca e calculo de taxas

Dado um vetor R, esta função determina qual a taxa de ocorrência de um determinado evento neste vetor. O evento pode ser a componente do vetor ser igual, diferente, maior ou menor que um determinado valor. Desse modo, para calcularmos uma taxa devemos especificar os vetor R, sua quantidade de componentes (janela) e o passo. Caso o passo seja zero, teremos apenas a taxa das operações mais recentes na janela.

Classe	SEA
Sub-classe	TAX
Parâmetros	Janela e Passo, no formato: [Janela,Passo] Obs.: quando o passo for zero, será calculada apenas a janela mais recente.
Expressões necessárias	Vetor R, no formato <i>R_{signal}Valor</i> Sinal pode ser: = → igual != → diferente > → maior < → menor
Modelo	SEA TAX [Jan,Passo] R=5

Exemplo: Parametrizar a taxa de ocorrência de erros nas trinta últimas operações de entrega. Sabe-se que os erros são indicados por valores diferentes de zero na sexta coluna da linha *PRESENT*.

TxErro= SEA TAX [30,0] PRESENT(5)!=0|

Mensuração de Busca e montagem de uma matriz

Monta uma nova matriz baseada em valores de uma determinada coluna de uma matriz dada.

Classe	SEA
Sub-classe	PRM
Parâmetros	Linhas, no formato: [Linhas]
Expressões necessárias	Matriz R, no formato R(col) <i>senal</i> Valor <i>Senal</i> pode ser: = → igual != → diferente > → maior < → menor col será uma das colunas da matriz
Modelo	SEA PRM [Jan] R(col)=5

Exemplo: Montar uma matriz das últimas trinta as operações de erro de contagem. Os erros de contagem são registrados na décima coluna da matriz *CONTAGEM*.

ERROS= SEA PRM [30] CONTAGEM(9)!=0|

Mensuração de Busca do valor mais recente

Busca o valor mais recente de uma determinada linha do histórico.

Classe	SEA
Sub-classe	LSV
Parâmetros	Vetor R, no formato R(col) col será uma das colunas da matriz
Expressões necessárias	<i>NENHUMA</i>
Modelo	SEA LSV R(col)

Exemplo: Buscar o último Status de cassete disponível. Esta informação é armazenada na segunda coluna da linha *STATUS* do log.

ERROS= SEA LSV STATUS(1)|

13.8 A Sintaxe do Arquivo de Mensagens

O arquivo de mensagens é formado a fim de permitir a localização das descrições das causas associadas a cada um dos diagnósticos fornecidos pelo SRD. O arquivo de mensagens é uma base de dados que centraliza informações de ações corretivas e outras.

Deve haver uma relação unívoca entre os diagnósticos descritos na seção diagnósticos do arquivo de parametrização e os diagnósticos descritos no arquivo de mensagens. O SRD encerra sua busca no arquivo de mensagens quando encontra o primeiro diagnóstico com o mesmo nome do diagnóstico descrito no arquivo de parâmetros. Abaixo temos um exemplo de descrição de um diagnóstico presente no arquivo de mensagens:

```
[D1]
percentage=90%|
entity=7500|
recovery_action_type=actual_text|
recovery_action= actual_troque X|
link_type=text|
link=c:\manuais>manual do modulo X|
END
```

Segue a descrição de cada ítem:

[D1]

É a identificação do diagnóstico, que consta tanto no arquivo de mensagem como no de parâmetros.

Esta linha de identificação é a única que não deve ser seguida de comentários ou espaços em branco. Após o símbolo “[] ” devemos ter um retorno.

percentage

Define a porcentagem ou o peso para cada *recover action*

entity

Identifica o componente do equipamento diagnosticado.

recovery_action

É a mensagem de recovery_action, ou seja, a ação corretiva que anula a presença de um determinado defeito. Uma situação especial pode acontecer quando um determinado diagnóstico possui mais de um recovery action, neste caso, os vários recoveries devem ser separados por “;”. Como no exemplo abaixo:

```
[D5-4]
percentage=p1%;p2%|
entity=7502|
recovery_action_type=actual_text|
recovery_action=m1;m2|
link_type=actual_text|
link=l1;l2|
END
```

Para este diagnóstico, temos a mensagem m1 relacionada com o percentual p1 e o link l1. O mesmo padrão deve ser seguido pelas mensagens de warning.

Devem ser colocadas no máximo dez mensagens de recovery ou warning em um mesmo diagnóstico.

A quantidade de mensagens descritas deve ser a mesma da quantidade de percentuais e de links descrito. Além disso, a quantidade de mensagens de warning também deve ser a mesma da quantidade de links de warning.

recovery_action_type

Indica os tipos de retorno para o campo recovery_action

link

Fornece o link para a documentação do respectivo recovery_action

link_type

Indica os tipos de retorno para o campo link

As outras linhas indicam, diretamente, os valores que devem ser preenchidos na estrutura de retorno do diagnóstico. Todas elas devem ser encerradas pelo símbolo `|` e podem ser seguidas de comentários.

Para as linhas descritas terminadas em `_type`, os tipos permitidos são:

`actual_text`, quando a mensagem for direta

`reference_link`, quando o que se fornece é um link para um arquivo de recursos com a descrição das mensagens.

13.9 O Processo de aquisição do conhecimento

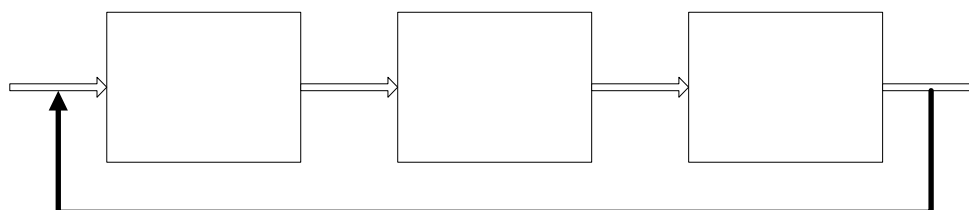
A representação do conhecimento acerca do domínio do problema implica a compreensão deste domínio. Isso significa que antes de um arquivo de parâmetros e de mensagem serem elaborados, o modelo do equipamento, do ponto de vista do diagnóstico, precisa ser analisado.

Como foi dito antes, a sintaxe do script ajuda a atenuar o formalismo do modelo, porém, a sintaxe da linguagem script também possui um certo grau de complexidade. Para isso foi desenvolvido o Wizard, um aplicativo *stand-alone* que auxilia o técnico na elaboração do script, isolando-o dos detalhes de sintaxe da linguagem. De modo que este só se preocupe com o processo de levantamento dos diagnósticos do equipamento.

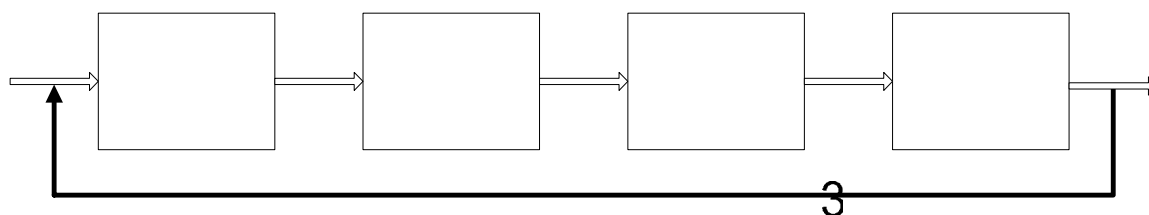
O desenvolvimento de um arquivo de parametrização (script) e de mensagens para um determinado equipamento, antes de sua implementação propriamente dita, seja através do Wizard ou não, necessita de algumas etapas para o amadurecimento da solução, estas etapas estão citadas abaixo:

1. Identificar os módulos (componentes) que podem ser trocadas.
2. Verificar quais componentes são monitorados por sensores.
3. Criar um formato de filtro.
4. Pesquisar e simular as possíveis falhas.
5. Verificar o quais eventos operacionais do componente são gravados.
6. Definir variáveis.
7. Definir os sintomas.
8. Definir os diagnósticos

Estas etapas compõem três grandes processos na construção de diagnóstico para os componentes: Processo e validação, Processo de construção de um diagnóstico e Processo de construção de novos diagnósticos. Estes processos estão descritos abaixo, na forma de diagrama de blocos:



Processo de Validação



Processo de Construção de novos diagnósticos

Segue abaixo uma descrição, em formato tabular, de cada uma das etapas citadas:

Etapa 1	Identificar os componentes que sofrem manutenção.
Descrição	<ul style="list-style-type: none"> • Uma análise detalhada do módulo permitirá avaliar quais são os componentes do periférico que sofrem manutenção em campo, ou que podem ser trocados pelo técnico. Além disso, podem ser identificados alguns procedimentos adotados em campo.
Entrada	<ul style="list-style-type: none"> • Documentos com a descrição física do dispositivo a ser diagnosticado • Referências da Assistência Técnica
Saída	<ul style="list-style-type: none"> • Possíveis entidades para diagnóstico

Etapa 2	Verificar se os componentes são monitorados.
Descrição	Identificar como o comportamento dos componentes identificados no passo anterior é monitorado pelo firmware do periférico e como eles são logados.
Entrada	Saída do item 1
Saída	<ul style="list-style-type: none"> • Lista com os componentes monitorados • Lista com os componentes não monitorados

Etapa 3	Criar um formato de filtro.
Descrição	Sabendo quais códigos de retorno estão relacionados com as entidades selecionadas (componentes a serem diagnosticados), O desenvolvedor pode fazer um filtro para o periférico selecionando apenas as informações relevantes.
Entrada	<ul style="list-style-type: none"> • Modelo de Filtro do CADs
Saída	<ul style="list-style-type: none"> • Documentação descrevendo o filtro • Implementação do filtro

Etapa 4	Pesquisar/Simular as possíveis falhas.
Descrição	Exercitar o equipamento para verificar quando (e em quais condições) este apresenta determinado defeito. Além disso, o desenvolvedor pode executar uma validação parcial da árvore de falhas.
Entrada	<ul style="list-style-type: none"> • Árvore de falhas • Manual de Troubleshooting • Assistência Técnica • Descrição Física do Hardware • Ambiente de testes
Saída	<ul style="list-style-type: none"> • Procedimento de testes (como gerar defeitos no componente)

Etapa 5	Verificar o que é gravado em log.
Descrição	Exercitar o equipamento para verificar se quando determinado defeito esta presente, o filtro grava as informações necessárias para detectar este comportamento.
Entrada	<ul style="list-style-type: none"> • Ambiente de testes • Filtro implementado
Saída	<ul style="list-style-type: none"> • Arquivos de filtro com situações de defeito e/ou desajustes

Etapa 6	Definir variáveis.
Descrição	Definir variáveis que identificam (ou podem identificar) possíveis defeitos. Definir algoritmos que processando determinadas variáveis também podem identificar possíveis defeitos.
Entrada	<ul style="list-style-type: none"> • Análise dos ensaios do item 5 • Lista das entidades monitoradas(item 2)
Saída	<ul style="list-style-type: none"> • Lista de variáveis de LOG • Lista de variáveis processadas (funções e parâmetros)

Etapa 7	Definir os sintomas.
Descrição	Definir limiares (restrições operacionais) para as variáveis do item anterior ou critérios para os algoritmos para determinar se determinado sintoma esta presente ou não.
Entrada	Resultados das etapas 4, 5 e 6
Saída	Lista de sintomas (baseado no item 6)

Etapa 8	Definir os diagnósticos
Descrição	Definir as síndromes, ou seja, o conjunto de sintomas que caracteriza determinado defeito, e, portanto, define qual ou quais entidades estão defeituosas.
Entrada	<ul style="list-style-type: none"> • Lista das entidades monitoradas. • Resultado das etapas 6 e 7. • Assistência Técnica (mensagens, interface com técnico) • Histórico de Diagnósticos da Assistência Técnica
Saída	<ul style="list-style-type: none"> • Lista dos diagnósticos • Arquivo de mensagens (mensagens, porcentagem). • Arquivo de parâmetros.

13.10 Validação

Foram gerados quarenta e quatro diagnósticos para o módulo MMD a partir dos manuais de Troubleshooting da Diebold. Em uma avaliação preliminar foram submetidos à análise vinte arquivos de filtro com situações em que o equipamento se encontrava falho e em situação onde o MMD estava funcionando perfeitamente. Todos os arquivos de filtro geraram detecções positivas no SRD, não ocorrendo nenhuma situação de perda ou falso alarme.

14 Conclusões

O diagnóstico de falhas é uma área bem ativa de pesquisa. Atualmente, existe um esforço dos pesquisadores das diferentes vertentes em buscar uma comparação entre os diferentes modelos de diagnóstico de falhas, para analisarem seus pontos convergentes e divergentes. Enunciar, de forma exaustiva, todos os modelos sob pesquisa hoje é uma tarefa árdua em função da grande variedade de teorias, mesmo dentro de um mesmo grupo.

Este trabalho se propôs a sugerir um modelo que possibilitasse o desenvolvimento de sistemas genéricos de diagnóstico. O modelo proposto se adequou à situação experimental de diagnóstico para o MMD, mostrando ser possível aplicá-lo em um processo configurável.

O módulo MMD se comportou como um SCD ao ser analisado: de maneira independente de qualquer pré-requisito que não fossem os elementos que compõem a definição do próprio SCD. Para a etapa inicial do SRD está sendo considerado um volume no intervalo de setenta e cinco (75) até dez mil (10.000) módulos dispensadores monitorados.

Existe um bom número de extensões possíveis para esse trabalho: um modelo fechado para diagnóstico preditivo que possa utilizar a configuração via *script* é um deles. Analisar se, para isso, deverão ser incluídos novas funções na API do CADs para possibilitar a manipulação de modelos de predição (utilizando séries temporais, por exemplo) é um outro exemplo.

Outro ponto para futuros estudos e extensões é um melhor tratamento, por parte do modelo, para as situações que envolvam incerteza. Um detalhamento maior nas relações de incerteza que envolva as associações entre causa raiz e sintomas pode ser averiguada.

Por último, fica a questão das situações de diagnóstico que envolve múltiplas falhas e testes não-binários, bem como a possibilidade de utilizar uma abordagem RBC para representar melhor o modelo de diagnóstico baseado em SCD. Todos afetam diretamente a estrutura do modelo.

Tais assuntos não foram tratados no presente trabalho. Existe uma preocupação muito grande dos pesquisadores acerca desses tópicos. Esses pontos, se tratados, enriqueceriam o modelo e incrementariam a potencialidade de cobertura contra falhas em diferentes sistemas.

15 Referências bibliográficas

- AAMODT, A., PLAZA, E. **Case-based reasoning: Foundational issues, methodological variations, and system approaches**. *AI Communications* 7(1), 1994.
- AGUIAR, M. A. **Diagnóstico de falhas em equipamentos usando técnicas de Redes Neurais Artificiais**. Dissertação de Mestrado, 2001, Escola Politécnica da USP.
- AGUIAR, M. A., KOVACS, Z. L. DESTRO, R. C., PELAEZ, J. R., SIMÕES, M. G. (2001) **Predictive Maintenance Oriented Neural Network System – PREMON** *In: IECON'01*, 2001, Denver, Colorado. *Anais do IECON'01*.
- BALAKRISHNAN, K. HONAVAR, V. **Intelligent Diagnosis Systems**, to appear in the *Journal of Intelligent Systems*, 1998.
- BARROWS, H. S., PICKELL G. C. **Developing clinical problem-solving skills: a guide to more effective diagnosis and treatment**. New York, W.W. Norton & Co., 1991; 1-216.
- BENJAMINS, V. R., JANSWEIJER, W. **Toward a Competence Theory of Diagnosis**, *IEEE Expert*, 9(5), pp. 43-52, October, 1994.
- BERGE, C., **Hypergraphs: Combinatorics of Finite Sets**, The Netherlands: Elsevier-North Holland, 1989.
- BERNARD, D. E.; DORAIS, G. A.; FRY, C.; GAMBLE Jr., E. B.; KANEFISKY, B.; KURIEN, J.; W.; MILLAR, N.; MUSCETTOLA, P. P.; NAYAK, B.; PELL, K.; RAJAN, N.; ROUQUETTE, B.; SMITH B, WILLIAMS. C. **Design of the remote agent experiment for spacecraft autonomy**. *Proceedings of IEEE Aerospace*, 1998.
- BERTALANFFY, L.Von. **General Theory of Systems**. N. York, 1969, George Braziller.
- BLANKE, M., KINNAERT, M., LUNZE, STAROSWIECKI, J. M. **Diagnosis and Fault-Tolerant Control**, Springer-Verlag, August 1, 2003, ISBN: 3540010564.
- BORDAGE, G., ZACKS, R. **The structure of medical knowledge in the memories of medical students and general practitioners: categories and prototypes**. *Med Educ* 18, 1984.
- BYLANDER, T. **Complexity of model-based diagnosis**, *Model Based Reasoning Workshop*, Detroit, MI, 1989.
- CARPANZANO, E. & MAFFEZZONI, C., **Symbolic manipulation techniques for model simplification in object-oriented modeling of large scale continuous systems**, *Mathematics and Computers in Simulations* 48, 133-150, 1998.
- CLARK, R. N. (1978). **Instrument fault detection**, *IEEE Trans. Aerospace Electron System*, 14, 456-465.
- CORDIER M.-O., DAGUE P., DUMAS M., LÉVY F., MONTMAIN J., STAROSWIECKI M., TRAVÉ-MASSUYÈS L., **A comparative analysis of AI and control theory approaches to model-based diagnosis**, *14th European Conference on Artificial Intelligence ECAI-00*. Berlin, 2000.

CORDIER, M. O. DAGUE, P. LEVY, F. MONTMAIN, J. STAROSWIECKI, MARCEL. TRAVÉ-MASSUYÈS, L. **Conflicts Versus Analytical Redundancy Relations: A Comparative Analysis of the Model Based Diagnosis Approach From the Artificial Intelligence and Automatic Control Perspectives.** IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, Vol. 34, No. 5, October, 2004.

CHANDRASEKARAN, B. **Generic tasks as building blocks for knowledge-based systems: the diagnosis and routine design examples,** The Knowledge Engineering Review 3 (3) 183-210. (1988)

CHEN, J., RON J., PATTON, R. **Model-Based Fault Diagnosis for Dynamic Systems,** Kluwer Academic Publishers, January 1, 1999, ISBN: 0-7923-8411-3.

CHIANG, L. H., RUSSELL, E. L., BRAATZ, R. D. **Fault Detection and Diagnosis in Industrial Systems,** Springer-Verlag, February 15, 2001, ISBN: 1-85233-327-8.

DA COSTA, V. A. C. **Manutenção Aeronáutica e Controle de Qualidade em Prol da Segurança de Vôo,** 2004.

DAVIS, R. **Diagnostic reasoning based on structure and behavior,** Artificial Intelligence, 1984.

DAVIS, R., HAMSCHER, W. **Model-Based Reasoning: Troubleshooting,** Exploring Artificial Intelligence, pp. 297-346, Morgan Kaufmann(ed), 1988.

DRESSLER O., STRUSS, P. **The Consistency-based approach to automated diagnosis of devices,** Principles of Knowledge Representation, Brewka (ed), 1996.

DIESTEL, R. **GraphTheory,** Springer, 1997. ISBN 0-387-98210-8.

FIJANY, A. VATAN, F. BARRETT, A. MACKEY, R. **New Approaches for Solving the Diagnosis Problem.** IPN Progress Report, 2002.

FLETCHER, R. H., FLETCHER, S. W., WAGNER, E. H. **Epidemiologia clínica: bases científicas da conduta médica.** Porto Alegre, Artes Médicas, 1989.

FRANK, P.M., **Analytical and qualitative model-based fault diagnosis – a survey and some new results,** European Journal of Control, 2, 6-28, 1996.

GASCA, R.M., DEL VALLE, C., CEBALLOS, R., TORO, M. **An integration of fdi and dx approaches to polinomial models,** DX, 2003.

GENESERETH, M. **The use of design descriptions in automated diagnosis.** Artificial Intelligence Vol 24, pp 411-436, 1984.

GERTLER ,J. **Fault Detection and Diagnosis in Engineering Systems.** Marcel Dekker, New York, May 1, 1998, ISBN: 0-8247-9427-3.

GICK, M.L., HOLYOAK, K.J. **Analogical problem solving.** Cognitive Psychology, 12, 306-355, 1980.

GOLDBARG, M. C. LUNA, H. P. L. **Otimização Combinatória e Programação Linear, Modelos e Algoritmos**. Editora Campus, 2000.

GOLDENFELD, N.; KADANOFF, L. P. **Simple Lessons from Complexity**. Science, Vol. 284, Iss. 5411, p. 87, 1999.

GRESSE, C. WANGENHEIM, A. **Raciocínio Baseado em Casos**, Editora Manole, 2003.

HAIJYEV, C., CALISKAN, F. **Fault Diagnosis and Reconfiguration in Flight Control Systems**, Kluwer Academic Publishers, October 2003, ISBN 1-4020-7605-3.

HAYKIN, S. **Redes Neurais: Princípios e Prática**. 2ª Ed; Porto Alegre: Bookman, 2001.

HEYLIGHEN, F. **Building a Science of Complexity**. Annual Conference of the Cybernetic Society. Londin, 1988.

HIMMELBLAU, D. M., **Fault Detection and Diagnosis in Chemical and Petrochemical Processes**, Elsevier, Amsterdam, Netherlands, 1978.

JUNQUERA, BP, GONZALEZ, C.A. **Possible conflicts, ARRs, and conflicts**, Proc. of the International Workshop on Principles of Diagnosis, 2003.

KASSIRER J. P., KOPELMAN, R. I. **Learning clinical reasoning**. Baltimore, Williams and Wilkins, 1991.

KLEER, J. **Problem solving with the ATMS**, Artificial Intelligence, vol. 28, pp. 197-224, 1986.

KLEER, J. de, MACKWORTH, A. K., REITER, R. **Characterizing diagnosis and systems**. Artificial Intelligence, 56:197-222, 1992.

KLEER, J. de. **Local methods for localizing faults in electronic circuits**, MIT AI Memo 394, Massachusetts Institute of Technology, Cambridge, MA, 1976.

KOLODNER, J. L. **Case-Based Reasoning**. Morgan Kaufmann, 1993.

KOVACS, Z. L. **Redes Neurais Artificiais: Fundamentos e Aplicações**. 3ª Ed. São Paulo: Livraria de Física, 2002.

KRYSANDER, M.: **Design and Analysis of diagnosis Systems Utilizing Structural Methods**, ISBN 91-7373-733-X, 2003.

LAFON, M. J., PASTOR, L. Trave-Massuy'es, B. Doyon, J.F. Demonet, P. Celsis, **Qualitative modeling of cerebral information propagation mechanisms**", *In*: Proceedings of the Third International Conference on Computational and Neurosciences, volume 2, pp. 21-23, Research Triangle Park, USA, 1998.

LAFORTUNE, S., MEERA, S. **A Discrete Event Systems Approach to Failure Diagnosis: Theory & Applications**, Eleventh International Workshop on Principles of Diagnosis (DX'00).

LASER, M. **Recent safety and environmental legislation.** Trans IchemE 78 (B), 419_422, 2000.

LASER, M. **AI and Automatic control approaches of model-based diagnosis: links and underlying hypotheses,** 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2000, Budapest, Hungary, June, 14-16, 2000.

LEWIS, L. **Managing Computer Networks: A Case-Based Reasoning Approach,** Artech House, 1995.

LUCAS, P. J. F. **Analysis of notions of diagnosis.** Artificial Intelligence 1998.

LUCAS, P. J. F. **Symbolic diagnosis and its formalization,** The Knowledge Engineering Review, 1997.

LUCAS, P. J. F. **Logic engineering in medicine,** The Knowledge Engineering Review 1995.

LUONG, M., MAQUIN, D., HUYNH, C.T., RAGOT, J. **Observability, redundancy, reliability, and integrated design of measurement systems,** 2nd IFAC Symposium on Intelligent Components for Control Applications SICICA'94, Budapest, Hungary, June 8-10, 1994.

MAGALDI, R.V. **CBR for Troubleshooting Aircraft on the Flightline.** In: *Proc. IEEE Colloquium on Case-Based Reasoning: Prospects for Applications*, Digest No: 1994/057.

MATTIAS, K., MATTIAS, N. **Structural analysis utilizing MSS sets with application to a paper plant,** 13th International Workshop on Principles of Diagnosis, 2002.

MCILRAITH, S. A. **Explanatory Diagnosis: Conjecturing actions to explain observations.** Knowledge Systems Laboratory, 1999.

MEHRA, R. K., PESHON, I. **An innovations approach to fault detection and diagnosis in dynamic systems,** Automatica, 7, 637-640, 1971.

MENDES, S., AGUIAR, T. C. **Métodos para especificação de Sistemas.** Ed Edgar Blucher, 1989.

MILLER, R. A., POPLE Jr., H. E., MYERS, J. D. **INTERNIST-1, An experimental computer-based diagnostic consultant for general internal medicine,** New England Journal of Medicine 307, 1982.

MOSTERMAN, P.J. **Hybrid dynamic systems: a hybrid bond graph modeling paradigm and its applications in diagnosis,** Phd thesis, Vanderbilt University, Nashville, Tennessee, USA, 1997.

MOZETIC I. **Model-Based Diagnosis: An Overview,** Austrian Research Institute for Artificial Intelligence 24, 1984.

MAHMOUD, M., Jiang, J., ZHANG, Y. **Active Fault Tolerant Control Systems: Stochastic Analysis and Synthesis,** Springer-Verlag, 2003, ISBN: 3540003185.

NETO, A R **Raciocínio clínico: O processo de decisão diagnóstica e terapêutica.** Revista da Associação Médica Brasileira. vol.44 n.4 São Paulo Oct./Dec. 1998.

NYBERG, M. **The polybox example using the framework of structured hypothesis tests.** Technical Report LiTH-ISY-R-2335, Linköping, Sweden, 2000.

PAL, S. K., DILLON, T. S., YEUNG, D. S. **Soft Computing Case Based Reasoning.** London, U.K.: Springer-Verlag, 2001.

PATTIPATI, K. R., DONTAMSETTY, M. **On a Generalized Test Sequencing Problem.** IEEE Transactions on Systems, Man and Cybernetics 22(2), 392-396, 1992.

PATTIPATI, K.R. ALEXANDRIDIS, M.G. “**Application of Heuristic Search and Information Theory to Sequential Fault Diagnosis.**” IEEE Transactions on Systems, Man, and Cybernetics 20, 872-887, 1990.

PAU, L. F. **Failure Diagnosis and Performance Monitoring,** Marcel Dekker, Inc., 1981.

POPPER, K. R. **A lógica da pesquisa científica.** São Paulo, Cultrix, 1974.

PREPARATA, F. P., METZE, G., CHEN, R. T. **On the connection assignment problem of diagnosable systems,** IEEE Transactions on Computers, 1967.

PREIST, C., ESHGHI, K. BERTOLINO, B. **Consistency-based and abductive diagnosis as generalized stable models,** Ann. Math. Artificial Intelligence 11 (1994) 51-74.

PULIDO, B. C., GONZALEZ, A. **Possible conflicts: a compilation technique for consistency-based diagnosis,** IEEE Transactions on Systems, Man, and Cybernetics, 2004.

RAGHAVAN, V. SHAKERI, M. PATTIPATI, NATICK. **Optimal and near-optimal test sequencing algorithms with realistic test models.** IEEE Transactions on Systems, Man and Cybernetics, 1999.

REITER, R. **A theory of diagnosis from first principles,** Artificial Intelligence, 1987.

RUSSEL S., NORVIG P., **Artificial intelligence: A modern approach,** Prentice Hall International, Inc., 2003.

RYMON, R. **A Final Determination of the Complexity of Current Formulations of Model-Based Diagnosis (Or Maybe Not Final?),** Proceedings of Model-Based Diagnosis Workshop, AAAI-91, Anaheim, CA, 1991.

SACHENBACHER, M., STRUSS, P. WEBER, R. **Advances in design and implementation of obd functions for diesel injection systems based on a qualitative approach to diagnosis,** Proceedings of the SAE 2000 World Congress, SAE, 2000.

SCHANK, R. C., ABELSON, R. P. **Scripts, plans, goals, and understanding.** Hillsdale, N. J.: Erlbaum, 1977.

SIMANI, S., FANTUZZI, C., Patton, R. J. **Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques,** Springer-Verlag, January 17, 2003, ISBN: 1-85233-685-4.

SIMOUDIS, E. **Using case-based reasoning for customer technical support**, IEEE Expert, 1992.

SMITH, E. E., ADAMS, N., SCHORR, D. **Fact retrieval and the paradox of interference**. Cognitive Psychology 10, 1978.

SOCHOROV, M. **General troubleshooting is NP-hard**. Proceedings of the Fifth Workshop on Uncertainty Processing, WUPES'2000 (2000)

STEELS, L. **The deepening of Expert Systems**, AICOM, No 1, 1988.

STOREY, N. **Safety-Critical Computer Systems**, Addison-Wesley, 1996.

STRUSS, P., SACHENBACHER, M., CARLEN, C. M. "**Insights from building a prototype for model-based on-board diagnosis of automotive systems**". In: Proceedings of the Eleventh International Workshop on Principles of Diagnosis (DX-00), pp. 201-212, Morelia, Mexico, (2000).

STRUSS, P., HELLER, U. **Intelligent Diagnosis in Industrial Applications**. Shaker Verlag, Aachen, Germany, 2001 ISBN3-8265-7002-2, p. 85-94.

THAGARD, P. **How scientists explain disease**. Princeton, NJ: Princeton University Press, 1999.

TU, F., PATTIPATI, K. R. **Rollout Strategies for Sequential Fault Diagnosis**. IEEE System, Man and Cybernetic – Part A, 2002.

VIJAY, V., GOVINDARAJ, T. **Knowledge Organization in Intelligent Tutoring Systems for Diagnostic Problem Solving in Complex Dynamic Domains**. International Conference on Systems, Man and Cybernetics, 1990. Conference Proceedings, IEEE

VENKATASUBRAMANIAN, V., RENGASWAMY, R. YIN, K., KAVURI, S. N. **A review of process fault detection and diagnosis, Part I: Quantitative model-based methods**. Computers in Chemical Engineering 27:293-311, 2003a.

VENKATASUBRAMANIAN, V., RENGASWAMY, R. YIN, K., KAVURI, S. N. **A review of process fault detection and diagnosis, Part II: Qualitative models and search strategies**. Computers in Chemical Engineering 27:313-326, 2003.

WATSON, I. MARIR, F. **Case-Based Reasoning: A Review**. The Knowledge Engineering Review, vol 9(4), pp327-354, 1994.

WEISS, S.M., KULIKOWSKI, C.A., AMAREL, S., SAFIR, A. **A model-based method for computer-aided medical decision making**, Artificial Intelligence 11, 1978.

WILLIAMS, B. C. NAYAK, P. P. **A model based approach to reactive self-configuring systems**, 14th National Conf. on Artificial Intelligence, 1996.

