



Universidade Federal de Minas Gerais

Programa de Pós-Graduação em Engenharia Elétrica

Raciocínio Baseado em Casos: Uma Abordagem Utilizando o Sistema Imune Artificial

Dissertação de mestrado submetida à banca examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

Autor: Eduardo Enrique Ostos Carvalho

Orientador: Professor Walmir Matos Caminhas

Belo Horizonte, MG

30 de Setembro de 2009

Dedico com carinho este trabalho à
Luciana e nossos filhos Rafael e Pedro.

Agradecimentos

Agradeço ao professor Walmir por ter acreditado na minha capacidade e pela liberdade necessária na criação deste trabalho.

Agradeço a Carlos Paiva, coordenador do Programa de Pós-Graduação em Engenharia Elétrica da PUC-MG, por ser uma grande energia incentivadora à realização deste trabalho, mesmo muito antes do seu início.

Agradeço ao CNPq pelo suporte financeiro no momento oportuno.

Agradeço à minha esposa Luciana pelo seu amor, carinho, lucidez e paciência em todos estes anos juntos.

Agradeço à minha mãe por ser exemplo de amor, fé e empenho infinitos para que tivéssemos a melhor educação e nos tornássemos amigos dos livros e do conhecimento.

Agradeço ao meu pai por ter plantado de forma lúdica a semente do interesse pela ciência e pelas diferenças.

Agradeço ao meu irmão e irmã pela cumplicidade na dedicação aos estudos como molde para o engenho e a arte tão necessários nos tempos atuais.

Agradeço à minha sogra Ana Maria Guerra Moreira pela paciência e apoio imprescindível nos momentos difíceis da rotina e nas revisões de português. Sua contribuição foi crucial.

Agradeço aos meus tios, tias e primos pelo saudável convívio familiar e em especial minha avó Gerolisa, pessoa de assombrosa perseverança e força. Desistir é a verdadeira derrota.

Agradeço às profissionais da secretaria do CPDEE pelo apoio e boa vontade, que vão muito além da sua obrigação.

Agradeço ao ambiente de camaradagem e colaboração do CPDEE que permitiu o encontro com diversos alunos e professores, de idéias e origens diferentes, quesito essencial para criatividade e inovação. Em especial, gostaria de agradecer aos colegas Celso Araújo, Thiago Guzella, Thiago Arreguy e Levi Lelis pela paciente escuta e apoio na resolução de problemas das mais diferentes formas.

A Deus por ter me dado o dom da vida e pelos contornos em que ela se deu até agora. Prometo extrair o máximo dela.

Resumo

O Raciocínio Baseado em Casos (RBC) é um poderoso método de resolução de problemas que se baseia no sistema cognitivo humano, alicerçado na utilização de soluções encontradas em problemas egressos semelhantes resolvidos. A solução em questão se dá pela busca na memória por um problema similar, que pode ou não ter a sua solução adaptada à demanda presente. Após a busca pelo problema mais parecido, este é analisado para detectar a necessidade de adaptação de algum componente, com o objetivo de evitar retrabalho e reaproveitar conhecimento previamente validado. Em seguida, e se necessário, a solução é avaliada para certificar a sua efetividade. Ao final, o caso, ou problema, resolvido é armazenado junto aos demais, contribuindo para aumentar o conhecimento do sistema. Estas quatro etapas, ou processos, operados em conjunto, constituem o ciclo de vida do RBC.

O RBC é, antes de tudo, uma metodologia, e como tal, não especifica a operação de cada um dos seus processos, o que abre a possibilidade para a utilização de diferentes paradigmas e ferramentas computacionais. Sob esta ótica, o Sistema Imune Artificial (SIA) é um paradigma computacional interessante que pode ser utilizado em conjunto com o RBC. A razão é que ele se apóia em sofisticados atributos do sistema biológico humano, como a detecção de padrões, compressão de dados, geração de população e adaptação dinâmica ao ambiente, características relevantes para o RBC.

Este trabalho apresenta um modelo híbrido de RBC e SIA, responsável por gerar contribuições nos processos de recuperação (busca e avaliação), adaptação (reutilização e revisão) e retenção (armazenamento) de casos. Dentre as contribuições mais relevantes estão a criação de uma maneira alternativa de agrupar casos, identificação de áreas de densidade de casos, melhoria na eficiência da busca no espaço de casos e armazenamento das relações entre casos correlatos.

O modelo proposto é aplicado a um problema de detecção e diagnóstico de falhas de uma máquina de corrente contínua, operando com simulações em modo supervisionado e não supervisionado. O método híbrido de RBC e SIA é confrontado com outros mais tradicionais de busca e armazenamento de casos. Ao final, os seus desdobramentos são comparados utilizando métricas específicas de desempenho do RBC, com resultados indicando um horizonte promissor para o modelo proposto neste trabalho.

Palavras-chave – Raciocínio Baseado em Casos, Sistema Imune Artificial, Anticorpo, Antígeno, Detecção de Falhas, Motor Elétrico.

Abstract

Case-Based Reasoning (CBR) is a powerful method for solving problems and is fundamented on the human cognitive system, which is strongly based on solving problems based on known similar ones. The solution is given by searching a memory for a similar situation, which in turn can or cannot be adapted to the current state. After searching for the more similar situation, it's analyzed to detect the adaptation needs, in order to avoid unnecessary effort and at the same time reuse previously validated knowledge. Then, if necessary, the solution is evaluated to ensure its effectiveness. Finally, the case, or problem solved, is stored together with others, helping to increase the system knowledge. These four steps, or processes, operated together, constitute the CBR life cycle.

CBR is first and foremost a methodology, and as such, does not specify operational details of each of its processes, what opens the possibility for the use of different paradigms and computational tools. On this point of view, the Artificial Immune System (AIS) is an interesting computational paradigm that can be used in conjunction with CBR. The reason is that it relies on sophisticated attributes of the human biological system, such as pattern recognition, data compression, population generation and dynamic environment adaptation, all those relevant characteristics to CBR.

This work presents a hybrid model of CBR and AIS, responsible for generating contributions in the process of recovery (search and evaluation), adaptation (reuse and revision) and retention (storage) of cases. Among the most relevant contributions are the creation of an alternative way of grouping cases, identifying high density areas, improve search efficiency in the case space and store the relationships of similar cases.

The proposed model is applied to a fault detection and diagnosis problem of a DC machine, which operates with simulations on supervised and unsupervised mode. The hybrid method of CBR and AIS is confronted with other more traditional CBR search and storage techniques. Finally, the results are compared using specific CBR performance metrics, what at the end showed positive perspectives for the proposed model.

Keywords – Case-Based Reasoning, Artificial Immune System, Antibody, Antigen, Fault Detection, Electrical Engine

Índice

Capítulo 1 - Introdução	1
1.1. Motivação.....	1
1.2. Objetivos	3
1.3. Organização do trabalho.....	4
Capítulo 2 - Raciocínio Baseado em Casos.....	5
2.1. Introdução	5
2.2. Histórico	6
2.2.1. Abordagens Precursoras	6
2.2.2. Teoria da Memória Dinâmica	7
2.3. Raciocínio Baseado em Casos	10
2.3.1. Estrutura do Conhecimento	10
2.3.2. Raciocínio Baseado em Casos, seus Tipos e Formação.....	11
2.3.3. Ciclo de Vida do Raciocínio Baseado em Casos	17
2.3.4. Raciocínio Baseado em Casos e outros métodos de raciocínio	29
2.4. Exemplos de Aplicações	31
2.4.1. CYRUS	31
2.4.2. Clavier.....	31
2.4.3. SMART	33
2.4.4. FormTool.....	33
2.4.5. Outros exemplos	34
2.5. Raciocínio Baseado em Casos e <i>Soft Computing</i>	35

2.5.1. Lógica Fuzzy	35
2.5.2. Redes Neurais Artificiais	36
2.5.3. Computação Evolutiva	36
Capítulo 3 - Sistema Imune Artificial	39
3.1. Introdução	39
3.2. Sistema Imune Biológico.....	40
3.2.1. Breve Histórico.....	40
3.2.2. Componentes.....	41
3.2.3. Sistema Imune Adaptativo	44
3.3. Sistema Imune Artificial.....	52
3.3.1. O Espaço de Formas	53
3.3.2. Representações e Medidas de Afinidade.....	54
3.3.3. Processos Imunes e Seus Algoritmos.....	56
3.4. Exemplos de Domínios de Aplicação.....	62
3.4.1. Aprendizado	62
3.4.2. Detecção de Anomalias	65
3.5. Inteligência Artificial, Computacional e o Sistema Imunológico Artificial	67
Capítulo 4 - Raciocínio Baseado em Casos e o Sistema Imune Artificial	71
4.1. Introdução	71
4.2. Trabalhos Anteriores	72
4.2.1. Memória de Casos e o Sistema Imunológico Artificial	72
4.2.2. Busca Direcionada de Casos	79
4.3. Proposta de utilização do Sistema Imunológico Artificial no Raciocínio Baseado em Casos.....	81
4.3.1. Agrupamento.....	81
4.3.2. SIA e o Ciclo do Raciocínio Baseado em Casos	84
Capítulo 5 - Experimentação e Análise de Resultados	107
5.1. Introdução	107

5.2.	Motivação Pelo Domínio de Falhas em Máquinas Elétricas	108
5.3.	Detecção e Diagnóstico de Falhas Em Máquina de Corrente Contínua	108
5.4.	Geração dos Dados do Motor Elétrico de Corrente Contínua.....	109
5.5.	Modelagem do Experimento	112
5.5.1.	Representação do Caso	113
5.5.2.	Algoritmo de Agrupamento	114
5.5.3.	Detecção e Diagnóstico de Falhas Através de um Protótipo de Sistema Híbrido de RBC e SIA	114
5.6.	Coleta e Análise de Resultados	119
5.6.1.	Primeira Modalidade	119
5.6.2.	Segunda Modalidade	145
5.7.	Principais dificuldades, limitações e avanços	147
	Capítulo 6 - Conclusão e Perspectivas Futuras.....	151
	Referências Bibliográficas	155
	Apêndice A - Máquina de corrente contínua.....	165
A.1.	Modelo do Sistema de Acionamento de Corrente Contínua	165
A.1.1.	Modelo de Máquina de Corrente Contínua	166
A.1.2.	Modelo de Carga Dinâmica.....	168
A.1.3.	Modelo de Conversores Estáticos.....	168
A.2.	Modelagem das Falhas para o Sistema de Acionamento de Corrente Contínua	169
A.2.1.	Falhas em Atuadores	169
A.2.2.	Falhas em Componentes da Planta	170
A.2.3.	Falhas de Instrumentação.....	171
A.3.	Modelo Geral das Falhas no Sistema de Acionamento de Corrente Contínua	172
	Apêndice B – Agrupamento Participativo	175
B.1.	Aprendizado Participativo	175

B.2. Agrupamento Participativo	178
--------------------------------------	-----

Lista de Figuras

Figura 2.1 - Ciclo de vida do RBC (Aamodt e Plaza 1994).....	18
Figura 2.2 – Processos de RBC e suas tarefas (Aamodt e Plaza 1994).	18
Figura 2.3 – Casos de referência e grupos de competência em uma base de casos (Smyth e McKenna 1999).....	25
Figura 3.1 – Níveis de defesa do sistema imune (de Castro 2001).	42
Figura 3.2– (a) A célula B possui todos os anticorpos iguais. (b) O antígeno possui diversidade de epítomos, reconhecidos por vários anticorpos (de Castro 2001).....	45
Figura 3.3 – Seleção clonal mostrando os possíveis desdobramentos da apresentação do antígeno ao anticorpo. Ao final ocorre a proliferação e diferenciação em células de memória e plasmócitos, que secretam anticorpos em grandes quantidades (de Castro 2001).	47
Figura 3.4 – Respostas entre anticorpo-antígeno e anticorpo-anticorpo da rede imunológica (de Castro 2001).....	50
Figura 3.5 – (a) Reconhecimento por complementaridade. (b) Espaço de formas S com região V onde um paratopo (ponto) pode reconhecer um epítopo (cruz) (de Castro 2001).....	54
Figura 3.6 – Variações de medidas no espaço de formas de Hamming. (a) Quantidade total de bits complementares. (b) Quantidade de <i>r-bits</i> complementares consecutivos. (c) Medida de Hunt, Equação (2.8) (de Castro 2001).....	56
Figura 3.7 – (a) Mutação de cadeia binária em ponto único. (b) Mutação de cadeia binária pontos múltiplos (de Castro 2001).....	59
Figura 4.1 – Inserção de novo caso (Hunt, Cooke e Holstein 1995).	74
Figura 4.2 – (a) Antígenos distribuídos no espaço. (b) Os anticorpos, círculos, se posicionam em distribuição muito semelhante aos antígenos. O asterisco de cor vermelha representa o centro do agrupamento...	86
Figura 4.3 – Diagrama de blocos da aiNet para uso em conjunto com o RBC.	87

Figura 4.4 - Espaço em três dimensões onde os anticorpos são representados pelos círculos e os antígenos pelos pontos em forma de estrela. O ponto sem nenhuma conexão é o centro da distribuição dos anticorpos. As linhas tracejadas são para dar a impressão de profundidade no espaço de três dimensões.	100
Figura 4.5 – Três casos e suas áreas de cobertura. Os casos 2 e 3 são casos auxiliares.	104
Figura 5.1 – Falha do sensor da corrente de armadura.....	111
Figura 5.2 – Falha do sensor da corrente de campo.	111
Figura 5.3 – Falha do sensor de velocidade.....	112
Figura 5.4 - Representação espacial dos casos de falhas da tabela 5.1.....	112
Figura 5.5 – Dados da execução 1 da 1ª. modalidade, com 3750 casos.	116
Figura 5.6 – Dados da execução 2 da 1ª. modalidade, com 1875 casos.	117
Figura 5.7 – Dados da execução 3 da 1ª. modalidade, com 932 casos.	117
Figura 5.8 – Resultado das detecções e diagnósticos dos casos na primeira modalidade.	120
Figura 5.9 – Casos de uma operação normal inicialmente fornecidos ao protótipo.....	121
Figura 5.10 – a) Os índices distribuídos no espaço e seu centro após a execução da aiNet considerando os casos fornecidos na Figura 5.9. b) Ampliação da área povoada com casos e índices da Figura 5.10a.....	121
Figura 5.11 – Casos agrupados com seus índices e centros durante o treinamento.....	122
Figura 5.12 – Casos agrupados durante a validação.	122
Figura 5.13 – Resultado final da detecção e diagnóstico de casos com formação da rede de casos.	123
Figura 5.14 – Busca baseada em índices utilizando os cinco anticorpos mais próximos, ou vizinhos, do caso candidato.	126
Figura 5.15 - Eficiência com a busca por índice com um vizinho.	127
Figura 5.16 – Competência com a busca por índice com um vizinho.....	127
Figura 5.17 - Qualidade com a busca por índice com um vizinho.	128
Figura 5.18 - Excelência com a busca por índice com um vizinho.	128

Figura 5.19 – Eficiência com a busca por índice com dois vizinhos.	129
Figura 5.20 - Competência com a busca por índice com dois vizinhos.	129
Figura 5.21 – Qualidade com a busca por índice com dois vizinhos.	130
Figura 5.22 – Excelência com a busca por índice com dois vizinhos.	130
Figura 5.23 - Eficiência com a busca por índice com três vizinhos.	131
Figura 5.24 - Competência com a busca por índice com três vizinhos.....	131
Figura 5.25 - Qualidade com a busca por índice com três vizinhos.	132
Figura 5.26 - Excelência com a busca por índice com três vizinhos.	132
Figura 5.27 – Eficiência com a busca por índice com quatro vizinhos.	133
Figura 5.28 – Competência com a busca por índice com quatro vizinhos.....	133
Figura 5.29 – Qualidade com a busca por índice com quatro vizinhos.	134
Figura 5.30 – Excelência com a busca por índice com quatro vizinhos.	134
Figura 5.31 – Eficiência com a busca por índice com cinco vizinhos.	135
Figura 5.32 – Competência com a busca por índice com cinco vizinhos.....	135
Figura 5.33 – Qualidade com a busca por índice com cinco vizinhos.	136
Figura 5.34 - Excelência com a busca por índice com cinco vizinhos.	136
Figura 5.35 – Através da média das distâncias entre os índices, foram criadas as conexões 1-2, 1-3, 2-3 e 4-5, mas os índices 1, 2, 3 não possuem conexão com 4, 5, pois as distâncias destes grupos de índices são maiores que a média das distâncias dos índices do grupo. As linhas tracejadas tem por objetivo passar a noção de perspectiva em um espaço tridimensional.....	139
Figura 5.36 - Identificação de índices para realização da manutenção – execução 1.	141
Figura 5.37 – Seleção de casos para eliminação da base de casos – execução 1.....	141
Figura 5.38 - Identificação de índices para realização da manutenção – execução 2.	142
Figura 5.39 - Seleção de casos para eliminação da base de casos – execução 2.....	142
Figura 5.40 - Identificação de índices para realização da manutenção – execução 3.	143

Figura 5.41 - Seleção de casos para eliminação da base de casos – execução 3.....	143
Figura 5.42 – valores e taxas de redução da base de casos após remoção dos casos auxiliares.	144
Figura 5.43 – Diagnóstico da falha de corrente de campo pelo supervisor. ...	146
Figura 5.44 - Diagnóstico da falha de corrente de armadura pelo supervisor.	146
Figura 5.45 - Diagnóstico da falha de velocidade pelo supervisor.....	147
Figura A.1 - Representação do sistema de acionamento de CC.....	165
Figura A.2 - Diagrama de blocos do sistema de acionamento CC.	166
Figura A.3 - Representação do sistema de acionamento C.C. para falhas nos atuadores.....	169
Figura A.4 - Representação do sistema de acionamento C.C. para os diversos tipos de falhas.	172
Figura B.1 – Mecanismo do aprendizado participativo	177
Figura B.2 – O ponto xk fará parte do grupo i , pois apresenta maior compatibilidade com i do que com j	180
Figura B.3 – O ponto xk origina a criação do grupo de centro vnk ($n = \text{novo}$), pois $ask + 1 e ajk + 1 \geq \tau$	180
Figura B.4 – O ponto vrk ($r = \text{resultante}$) é o centro do novo grupo originado da união dos grupos cujos centros são vik e vjk , já que $\rho_{ijk} = 1 - d_{vik}, vjk \geq \lambda$	181

Lista de Tabelas

Tabela 2.1 - Principais medidas de distância	22
Tabela 2.2 – Medidas baseadas em distância (não utilizam exclusivamente a distância).	23
Tabela 3.1 - * A semelhança se dá com qualquer um dos seguintes paradigmas: redes neurais, lógica fuzzy e computação evolucionária (de Castro e Timmis 2002).	69
Tabela 4.1 - Lista dos atributos e cálculo da semelhança entre os casos.....	73
Tabela 4.2 – Analogia entre elementos do sistema imune e do Raciocínio Baseado em Casos.	77
Tabela 5.1 – Distribuição dos pontos por evento.....	110
Tabela 5.2 – Parâmetros do motor.	111
Tabela 5.3 – Distribuição dos casos na 1ª. execução da 1ª. modalidade.....	115
Tabela 5.4 – distribuição dos casos na 2ª. execução da 1ª. modalidade.	115
Tabela 5.5 – distribuição dos casos na 3ª. execução da 1ª. modalidade.	116
Tabela 5.6 – Quantidade de casos, índices e compressão dada a presença dos casos de início, treinamento e validação nas execuções das Tabelas 5.4 a 5.6.	137
Tabela A.1 - Resumo das falhas no sistema de acionamento CC.....	173

Lista de Siglas

CBR ou RBC	: Raciocínio Baseado em Casos
AIS ou SIA	: Sistema Imune Artificial
ANN ou RNA	: Redes Neurais Artificial
MOP	: Pacote Organizador de Memória
TOP	: Pacote Organizador Temático
CBM	: Manutenção da Base de Casos
CE	: Computação Evolutiva
BCR	: Receptores de Células B
TCR	: Receptores de Células T
APC	: Célula Apresentadora de Antígeno
MHC	: Complexo de Histocompatibilidade Principal
AI ou IA	: Inteligência Artificial
CI ou IC	: Inteligência Computacional
CLONALG	: Algoritmo de Seleção Clonal
aiNet	: Algoritmo gerador da Rede Imune Artificial
CC	: Corrente Contínua

Capítulo 1 - Introdução

1.1. Motivação

A criação de dispositivos ou máquinas inteligentes que possam realizar atividades semelhantes aos humanos ou até mesmo superá-las é tema de intensa pesquisa da comunidade científica, pois gera desdobramentos importantes em áreas tão diversas quanto a economia, saúde, indústria e lazer. Dentre alguns exemplos de aplicação, citam-se as análises preditivas no mercado de ações, o apoio na análise e diagnóstico de pacientes, a prevenção de falhas em processos fabris e o comportamento adaptativo de personagens em jogos digitais.

Existem diferentes maneiras de se criar dispositivos e programas com características inteligentes, apesar de ainda estar muito distante a simulação ou criação de uma máquina que incorpore completamente a inteligência humana. Muitas destas maneiras, ou paradigmas computacionais, se inspiraram em sistemas ou fenômenos possuidores de atributos inteligentes (percepção, aprendizado, generalização e adaptação, dentre outros) como o sistema nervoso humano, a biologia evolutiva, o sistema imune humano e o comportamento social de animais e insetos, para colocar somente aqueles inspirados na natureza. Dentre estes paradigmas computacionais, podemos nomear as Redes Neurais Artificiais (RNAs), computação evolutiva, Sistema Imune Artificial (SIA) e inteligência de enxame.

Diferente das fontes de inspiração citadas acima, o modelo cognitivo humano é uma poderosa ferramenta que possibilitou aos humanos compreender o seu entorno e interagir com ele de modo a construir complexas sociedades organizadas, criar ferramentas sofisticadas, melhorar drasticamente a sua qualidade e expectativa de vida. Todas estas realizações indicam um grande potencial para que os princípios do modelo cognitivo humano possam também ser utilizados no desenvolvimento de programas ou dispositivos com características inteligentes para a resolução de problemas.

O resultado de estudos nessa área objetivando a concepção de máquinas inteligentes criou o Raciocínio Baseado em Casos (*Case-Based Reasoning* – CBR ou RBC) (Kolodner 1993), alicerçado em características como o aprendizado, memória, raciocínio e experiência humana. Diferentemente de outros paradigmas computacionais que fazem uso de um conhecimento genérico ou associações mais amplas entre o problema e a solução, o RBC é capaz de utilizar a experiência específica contextualizada, ou casos,

armazenada na sua memória, também conhecida como base de casos. Sob esta ótica, um novo problema é resolvido, encontrando-se o caso mais similar na memória e reutilizando a solução presente nele no problema proposto, armazenando em seguida o problema solucionado para que seja útil em situações futuras. A apresentação sucessiva de problemas faz com que o RBC possua uma aprendizagem incremental e sustentável, já que os problemas solucionados estão imediatamente disponíveis para uso (Aamodt e Plaza 1994).

O RBC propõe uma maneira diferente de abordar problemas, solucionando-os através do reaproveitamento de soluções de problemas semelhantes previamente conhecidos. Para cumprir este objetivo de maneira estruturada, Aamodt e Plaza (1994) propuseram um ciclo de vida com quatro processos: recuperação, reutilização, revisão e retenção, conhecidos também como 4Rs. Cada processo tem uma função específica que somados necessitam possuir as características do modelo cognitivo humano nas quais o RBC foi concebido.

O RBC propõe métodos e estratégias que direcionam a construção de um sistema inteligente baseado em seus princípios, mas deve-se entender que são direções, caminhos e não técnicas ou algoritmos (Pal e Shiu 2004). Isto significa que o RBC não dispõe de ferramental específico para lidar com a complexidade dos conceitos e questões a que é submetido, o que faz com que ele conte com o conhecimento de diferentes áreas ou paradigmas computacionais que possuam técnicas e algoritmos próprios, passíveis de utilização em tarefas de representação, indexação, busca, avaliação e adaptação de casos.

Partindo desta realidade, o RBC primordialmente utilizou técnicas ou modelos atualmente considerados tradicionais, como a representação e indexação de casos baseada em modelo de banco de dados relacionais ou orientado ao objeto; avaliação da semelhança de casos através de técnicas do vizinho mais próximo e a busca de casos, de forma exaustiva, na base de casos, dentre outros (Pal e Shiu 2004). Sistemas RBC foram construídos segundo estas técnicas e obtiveram resultados satisfatórios, mas à medida que foi necessário interagir com problemas ou situações de imprecisão, incerteza e que demandassem raciocínio aproximado, outras opções precisaram ser buscadas (Cheetham, Shiu e Weber 2005). Nestas opções, estão incluídas a lógica fuzzy, RNAs, SIA, computação evolutiva e aprendizado estatístico, para citar algumas, que podem ser usadas para apoiar os processos de RBC na resolução de problemas.

O uso de paradigmas computacionais de inteligência não é uma novidade no RBC, apesar de pouco explorado, principalmente devido ao caráter prático destas técnicas para lidar com restrições, geralmente presentes no mundo real na implantação de sistemas RBC e, em menor escala, em projetos de pesquisa. Este trabalho abordará o RBC através de um paradigma computacional específico, o Sistema Imune Artificial. O SIA é um campo relativamente novo, sua origem data de um artigo publicado por Farmer *et al.* (1986) e sua primeira conferência internacional (*International Conference on Artificial Immune Systems – ICARIS*), ocorreu em 2002. O SIA refere-se a

sistemas adaptativos inspirados na imunologia teórica e experimental, objetivando a solução de problemas (de Castro e Timmis 2002).

SIA apresenta características interessantes que podem ser utilizadas através de seus princípios e algoritmos no RBC, sendo as mais expressivas representadas pela detecção de padrões, geração de memória, capacidade de aprendizado, diversidade de indivíduos, descentralização e resiliência.

A motivação da escolha pelo SIA no RBC se deu por ser pouquíssimo explorado na literatura de RBC, possuindo até onde este trabalho conseguir levantar, somente dois artigos publicados (Hunt, Cooke e Holstein 1995) (Hunt e Fellows 1996), ambos de meados da década de 90. Também a escolha se pautou pela percepção do potencial de contribuição do SIA no RBC. Em mais de dez anos após a publicação destes artigos, houve evolutiva e significativa produção acadêmica do SIA, o que catalisa o surgimento de novas técnicas e algoritmos que podem ser aplicados no RBC.

A outra motivação se deu como consequência de palestra proferida na Conferência Internacional de RBC em 2007 (*International Conference of Case Based Reasoning - ICCBR*) por um dos principais pesquisadores envolvidos com o RBC. Ele pontuou que o RBC corre o risco de tornar-se irrelevante, se mais trabalhos teóricos não forem publicados (Holloway 2007). Sendo assim, este trabalho vai de encontro a essa necessidade da comunidade de RBC, na medida em que propõe o uso de um paradigma computacional recente da computação natural como uma alternativa a ser utilizada no ciclo de vida do RBC.

1.2. Objetivos

O objetivo deste trabalho é apresentar uma proposta híbrida do Raciocínio Baseado em Casos com o Sistema Imune Artificial, colocando-se como uma alternativa a ser utilizada nos processos de recuperação, reutilização, revisão e retenção de casos em iniciativas de pesquisa acadêmica e implantações no mundo real.

A proposta consta mais especificamente da utilização de princípios e algoritmos do SIA, como a seleção clonal e rede imunológica, de modo a:

- Efetuar a busca de casos em regiões específicas do espaço sem necessidade de percorrer toda a base de casos;
- Adaptar ou solucionar casos através de operadores genéticos que não encontrem soluções diretas na base de casos;
- Explicitar regiões e permitir a exploração do espaço de forma racionalizada através de uma rede de casos;
- Facilitar a identificação e remoção de casos considerados desnecessários na base de casos, contribuindo para melhoria na eficiência de um sistema de RBC.

Para demonstrar algumas das contribuições, foram realizados experimentos através de um protótipo que utiliza este modelo híbrido na detecção e diagnóstico de falhas no domínio de um motor elétrico de corrente contínua. Os experimentos variaram na quantidade de casos tratados e nas técnicas utilizadas nos processos do RBC.

É importante ressaltar que este trabalho não tem a intenção de realizar comparações com outros paradigmas computacionais e seus algoritmos, mas estes são apresentados e relacionados com o RBC para que os interessados possam ter uma melhor percepção da aplicação de seus princípios.

1.3. Organização do trabalho

O Capítulo 2 discute o Raciocínio Baseado em Casos, seu histórico, elementos formadores, ciclo de vida, exemplos e relações com outros paradigmas computacionais relevantes de *Soft Computing*.

O Capítulo 3 descreve o Sistema Imune Artificial, apresentando o sistema imune biológico humano como fonte de inspiração para o Sistema Imune Artificial, os princípios biológicos e principais algoritmos do Sistema Imune Artificial, exemplos de aplicação e sua relação com a inteligência computacional.

O Capítulo 4 apresenta os trabalhos precursores (Hunt, Cooke e Holstein 1995) (Hunt e Fellows 1996) que relacionaram RBC com o SIA. Na sequência, são realizadas as considerações sobre estes trabalhos e é proposto um modelo híbrido do Raciocínio Baseado em Casos e do Sistema Imune Artificial que utiliza princípios imunológicos no ciclo de vida do Raciocínio Baseado em Casos.

O Capítulo 5 trata dos experimentos e da análise de resultados. Os experimentos são realizados no domínio do diagnóstico de falhas de um motor de corrente contínua, com objetivo de avaliar o modelo híbrido, segundo medidas de desempenho do Raciocínio Baseado em Casos. A intenção dos experimentos não é a de resolver de maneira ótima o problema de detecção de falhas no campo de máquinas elétricas ou apresentar o Sistema Imune Artificial como um paradigma computacional superior aos outros existentes, mas, sim, a de constatar seu potencial para realçar a sua posição de opção viável a ser aplicada no RBC.

O Capítulo 6 conclui o trabalho, resumindo suas principais contribuições e propondo os tópicos a serem desenvolvidos futuramente.

Capítulo 2 - Raciocínio Baseado em Casos

2.1. Introdução

O cotidiano nos apresenta as mais diferentes situações, algumas familiares e outras não. À medida que um ser humano passa por diversas experiências, ele tende a aumentar a quantidade de decisões acertadas, pois possui mais conhecimento acumulado. Estas decisões acertadas, ou a falta delas, são frutos, em grande parte, da quantidade, tipo e relevância das experiências ou situações, a que uma pessoa foi submetida ao longo de sua vida.

Essa abordagem de aprendizado baseada em experiências vividas ou observadas é rotineiramente praticada por todas as pessoas em maior ou menor escala e encontra seus mecanismos de funcionamento explicados pela ciência cognitiva, linha da psicologia que estuda, dentre outras áreas, a memória, raciocínio, e resolução de problemas, todas estas de grande interesse na construção de dispositivos inteligentes.

O Raciocínio Baseado em Casos (*Case-Based Reasoning* – CBR ou RBC) surge justamente como uma abordagem para resolução de problemas baseada em experiências, tendo suas raízes na ciência cognitiva, representação do conhecimento, matemática e aprendizado de máquina (Richter e Aamodt 2005).

A partir de pesquisas realizadas sobre o processo da solução de problemas e aprendizado humanos, no final da década de 70 e com a divulgação do RBC em workshops, no final da década de 80 (Kolodner 1993) (Bergmann 2000), a comunidade científica tomou maior conhecimento do RBC. Esta disseminação gerou aumento da produção acadêmica e criação de vários protótipos e sistemas inteligentes, muitos deles com grande sucesso na resolução de problemas reais onde foram implantados.

Discute-se aqui a origem do RBC, seus processos formadores, sua relação com outras áreas de conhecimento, outros métodos de raciocínio e alguns exemplos que explicitem a sua maneira única de criar um sistema inteligente a partir de experiências contextualizadas. Esta característica habilita o RBC a ser usado em domínios nos quais os fenômenos e suas relações causais não são bem compreendidos, dificultando a representação do conhecimento através de uma linguagem formal.

A abordagem do RBC aqui apresentada leva em conta principalmente o trabalho pioneiro realizado por Janet Kolodner (1993).

2.2. Histórico

Todas as pessoas se utilizam de algum tipo de processo cognitivo que lhes permita interpretar o mundo à sua volta e tomar decisões. O conhecimento específico adquirido através de exemplos pode ser ilustrado em diversos campos, como no uso de jurisprudências no direito, no diagnóstico de um profissional da saúde e no atendimento de usuários de computadores através de um *help-desk*. Consideremos tal afirmativa em mais detalhes para o caso de um médico que atende um paciente. O médico pode se deparar com uma combinação não usual de sintomas, mas se ele já tratou anteriormente um paciente com sintomas semelhantes, é bem provável que se lembre do caso e proponha o antigo diagnóstico como possível solução para o novo problema. Se o diagnóstico do caso similar mais antigo levou muito tempo, significa que ele poderá resolver mais rapidamente a situação recém apresentada. O médico não pode assumir que o antigo diagnóstico se aplica integralmente ao novo paciente, mas ele pode adaptá-lo, de modo que gere uma solução plausível de modo mais fácil e rápido.

Geralmente, na segunda vez que nos deparamos com uma situação igual a outra já vivenciada ou semelhante em algum nível (concreto ou abstrato), a resolvemos de forma mais fácil que a primeira, pois lembramos e repetimos a solução anterior, evitando seus possíveis erros e copiando seus acertos, com eventuais adaptações.

A abordagem cognitiva na qual o RBC se fundamenta para apresentar soluções baseadas em experiências egressas semelhantes é a teoria da memória dinâmica (Schank 1982). Antes da apresentação do RBC, serão introduzidas algumas abordagens precursoras à teoria da memória dinâmica e suas restrições.

2.2.1. Abordagens Precursoras

Segundo Kolodner (1993), a psicologia moderna sustenta a tradição do uso da teoria do esquema (*schema*), o que em linhas gerais significa a representação do conhecimento geral armazenado em blocos na memória e estes blocos sendo usados no processo de raciocínio.

Schank e Abelson (1977) definiram tipos mais peculiares de representação, chamados de *scripts*, que são responsáveis por especificar e armazenar sequências de eventos. Os *scripts* são estruturas monolíticas contendo toda a sequência de eventos para um tipo de situação. Para que fique mais claro, consideremos o exemplo da visita a um médico e a um dentista (Kolodner 1993). As duas visitas têm diversas etapas em comum, como fazer o agendamento, deslocar-se ao consultório, entrar no consultório, esperar pelo atendimento, pagar e sair, mas também tem diferenças, como o tipo de serviço, podendo ser tratamento de canal, limpeza dentária ou exame

cardíaco. Os *scripts* representam as duas situações, visita ao dentista e ao médico, de maneira diferente, onde existiriam dois *scripts*, um que possui todos os eventos ligados na visita ao médico e outro na visita ao dentista.

Ao se deparar com uma nova ou inesperada situação, é natural que exista um maior grau de inferência para preencher as lacunas da falta de experiência. Em contrapartida, após a experiência inicial, o processo cognitivo cria expectativas e as analisa, pois ele já possui uma experiência semelhante que pode ser aproveitada, caso contrário, a nova situação seria tratada sem a utilização de conhecimento prévio, o que não condiz com a realidade humana.

Os esquemas e *scripts* têm restrições e lidam com a situação exposta acima de modo normativo, ou seja, eles não fornecem um tratamento especial para as instâncias, que são casos mais específicos das situações gerais e têm grande representatividade no processo cognitivo. Como utilizamos constantemente estruturas de conhecimento geral e específico (instâncias) no processo cognitivo, torna-se crucial a apresentação de um modelo que represente esta separação e enderece as deficiências e limitações dos *scripts*.

2.2.2. Teoria da Memória Dinâmica

Os princípios que regem o modelo cognitivo baseado na experiência estão contidos no modelo de memória dinâmica. Este modelo apresenta abordagens diferentes em relação às anteriores, principalmente no tocante à recuperação do conhecimento para reutilização futura e à classificação e representação de situações gerais e específicas.

A recuperação é um dos estágios mais críticos, pois executada da maneira correta possibilita o resgate dos exemplos, também conhecidos como casos, mais importantes para a situação corrente. Neste sentido, a escolha correta de como organizar o conhecimento deve passar por como rotular as estruturas de memória para que se tornem acessíveis, quando necessárias. Estas estruturas (gerais e específicas) devem ser rotuladas de acordo com o seu tipo e sua diversidade: a visita a um mecânico que consertou o carro, enquanto contava piadas e teve um bom resultado, poderia ser referenciada como “visita ao bom e engraçado mecânico”. Desta maneira, podemos recuperar a informação de modo mais fácil e rápido, quando estivermos em um contexto semelhante para resolver uma situação ou problema. A este rótulo damos o nome de índice.

A primeira premissa deste modelo (Schank 1982) é que a lembrança, compreensão, experiência e aprendizado não podem estar separados. A nossa memória muda devido às experiências a que somos submetidos, às questões que surgem dos resultados destas experiências e ao modo como as abordamos. Nós realizamos o entendimento na tentativa de integrar novas experiências àquelas que já conhecemos. Todas estas experiências permitem que o processo do conhecimento tenha um ponto de partida mais simples e próximo do objetivo desejado. É importante ressaltar que as experiências individuais e gerais (normativas ou comuns) nos fornecem informações usadas no processo de compreensão, o que naturalmente gera mudanças na nossa

memória, tornando-a dinâmica, pois estabelecemos novas relações e armazenamos conhecimento.

Dentre diversos pontos, alguns dos mais importantes na memória dinâmica são a integração entre as experiências gerais e específicas e a criação de índices relevantes. No primeiro ponto, a memória dinâmica cria a integração através da proposição de que a memória é organizada em Pacotes Organizadores de Memória (*Memory Organization Packets* - MOPs) e Pacotes Organizadores Temáticos (*Thematic Organization Packets* - TOPs), posteriormente detalhados.

O segundo ponto deste modelo é que as estruturas usadas para armazenamento são as mesmas usadas para processamento, fornecendo expectativas e sugerindo inferências.

Um indício desta premissa pode ser ilustrado pelo seguinte exemplo: uma pessoa encontra uma situação inusitada ao fazer compras em uma padaria, o atendente do caixa está muito nervoso e um homem grita para que ele entregue todo o dinheiro. Isto configura um assalto e o nosso protagonista tentará encontrar uma situação semelhante para tomar uma decisão favorável (sair correndo, gritar ou impedir o assalto). Caso ele não possua experiências parecidas, será processada gerando um maior grau de inferências, podendo ter sucesso ou não no seu objetivo de encontrar uma solução. Considerando que a mesma pessoa se depara com uma situação semelhante no futuro, a experiência prévia será resgatada e ela poderá chegar a conclusões mais rápidas e até mesmo seguras, pois sabe o que deu certo ou errado, parecido ou diferente em relação ao exemplo anterior. O armazenamento da primeira experiência explicita dois pontos ligados à segunda premissa: o armazenamento da experiência e o fornecimento de expectativas para compreensão de próximas experiências semelhantes. Desta maneira, a experiência foi usada para armazenamento e processamento. A consequência da segunda premissa é que se descobrimos novas maneiras de conectar as experiências contidas na memória, através de objetivos, planos ou intenções em comum, de maneira concreta ou abstrata, descobrimos novos tipos de processamento e vice-versa.

Com as premissas apresentadas, podemos iniciar o detalhamento dos MOPs e TOPs.

Os MOPs têm duas funções: armazenar o conhecimento geral e organizar em uma complexa hierarquia o conhecimento específico, ou casos, derivados de situações mais gerais. MOPs organizam informações. Neste sentido, os MOPs vão ao encontro do modelo de *scripts* apresentado, mas propondo soluções para suas restrições. Como citado anteriormente, os *scripts* são estruturas monolíticas e duplicam informações de situações semelhantes, mesmo que ambas tenham pequenas diferenças, como no exemplo da visita ao dentista e ao médico. Os MOPs descrevem as situações de uma maneira mais distribuída, cada MOP é composto por uma sequência de cenas (sequência de eventos) e cada cena é descrita na sua própria estrutura de memória, que retém informações normativas e específicas. Uma cena pode ser subdividida em *scripts* de um novo tipo e também indexar casos em mais

detalhes, criando assim uma estrutura em árvore orientada no sentido geral-específico. Esta estrutura permite que situações semelhantes, que seriam representadas pelo antigo conceito de *scripts*, sejam representadas no mesmo MOP, agora sem duplicidade e com a possibilidade de relacionar situações diferentes com algo em comum, o que é de suma importância no processo cognitivo. Retornando ao exemplo do dentista e do médico, a diferença entre as duas visitas utilizando MOPs seria tratada através de diferentes representações (variações da cena de atendimento) que fazem referência ao tipo de consulta e ao problema relatado pelo paciente. Os MOPs possibilitam que os casos, ou situações mais específicas, sejam devidamente indexados e esta indexação deve ser tal, que as características do caso usadas na indexação sejam uma combinação que explicita a sua diferença ou ineditismo.

Por sua vez, os TOPs são um tipo diferente de estrutura de memória. TOPs são muito semelhantes aos MOPs, mas com uma diferença. Segundo Kolodner (1993):

As situações que estas estruturas descrevem e organizam são tematicamente similares, ou seja, elas provêem meios de categorizar situações baseados nas intenções ligadas às situações e não nos detalhes das atividades contidos nelas. De acordo com Schank, os TOPs são responsáveis por diferentes comportamentos, como (Schank 1982):

- Ser lembrado de um caso que ilustra um ponto específico;
- Reconhecer o mesmo assunto ou caso com uma nova roupagem;
- Lembrar de ditados em situações oportunas;
- Tirar lições de uma situação que possa ser usada em outra;
- Perceber ocorrências aparentemente desconexas entre situações diferentes e tirar conclusões destas ocorrências;
- Prever a consequência de uma nova situação baseada no seu andamento atual;
- Saber qual o resultado de um novo acontecimento.

Para que o comportamento acima seja possível, é necessário que o processo de resgate da memória cruze domínios de conhecimento, ou seja, possa relacionar assuntos diferentes, mas com algo em comum. Como exemplo, cita-se uma situação em que um engenheiro civil que gerencia a reforma de um prédio e se depara com um problema de difícil solução. Para aumentar suas chances de sucesso, ele pede a planta elétrica e hidráulica do edifício e analisando-as, consegue vislumbrar a solução. O mesmo engenheiro possui um filho que está muito doente, com problemas respiratórios e a mãe não sabe o que pode ser, pois ela tentou os procedimentos que conhecia e estes não funcionaram. O engenheiro leva o filho ao pronto socorro e quando é atendido, se lembra de como resolveu o problema na obra e sugere ao médico uma radiografia do pulmão do seu filho.

O exemplo acima é nada mais que um caso específico de como os TOPs funcionam. De maneira mais ampla, os TOPs são formados pela combinação de descrições de objetivos e condições, que geralmente são derivados de visões diferentes dos planos seguidos para atingir o objetivo, ou seja, tipos de planos (detecção de viabilidade ou execução de objetivos), suas consequências (grande sucesso ou fracasso), intenções (desonestas ou legítimas) ou suas soluções (arriscadas ou esperadas). Assim, situações semelhantes seriam resgatadas para um objetivo “pedir empréstimo” e condição “pagamento em longas parcelas”, mesmo se os seus casos e generalizações existentes não estivessem no mesmo domínio do conhecimento.

2.3. Raciocínio Baseado em Casos

Antes desta seção, o termo Raciocínio Baseado em Casos (RBC) foi pouco abordado. Este fato foi proposital, pois desta maneira acredita-se apresentar a teoria da Memória Dinâmica de forma mais independente e permitir a possibilidade da criação de outras relações entre o RBC e outros modelos cognitivos que não foram estabelecidos neste trabalho.

Como um objetivo mais amplo, o Raciocínio Baseado em Casos se propõe a ser um meio para a criação de uma máquina inteligente, um antigo sonho da Inteligência Artificial (Kolodner 1993). No RBC, novos problemas são abordados, levando-se em conta referências a experiências anteriores semelhantes e tratando a solução como algo incremental e não integral. Os problemas e as situações são avaliados para que uma solução adequada seja encontrada, fazendo inferências e questionamentos, quando necessário, para completar as lacunas. Como resultado, o aprendizado surge do enriquecimento da memória com novos casos.

Torna-se, então, necessário apresentar os elementos formadores do RBC e suas relações para que sejam traçadas estratégias e escolhidas ferramentas e métodos que possibilitem a construção de um sistema ou dispositivo com características inteligentes baseado em experiências.

2.3.1. Estrutura do Conhecimento

Aqui será tratada a definição formal do conhecimento, suas partes constituintes e sua relação com a experiência.

Antes de definir diretamente o termo conhecimento, é importante conceituar seus elementos formadores, sob a abordagem da ciência da computação. Segue, segundo Bergmann (2002):

- Dado: são padrões sem contexto e podem ser armazenados e processados por dispositivos computacionais. Os dados, por sua vez, são formados pelos caracteres, suas estruturas precursoras, que incluem letras, números e símbolos;

- Informação: é o dado com significado, ou seja, já foi inserido em contexto.

Assim, o conhecimento é uma coleção de informações que se relacionam em um contexto mais amplo que a informação e é muitas vezes orientado pelos objetivos.

Com as definições citadas acima e a apresentação do modelo cognitivo, fica mais fácil definir formalmente o termo experiência. A experiência é conhecimento contextualizado armazenado na memória, resultado da solução direta de um problema ou interpretação de uma situação.

É importante notar que o termo problema não é usado exclusivamente como o único tipo de situação a ser resolvida (Kolodner 1993), pois existem casos que demandam tipos diferentes de solução, como processos jurídicos e argumentações, que na sua grande maioria esperam respostas discursivas na forma de justificativas ou críticas. Estas situações diferentes são conhecidas como interpretativas.

2.3.2. Raciocínio Baseado em Casos, seus Tipos e Formação

Após apresentação da teoria da Memória Dinâmica, é o momento de definir o RBC. Segundo Kolodner (1993), RBC é um modelo cognitivo e também um método para a construção de máquinas inteligentes, baseando-se nas suas experiências para solucionar novas situações ou casos.

2.3.2.1. Tipo

Partindo do princípio que o RBC deve possibilitar a criação de máquinas inteligentes, é interessante expor a extensão com que esta tarefa pode ser tomada, analisando a sua amplitude de tipos de RBC (Kolodner 1993).

Os sistemas de RBC podem ser criados com a intenção de contemplar todo o espectro das características do seu modelo cognitivo, como manter diversas opções de indexação, busca, inferência e adaptação, com maior grau de automatização possível, reduzindo assim a interação com ambiente externo para obtenção de caminhos para a solução.

No extremo inferior desta escala se encontram os sistemas que fazem uso somente de algumas de suas características para resolver problemas, como, por exemplo, a busca e o armazenamento de casos. Apesar de parecer insuficiente, esta decisão possibilita ações relevantes de um sistema de RBC, como a recuperação de informações em um contexto de incertezas e a interação com usuários ou outros sistemas para que estes forneçam as direções para uma solução e o posterior armazenamento para reaproveitamento futuro, o que implica implicitamente em um sistema de indexação e armazenamento.

É importante ter em mente que a abrangência de características de uma proposta de um sistema RBC está sujeita a diversos fatores, como domínio do problema, objetivos, recursos existentes e premissas de desempenho, não havendo necessariamente a obrigatoriedade de se utilizar todos os processos de seu ciclo de vida

2.3.2.2. Formação

A escolha dos elementos formadores de um sistema de RBC é muito importante, pois seus principais processos formadores serão baseados nesta escolha. Apesar das várias diferenças de abordagens para se representar o conhecimento, ele pode ser dividido em três elementos fundamentais (Bergmann 2002), que são o vocabulário, a base de casos e conhecimento reutilizável. Este três elementos formam o contêiner de conhecimento. A base de casos é o conjunto de casos armazenados. O conhecimento reutilizável é o conhecimento geral relativo ao uso de experiências pregressas, podendo assumir diversas formas, como possibilitar a detecção da similaridade entre casos e permitir a transformação da experiência para ser adaptada a um novo problema. Por sua vez, o vocabulário permeia a base de casos e o conhecimento reutilizável, pois define as entidades de informação e estruturas que representam estes dois elementos.

Por motivos didáticos a formação dos casos será separada em casos, base de casos, suas representações e índices. Devido à grande importância dos processos que compõem o ciclo de vida do RBC, eles serão tratados na seção 2.3.3.

Após a exposição de termos como experiências, problemas e situações, é o momento de definir o que é um caso. Um caso é o conhecimento adquirido a partir da experiência de uma situação concreta prévia para se atingir um objetivo (Kolodner 1993) (Aamodt e Plaza 1994). Em outra definição (Watson e Marir 1994), caso é uma situação anterior.

Em linhas gerais os casos são formados pelas seguintes partes (Kolodner 1993):

- A situação ou problema: é a representação do ponto de partida para a atuação do RBC, podendo contemplar objetivos, restrições, características da situação e as relações entre os objetivos e restrições;
- Solução: a resposta para a situação ou problema. A solução contempla a resposta propriamente dita e também pode conter os passos trilhados para atingir a resposta, as justificativas, as opções aceitáveis e inaceitáveis, motivo pelo qual não foram escolhidas e expectativas de execução da resposta ou solução;
- Desdobramentos da solução: é o resultado da execução da resposta ou solução. Esta parte tem a função de fornecer um retorno sobre o status desta execução, como se houve falha ou sucesso, explicações,

diferenças entre expectativas e realizações e até mesmo uma possível estratégia de reparação.

Existe outra abordagem no tocante à formação dos casos (Bergmann 2002), onde os casos, dependendo do domínio, podem ter estruturas diferentes, mas possuem basicamente duas partes. A primeira é a caracterização, que descreve o caso de modo a permitir a sua reutilização em um determinado contexto (objetivos, restrições e valores, dentre outros) e a segunda é a lição que, em um sentido mais amplo, contém todo o conhecimento adquirido com a experiência, ou seja, além da solução, possui as lições e caminhos percorridos, positivos ou negativos para se chegar à solução.

Por sua vez, uma base de casos é um recipiente onde os casos são armazenados ao longo do tempo. Cada um dos casos tem um significado próprio e está disponível para uma nova utilização. A base de casos também pode conter generalizações de casos, tornando-se possível inferir através destas, se não houver nenhum caso mais específico semelhante a uma situação que deve ser resolvida.

Uma base de casos pode ser organizada de diversos modos, sempre tendo em mente medidas de desempenho, como eficiência, precisão e facilidade de manutenção, de tal modo que os processos formadores do RBC tenham os seus objetivos atingidos.

A maneira mais simples de armazenar os casos para uso direto é a sua retenção na memória volátil (Bergmann 2002), mas isto só é possível para bases com poucos casos, ao passo que grandes quantidades de casos necessitam de bancos de dados. Como atualmente existem diversos tipos de banco de dados, relacional, orientado a objeto e hierárquico, dentre outros, torna-se imperativo compatibilizar o armazenamento com a estrutura de dados e as medidas de proximidade entre os casos armazenados e com as novas situações.

Outro ponto importante é a representatividade da base de casos, ou seja, ela deve conter uma grande diversidade das situações do domínio, caso contrário, poucos casos apresentados se utilizarão do conhecimento armazenado, o que gerará adaptações e casos não resolvidos. Para minimizar esta situação, é interessante, ao montar a base de casos, inicializá-la (Kolodner 1993) com casos representativos e efetuar o treinamento com diferentes tipos de situações para que possíveis lacunas de conhecimento sejam identificadas.

Seguindo na formação dos casos, a representação destes trata das diferentes estruturas usadas. Em uma das propostas de representação existentes, coloca-se que existem três abordagens e elas diferem basicamente no conhecimento do domínio que podem representar, nas suas estruturas e disponibilidade da informação para alimentar a base de casos (Bergmann 2002).

- Textual: a informação dos casos pode ser armazenada de forma discursiva, em que a recuperação é realizada por uma máquina de buscas. A facilidade é que relatórios, laudos e outras informações

representadas da mesma maneira são facilmente transpostos para a base de casos mas, em contrapartida, a precisão da recuperação está ligada aos métodos de busca, muitas vezes sintáticos (palavra chave) e pouco semânticos;

- Interativa: o objetivo desta abordagem é ser usada onde a coleta dos casos é obtida através de um diálogo entre operador/usuário ou fornecedor/cliente e um caso é representado por uma lista de perguntas e respostas que variam entre os casos. Não existe uma estrutura padronizada para todos os casos da base. O autor dos casos deve ordenar as perguntas feitas ao usuário ou cliente, pois estes questionamentos são usados para organizar o caso na base de casos (Menegazzo 2000) na forma de árvore de decisão. Este método é muito útil em domínios nos quais os problemas são simples e a repetição é alta. A desvantagem é que requer grande esforço de manutenção, pois alterações na estrutura dos casos e na importância dos atributos geram necessidade de intervenção manual. Um exemplo é a proposta de uma abordagem interativa para a recomendação de produtos onde os motivos de cada questionamentos são explicados (López de Mántaras, et al. 2005).
- Estrutural: engloba as abordagens representativas que permitem aos computadores tratar de maneira mais eficiente o conhecimento armazenado. Como elementos mais comuns desta abordagem, existem as tabelas de atributos e valores, orientação a objeto, grafos e lógica de predicados. Cada um deles está descrito a seguir:
 - ✓ Atributo-valor: é uma das mais usadas. Toda a informação é representada por pares atributo-valor, sendo que a definição do tipo de cada atributo está ligada à aplicação do domínio. Os tipos podem ser diversos, como inteiros ou reais, símbolos e *strings*. É interessante lembrar que os casos não precisam necessariamente conter todos os atributos possíveis do domínio, mas todo atributo deve conter um valor, desta maneira é possível armazenar somente a informação relevante para o processamento do raciocinador. Exemplo: uma tabela com os endereços dos moradores de uma cidade, contendo atributos como bairro, rua, número do apartamento ou casa, nome completo e CEP;
 - ✓ Orientação a objetos: é um paradigma que permite relações hierárquicas e de herança, além de facilitar a modelagem de domínios complexos. Estes objetos possuem a sua descrição feita por pares de atributo-valor, mas também fazem parte de uma classe, o que pode ser visto como um conjunto ou subconjunto de atributo-valor presente em algum ponto da hierarquia de objetos e que está relacionado ao objeto em questão. Os atributos podem ser do mesmo tipo que os já citados na abordagem atributo-valor acima e também podem ser do tipo relacional, que faz a ligação entre objetos, criando assim a hierarquia de objetos ou casos. Exemplo: uma fábrica cria veículos para diversos usos, mas cada um deles compartilha características em comum, como vidros

elétricos, banco de couro e opções de pagamento, mas todos têm suas seus próprios tipos de motor e sistema de ar condicionado;

- ✓ Grafos: grafos são conjuntos de pontos, também chamados de nós ou vértices, que têm conexões entre si, parciais ou totais. Esta abordagem é muito útil se for necessário representar casos que tenham relações semelhantes às de árvores ou grafos. Um exemplo é a representação de um caminho para se chegar de um ponto A ao B através de pontos intermediários. Neste exemplo a representação de grafo é interessante, por refletir a relação de vizinhança entre os pontos;
- ✓ Lógica de predicados: usada quando é necessário estabelecer relações de condição e ação entre uma pequena quantidade de casos com objetivo de gerar conclusões e não resultados.

É importante comentar que quando o termo *caso* é citado na representação de casos, trata-se implicitamente de casos concretos, ou seja, casos com todas as suas descrições, sem nenhuma generalização ou abstração. Os casos abstratos são tratados através do conceito de *hierarquia de casos* (Bergmann 2002), e são o resultado de um processo de abstração que reduz a quantidade de casos e seus atributos à medida que se ascende na hierarquia de abstração. Dentre as vantagens desta abordagem, tem-se a redução do esforço computacional na realização de operações e aumento da cobertura de influência no espaço de casos, facilitando o uso em testes com protótipos. Os casos abstratos não serão tratados neste trabalho, para maiores informações, consultar (Bergmann 2002) (Borner 1994).

Finalizando a formação dos casos, indexação é a seleção das descrições ou características de casos de modo a facilitar sua futura recuperação e comparação (Pal e Shiu 2004). O objetivo dos índices é melhorar o desempenho na recuperação de casos em relação a uma busca serial pela base de casos. A escolha dos índices deve ser feita com base em alguns pontos importantes (Kolodner 1993):

- Características preditivas: os índices devem refletir as características importantes de um caso de acordo com as decisões a serem tomadas pelo raciocinador. Se for necessário recuperar uma solução baseada nos seus objetivos e restrições, para em seguida analisar as suas conseqüências, decisões corretas e incorretas, então deve-se ter em mente o uso de diferentes índices, um para cada situação, de tal modo que os melhores resultados possíveis serão obtidos para cada tipo de busca;
- Abstração: a abstração reflete a característica dos casos serem recuperados em um cenário mais amplo para sua aplicação, pois esta é capaz de prever usos similares em características que são especificamente diferentes;
- Concretude: concretude é o contrário da abstração. Se um índice for exageradamente abstrato, ele deve demandar um grande esforço de inferência para encontrar o caso mais adequado, sendo assim, um

caso deve ter um equilíbrio adequado entre generalização e especificidade.

Em RBC, a criação de índices é um processo muitas vezes manual e se aplica a situações de indexação de casos complexos, quando houver necessidade de grande precisão no processo de indexação ou dificuldade de manipulá-los computacionalmente. Em contrapartida, quando não existem as exigências listadas acima, a indexação no RBC pode ser trabalhada de modo automático (Kolodner 1993) (Watson e Marir 1994) e é normalmente dividida nas seguintes categorias:

- Baseada em *checklist*: os *checklists* são um conjunto das características dos casos que são criados por um especialista e usados pelo sistema RBC dependendo do tipo de busca a ser realizada. A criação do *checklist* é feita por um profissional experiente no domínio dos casos onde um conjunto de atributos é escolhido com um objetivo preditivo. Apesar deste tipo de indexação possibilitar um tratamento computacional mais eficiente, é importante realçar que a sua precisão na seleção de casos relevantes é tão boa quanto a escolha das características que formam o *checklist*;
- Baseada na diferença: fundamenta-se na análise dos casos da base de casos para extrair quais são as características que mais os diferenciam. É interessante notar que nem sempre estas características são as mais relevantes, isto reforça a necessidade de boas escolhas para índices de tal modo que sejam preditivos. Para ajudar nesta necessidade preditiva, este método é muitas vezes utilizado em conjunto com o *checklist*;
- Baseada em explicação: este método é exaustivo computacionalmente, pois analisa cada caso para detectar quais características são as melhores para se identificar porque um caso teve sucesso ou não na resolução de um problema.

Como métodos tradicionais de indexação, podemos citar os índices de bancos de dados relacionais, que fazem referência a chaves primárias e secundárias de registros, as estruturas em árvores e suas variações, como a as árvores kd (*kd-Trees*) (Wess, Althoff e Derwand 1994), árvores de decisão (Cardie 1993) e uso de árvores em conjunto com outras técnicas (Patterson, Rooney e Galusshka 2002), fazendo com que estas estruturas muitas vezes dividam o espaço de buscas pelos atributos, mas em contrapartida demandem um maior esforço para acomodar mudanças na sua estrutura. Outros métodos usados, mas menos tradicionais são as redes Bayesianas (Pal e Shiu 2004) que tem uma abordagem estatística de indexação, e as Redes de Recuperação de Casos (*Case Retrieval Nets*) (Bergmann 2002) que usam uma topologia de redes muito semelhante às Redes Bayesianas, mas sem abordagem estatística. Outra técnica diferente das anteriormente citadas é a Recuperação Baseada na Adaptação (*Adaptation Guided Retrieval*) (Smyth e Keane 1998), na qual as características para extração dos índices são decididas com base em informações que não são as que necessariamente compõem explicitamente

o caso, mas também as que são extraídas da correlação destas características explícitas que permitam a adaptação do caso com maior facilidade.

2.3.3. Ciclo de Vida do Raciocínio Baseado em Casos

Kolodner (1993) propôs um ciclo de vida de RBC com os seguintes processos:

- Recuperar, que inclui encontrar o caso mais próximo ao caso candidato;
- Adaptar, onde são realizados ajustes se a situação proposta for a solução parcial de um problema;
- Criticar (opcional), onde situações hipotéticas são geradas e as soluções encontradas são aplicadas a essas situações;
- Avaliar, validar a solução no mundo real;
- Armazenar os casos, constituindo a retenção do conhecimento.

Aammodt e Plaza (1994) propuseram um ciclo mais conciso na Figura 2.1e um conjunto de tarefas para cada processo, como ilustrado na Figura 2.2. Cada um dos processos do ciclo de Aammodt e Plaza será explicado em mais detalhes, pois é a proposta mais utilizada no RBC.

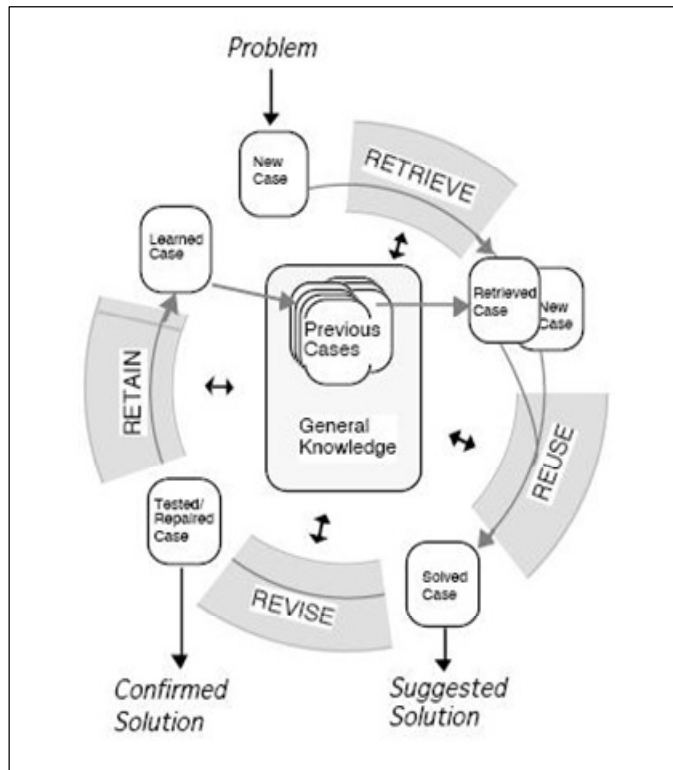


Figura 2.1 - Ciclo de vida do RBC (Aamodt e Plaza 1994).

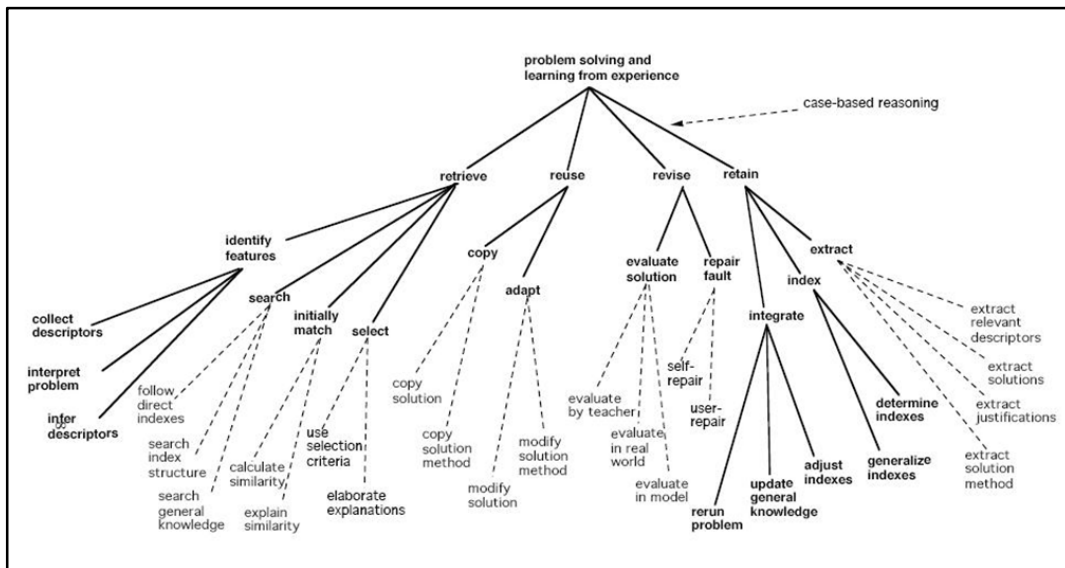


Figura 2.2 – Processos de RBC e suas tarefas (Aamodt e Plaza 1994).

2.3.3.1. Recuperação

Recuperação no RBC é o processo de selecionar um ou mais casos da base de casos que sejam úteis na resolução da situação ou problema proposto.

O processo de recuperação de casos é tido como o mais importante do RBC, pois cabe a ele encontrar casos na base de casos que forneçam a melhor solução para a situação a ser resolvida, além de ser pré-requisito para

os demais processos. De acordo com Aamodt & Plaza (1994), a recuperação é composta por quatro tarefas: identificação de características, busca, identificação de compatibilidade e seleção. Serão tratadas neste trabalho especificamente duas destas tarefas, a busca e identificação de compatibilidade (através do cálculo da similaridade), pois são as mais relevantes.

A recuperação de casos depende fortemente da maneira como os casos estão representados na base de casos, dos índices usados (quando aplicáveis) e dos algoritmos de busca e cálculo de similaridade.

O desempenho na busca é um fator muito importante para alguns domínios, principalmente em bases de casos com dezenas de milhares de casos. Assim como em um banco de dados comum, a recuperação de casos pode ser vista como um problema de busca, mas com uma importante diferença: não se espera encontrar um caso na base de casos exatamente igual ao caso procurado, mesmo porque os algoritmos de busca em bancos de dados procuram por valores exatos em relação à consulta. O RBC se baseia em semelhanças, não necessariamente em exatidão. Desta maneira, é de se esperar que algoritmos específicos sejam utilizados levando em conta a representação dos casos, tipos de atributos que compõem os índices, objetivo do raciocinador em determinada operação de busca, a precisão da solução e necessidade de buscas sequenciais ou paralelas (Kolodner 1993).

A seguir serão apresentados métodos de busca e cálculo de similaridade.

É interessante notar que, dependendo do método de busca, a medida de similaridade já pode ser calculada automaticamente ou ser uma tarefa posterior à busca, refinando assim os casos fornecidos pela recuperação.

De maneira geral existem duas abordagens de busca, baseadas no cálculo da distância entre os casos ou na sua representação (Liao, Zhang e Mount 1998). Na primeira abordagem, leva-se em conta que os mais similares são escolhidos com base na distância entre eles, segundo a avaliação da medida de similaridade. Na segunda abordagem, a busca está ligada à representação ou indexação dos casos, sendo que o método de buscas percorre os casos de acordo com a sua organização para encontrar o(s) mais similar(es) dentre ele(s).

Existem diversos métodos de recuperação (Bergmann 2002) (Watson e Marir 1994), e como em alguns pontos os métodos citados por cada autor são complementares, eles serão expostos em conjunto:

- Força bruta: é uma recuperação sequencial, na qual se percorre toda uma tabela, vetor, arquivo ou lista para encontrar o caso mais similar em relação à situação apresentada. Se usado com um método adequado de similaridade, o melhor caso será realmente recuperado, em contrapartida todos os casos devem ser pesquisados, o que é inviável para grandes bases de casos. Uma das medidas de similaridade mais comuns usadas com este método é a do vizinho mais próximo (*Nearest-Neighbor* - NN);

- Indução: identifica as características que melhor descrevem os casos, o que gera um armazenamento hierárquico, desta maneira o espaço de buscas e o tempo são reduzidos. Como exemplo, pode-se citar as árvores de decisão (Cardie 1993) e, de certo modo, também a kd-tree (Wess, Althoff e Derwand 1994), pois a sua montagem leva em conta a dispersão dos dados dos casos, que é derivada da análise dos casos da base. As árvores fornecem uma busca eficiente, mas não respondem bem a casos com características incompletas;
- Indução guiada pelo conhecimento: é uma abordagem semelhante à abordagem da indução, só que as características relevantes dos casos são manualmente selecionadas para se obter uma recuperação com maior precisão. Este método é mais usado em situações de conhecimento intenso, como na busca por explicações de falha ou sucesso no encontro de casos semelhantes ou na sua adaptação.
- SQL ou recuperação de padrões: a sua execução pode ser vista como uma consulta SQL que pode aumentar ou diminuir sua abrangência para atender ao número mínimo ou máximo de casos que obedeçam ao critério estabelecido (este processo é conhecido como relaxamento da consulta). Este método é muito usado como precursor de outros métodos, pois reduz o espaço de buscas de maneira simples, de tal modo que um método como o vizinho mais próximo e divisão da base de casos possam ser tratados, evitando processamento desnecessário. Este método evita a compilação de índices e é útil em bases de casos muito dinâmicas (Bergmann 2002);
- Recuperação validada: este método utiliza duas etapas e recupera na primeira delas um maior número de casos levando em conta as características mais superficiais (são as características existentes na descrição do caso). Em seguida, efetua-se um tratamento mais complexo, que pode necessitar da extração de características (características derivadas). Este método não exige uma análise complexa em todos os casos mas, sim em um universo menor retornado pelo primeiro estágio da busca (Simoudis 1992).

Bergmann (2002) inclui um método chamado indexação (métodos que utilizam índices), que contempla o de indução, as redes de recuperação de casos (eficientes para lidar com a falta de informação, mas demandam grande esforço na sua criação) e a recuperação Separação e Diminuição (*Fish and Shrink*), sendo este último voltado para situações complexas de similaridade.

A similaridade é o termo usado para quantificar o grau de semelhança entre casos. Esta tarefa é de vital importância, pois cabe a ela retornar os casos mais relevantes para a solução do caso corrente. Na comunidade de Inteligência Artificial (IA), assim como na de RBC, existe a idéia de que problemas semelhantes têm soluções semelhantes e muitas medidas de similaridade são construídas segundo esta condição. No entanto, há diversas abordagens mostrando que esta similaridade ligada às características diretas do problema (superficiais) não são uma garantia da recuperação dos casos mais relevantes, pois existem domínios em que o sucesso do caso recuperado

está vinculado à sua capacidade de adaptação e utilidade em resolver a situação ou problema apresentado (Smyth e Keane 1998) (Gabel e Stahl 2004). Muitas vezes são as características mais profundas, extraídas das características superficiais que possibilitam o encontro de uma solução satisfatória (Borner 1994) (Simoudis 1992).

É interessante ressaltar que o objetivo é o mesmo, encontrar casos que representem a melhor solução. A escolha da medida de similaridade dependerá diretamente do domínio e do objetivo do raciocinador no momento. Em domínios ou estruturas complexas, é comum haver a intervenção humana nos últimos estágios para seleção dos melhores casos. As medidas de similaridade devem levar em conta, sempre que aplicáveis, as propriedades dos atributos superficiais, seus valores e também as características derivadas (profundas ou densas), extraídas das características superficiais.

Cada medida é utilizada dependendo dos tipos de atributos, domínio da aplicação, desempenho de processamento e precisão desejada. Para efeitos práticos no cálculo da similaridade, o seu valor será considerado no intervalo $[0,1]$, ou seja, 1 significa similaridade máxima e 0 nenhuma similaridade.

Dentre as medidas de similaridade, as mais comuns são as ligadas à distância (Xu e Wunsch 2005), e as demais ao aprendizado de máquina (Liao, Zhang e Mount 1998) (Pal e Shiu 2004), e características profundas dos casos. Para conhecer de maneira mais ampla a formação de medidas de similaridade, é importante entender o seu *framework*, ou estrutura geradora (Osborne e Bridge 1997). Devido ao seu intenso uso, serão tratadas aqui em mais detalhes as medidas ligadas a distância.

As medidas que usam a distância entre os casos são muitas vezes utilizadas com a representação do caso na forma do par atributo-valor, sendo assim, assumiremos que os casos são formados por um conjunto de atributos em que cada um tem um tipo pré-definido e x e y são vetores de atributos a serem comparados. A partir da definição de x e y como $x = (x_1, x_2, \dots, x_{i,\dots}, x_n) \in D$ e $y = (y_1, y_2, \dots, y_{i,\dots}, y_n) \in D$, cada atributo pode ter a sua importância incluída no cálculo da similaridade. Sendo w_i o peso ou importância de cada atributo e $w_i \in [0,1]$.

É importante ressaltar que as funções de distância são usadas para medir variáveis contínuas e que a distância é uma medida de dissimilaridade, ou seja, quando maior distância, menor a similaridade e vice-versa. Note-se que não é feita nenhuma exigência quanto ao atendimento de alguma propriedade específica (Xu e Wunsch 2005) de distância ou similaridade. As principais medidas de distâncias são apresentadas na Tabela 2.1.

Nome	Equação	Comentários
Euclidiana	$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$	Muito utilizada para cálculos de valores reais. É a medida de distância mais usada.
Euclidiana ponderada	$d(x, y) = \sqrt{\sum_{i=1}^n w_i^2 (x_i - y_i)^2} \quad (2.2)$	Cada atributo possui um peso, indicando a sua relevância na composição do caso.
Manhattan	$d(x, y) = \sum_{i=1}^n x_i - y_i \quad (2.3)$	Também conhecida como <i>City-block</i> . Um exemplo simples de uso é a distância entre peças de xadrez, onde o caminho a ser percorrido não é uma linha reta.
Hamming	$d = \sum_{i=1}^n \delta_i, \quad \text{onde}$ $\delta_i = \begin{cases} 1 & \text{se } x_i \neq y_i \\ 0 & \text{em outros casos} \end{cases} \quad (2.4)$	Hamming foi concebida para tratar valores binários e é muito usada para calcular similaridade de atributos simbólicos.
Minkowsky ponderada	$d(x, y) = \left(\sum_{i=1}^n w_i x_i - y_i ^k \right)^{\frac{1}{k}} \quad (2.5)$	É uma generalização da distância euclidiana e Manhattan, quando $k=2$ e $k=1$, respectivamente.
Tchebyshev	$d(x, y) = \max\{ x_1 - y_1 , \dots, x_n - y_n \} \quad (2.6)$	É um caso especial onde o valor de $k \rightarrow \infty$ na distância de Minkowski.

Tabela 2.1 - Principais medidas de distância

Algumas medidas diretas de similaridade (SM) baseadas em distância estão listadas na Tabela 2.2:

Nome	Equação	Comentários
Taxa de Tversky	$SM(x, y) = \frac{\alpha(\text{comum})}{\alpha(\text{comum}) + \beta(\text{diferente})} \quad (2.7)$	Mede a similaridade com base na frequência de atributos comuns. Os dois coeficientes, ou pesos, α e β são obtidos a partir de intervenção humana ou processo de aprendizado de máquina.
Hunt	$SM(x, y) = D_H + \sum_i 2^{l_i} \quad (2.8)$	Onde D_H é a distância de Hamming total dada pela Equação (2.4), e l_i é o comprimento l de cada parte i da região que possui mais do que 2 atributos iguais.

Tabela 2.2 – Medidas baseadas em distância (não utilizam exclusivamente a distância).

As medidas das distâncias podem não ser usadas diretamente para medir similaridade, mas, sim, como um estágio anterior ao cálculo da similaridade propriamente dita. Como exemplo, pode ser citado o algoritmo *k*-NN (*k*-Nearest Neighbor). Este algoritmo é tido como um dos mais simples da área de aprendizado de máquina e tem por objetivo classificar um elemento com base nos votos dos seus *k* vizinhos mais próximos já previamente classificados. Depois que um ponto específico, ou caso, tiver a sua distância calculada em relação aos seus *k* vizinhos mais próximos, ele será classificado como membro da classe que tiver mais representantes dentre os vizinhos.

Além das medidas de distância descritas, existem outras que lidam com a extração de informação das características superficiais para derivar outras características ou informações que facilitem a tomada de decisão no reaproveitamento ou adaptação de um caso. Em uma abordagem interessante (Smyth e Keane 1998) questiona-se a utilização das características superficiais (presentes de maneira explícita na parte problema do caso) na recuperação e cálculo de similaridade de casos baseados na idéia de que estas características não fornecem garantia que um caso recuperado possa ser reutilizado. O seu tratamento foi baseado na relação direta entre o caso corrente e os casos candidatos à solução através do conhecimento explícito de adaptação presente na base de casos. Se houver casos na base de casos que possam ser adaptados (ou utilizados diretamente sem adaptação) para uso no problema ou situação apresentada, estes são recuperados e selecionados de acordo com um critério de custo global de casos, sendo que o menor custo

representa o caso mais viável para ser utilizado como solução. Esta estratégia evita a recuperação de casos que aparentemente têm uma solução semelhante, devido à compatibilidade entre suas características superficiais, mas que geram um esforço intenso de adaptação para serem utilizados.

Descritas as principais características do processo de recuperação, é o momento de introduzir um método que constitui fonte de inspiração para contribuições no RBC. O método, relacionado com o Sistema Imune Artificial (SIA), será apresentado no Capítulo 3, e a correlação entre os dois métodos será abordada no Capítulo 4.

O método é a Recuperação Através de Referência (*Footprint-Based Retrieval*) (Smyth e McKenna 1999). Este método é diferente de outros, como a criação de estrutura de árvores, redes de recuperação e busca completa em uma base de casos, já descritos anteriormente com as suas vantagens e desvantagens. Esse método possibilita a busca em uma quantidade menor de casos de uma base de casos, utilizando para isto casos mais representativos, como se estes fossem índices para o restante da base de casos, criando uma área de abrangência que engloba casos menos representativos ou comuns. A escolha destes casos mais representativos é feita baseada no critério da competência (Smyth e McKenna 1998), que é a amplitude de problemas que um sistema pode resolver, como ilustrado na Figura 2.3. Aplicado aos casos da base de casos, este critério postula que quanto mais casos um determinado caso possa resolver, mais relevante ele será para se tornar um caso representativo e fazer parte dos casos de referência na base de casos reduzida. Assim, os casos representativos criam áreas de abrangência que incluem casos não representativos, também chamados de auxiliares. Estas áreas de abrangência são também chamadas de conjuntos relacionados. A partir deste ponto, os casos mais representativos e seus auxiliares são agrupados em grupos de competência. A partir deste estágio o processo de busca pode ser iniciado. A busca funciona em duas etapas:

- Etapa 1: realiza-se a busca nos casos mais representativos da base de casos para encontrar o que tem a melhor solução aplicável ao caso corrente (candidato);
- Etapa 2: Após esta primeira busca pelos casos representativos (referências), o caso representativo mais adequado ao caso candidato é recuperado. Em seguida, uma segunda busca é realizada, agora em um conjunto de casos menos representativos, os auxiliares, que o caso representativo indexa. O processo termina com a seleção do caso mais adequado, seja ele o caso de referência ou um auxiliar na área de abrangência do caso de referência.

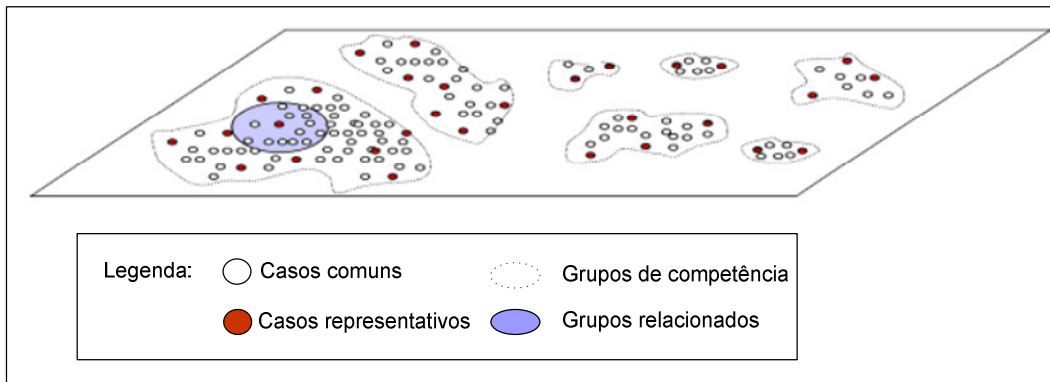


Figura 2.3 – Casos de referência e grupos de competência em uma base de casos (Smyth e McKenna 1999).

2.3.3.2. Reutilização

Como dificilmente os casos da base de casos são idênticos ao caso candidato, as soluções encontradas a partir da recuperação podem demandar alguma adequação para serem utilizadas. O processo de reutilização tem por objetivo adaptar os casos selecionados para que forneçam uma solução que atenda às premissas do caso apresentado. Um exemplo que ilustra a adaptação pode ser dado por um sistema que tem por objetivo preparar a receita de uma refeição (Kolodner 1993). O sistema possui diversas receitas cadastradas com vários tipos de ingredientes. O usuário do sistema insere informações do prato que deseja, pedindo que seja uma refeição italiana, de fácil preparo e para vegetarianos. Após pesquisa, o sistema seleciona uma receita de “refeição italiana comum”. O sistema decide que a refeição deve conter a entrada, uma massa, o prato principal e a sobremesa. O sistema começa a trabalhar no prato principal, sugerindo que seja uma lasanha, pois se encaixa na maioria dos critérios fornecidos pelo usuário, mas existem pontos a serem resolvidos, pois a lasanha é uma massa (já existe uma massa na receita) e possui carne, um ingrediente indesejado para vegetarianos. A partir deste ponto, a solução fornecida deve ser adaptada, pois ainda não obedece aos critérios do usuário. O sistema adapta a receita, retirando a massa antes do prato principal e mantendo a lasanha, mas substituindo a carne por espinafre.

Estas mudanças do sistema indicam alguns tipos importantes de adaptação, ambas ligadas à alteração e não a uma simples substituição de ingredientes.

Existem diversos métodos e técnicas de reutilização propostos por vários autores (Kolodner 1993) (Aamodt e Plaza 1994) (Watson e Marir 1994) (Pal e Shiu 2004) (Bergmann 2002). Por motivos didáticos, aqui será adotada uma abordagem mista de Kolodner e Aamodt & Plaza, visto que os últimos apresentam uma proposta mais sucinta e Kolodner mais detalhada, mas elas são complementares:

- Cópia: Aamodt & Plaza propõem que, se a solução do caso mais útil encontrado atende integralmente à demanda do caso corrente, ela deve ser copiada;

- **Substituição:** este método trata da substituição de valores em algumas partes do caso ou nele todo, preservando a estratégia da solução. Os valores substituídos podem ser do caso mais semelhante encontrado ou até mesmo de outros casos da base que possam ser úteis. Existem diversas técnicas propostas (Kolodner 1993) para executar a substituição, como reinstância, ajuste de parâmetros, busca local, pesquisa na memória, busca especializada e substituição de casos;
- **Transformação:** este método é mais amplo que a substituição e transforma a solução do caso recuperado de tal modo que ela funcione com o caso corrente. Existem duas técnicas propostas: a transformação de senso comum e a reparação guiada por modelo (Kolodner 1993);
- **Derivação:** a estratégia da solução é mantida, mas todas ou algumas das etapas da solução do caso mais útil encontrado são executadas para se chegar ao resultado final para o caso corrente. Um exemplo é a resolução de um exercício matemático, onde a solução não pode ser dada através de uma mera substituição ou transformação de valores, mas, sim, pela execução das mesmas etapas usadas para resolver o exercício recuperado da base de casos.

É importante deixar claro que a adaptação não se dá exclusivamente logo após o resgate do caso mais útil, mas pode ser realizada também em um estágio de reparação, ou seja, se uma adaptação não teve o resultado esperado após uma simulação ou aplicação no mundo real, ela pode ser novamente adaptada.

2.3.3.3. Revisão

Revisão é o processo de verificação do funcionamento da solução para o caso corrente. Geralmente a solução é verificada através de uma simulação de execução no mundo real, como no caso de uma receita médica ou tratamento. Basicamente este processo é constituído de dois subprocessos:

- **Avaliação:** aplicar a solução em uma simulação ou no mundo real para constatar o resultado dos processos anteriores;
- **Reparação:** este estágio pode, dependendo do sistema, gerar uma explicação para o motivo da falha e, com base nesta informação, fornecer uma nova adaptação que será novamente validada pelo mesmo processo de revisão.

O resultado de um tratamento unificado dos processos de revisão e reutilização é denominado de adaptação, pois, em muitas aplicações práticas, estes dois processos são difíceis de serem distinguidos (Pal e Shiu 2004) (Bergmann 2002).

2.3.3.4. Retenção

O processo de retenção é também conhecido como o momento do aprendizado, pois ele trata da incorporação de toda a informação útil obtida na resolução do caso corrente. Além de envolver a retenção da solução e lições do caso proposto (quando aplicável), este processo também absorve uma possível readequação dos índices dos casos, medidas de similaridade e estrutura do armazenamento resultante de uma necessidade de melhoria.

O armazenamento tem importantes desdobramentos além da retenção do conhecimento, pois, com o passar do tempo, a base de casos cresce em quantidade, armazena casos muito parecidos e pouco usados, o que gera o problema da utilidade (Smyth e Keane 1995) Este problema refere-se à relação entre o crescimento da base de casos e da consequente degradação do tempo de recuperação, pois a tendência é que, quanto mais casos existam na base de casos, mais fácil a solução, devido ao provável baixo esforço na etapa de adaptação (López de Mántaras, et al. 2005). Em contrapartida, quanto mais casos armazenados, menos eficiente será o processo de recuperação. Esta relação se sustenta até um ponto aceitável, a partir deste, o custo da recuperação supera o da adaptação e algo necessita ser feito para evitar que o problema cresça ou mesmo venha a acontecer. As primeiras abordagens lidaram com técnicas de edição de base, como Vizinho Mais Próximo Condensado (*Condensed Nearest Neighbor* – CNN), (Hart 1968), no qual os casos redundantes eram removidos da base de casos. Esta técnica inspirou diversas outras de diferentes autores, como Vizinho Mais Próximo Seletivo (*Selective Nearest Neighbor* - SNN), Vizinho Mais Próximo Reduzido (*Reduced Nearest Neighbor* - RNN) e Vizinho Mais Próximo Editado (*Edited Nearest Neighbor* - ENN), todas resumidas em (Wilson e Martinez 2000). Mais tarde, no início da década de 90, realizaram-se pesquisas ligadas à família de algoritmos baseados em instância (*Instance-Based* - IB) (Aha, Kibler e Albert 1991), que lidam com o aprendizado lento (*lazy learning*), no qual a generalização é realizada no momento da busca do caso mais útil. O uso destas técnicas no RBC leva ao aumento da velocidade na busca de casos pela base, mas gera um inconveniente, a perda de conhecimento relevante, que pode levar à incapacidade de solução de casos futuros.

Para tratar esta situação, foi proposto (Smyth e McKenna 1998) um modelo de competência, baseado na análise e contribuição individual de cada caso na resolução de problemas. Desta maneira, é possível encontrar uma medida segura para remover os casos menos relevantes da base de casos e, ao mesmo tempo, minimizar o problema da utilidade. A partir da percepção da necessidade de manutenção das bases de casos (Smyth 1998) e a proposta de um *framework* consistente que endereça diretivas para lidar com a manutenção (Wilson e Leake 2001), esta começou a constituir-se como uma área consistente do RBC, a ponto de ser proposta (Reinartz, Iglezakis e Roth-Berghofer 2001) a inclusão de dois processos no ciclo do RBC, chamados de revisão e restauração. O primeiro realiza o monitoramento da qualidade do conhecimento armazenado e o segundo as tarefas de manutenção propriamente ditas.

Cabe neste momento conceituar a manutenção de casos. A Manutenção Baseada em Casos (*Case-Base Maintenance* ou CBM) é um conjunto de diretivas e técnicas que envolvem a inserção, remoção e edição de casos e outros tipos de conhecimento em um sistema de RBC de modo a garantir a continuidade da eficiência e precisão. As atividades de manutenção podem ser separadas em duas grandes categorias, qualitativas e quantitativas (Pal e Shiu 2004).

- Manutenção qualitativa: trata da efetividade de um sistema de RBC. É dividida em:
 - ✓ Precisão: habilidade de resolver a situação ou os casos apresentados, o que depende da aplicação e domínio do RBC;
 - ✓ Consistência: trata da qualidade da solução em relação a outros casos existentes, de tal modo que não contradiga ou entre em conflito com um mesmo problema. Exemplo: casos redundantes, conflitantes (mesmo problema com soluções contraditórias), incorretos (soluções errôneas), alternativos (forneça soluções mais distantes que uma mais próxima e mais precisa) e isolados (incapazes de serem recuperados da base);
 - ✓ Completude: lida com a amplitude da ocupação do espaço do problema. A completude se baseia em algumas propriedades da competência, como a cobertura e o alcance (Smyth e McKenna 1999). A cobertura trata dos casos que um determinado caso da base consegue resolver e o alcance lida com a determinação de quais casos da base podem resolver o caso em questão.
- Manutenção quantitativa: trata da eficiência na solução dos casos ou situações apresentadas. É dividida em:
 - ✓ Controle do tamanho da base: a base de casos sempre cresce com o tempo, o que gera latência na recuperação. O número de casos mantidos na base deve ser tal que a competência do sistema seja mantida;
 - ✓ Revisão de índices: à medida que o domínio amplia seus conhecimentos e estes são refletidos nos casos, os índices devem ser analisados para refletir esta mudança e possibilitar uma busca eficiente e precisa;
 - ✓ Estatística de casos: envolve a coleta de informações sobre casos, como frequência de acesso, reutilização e custo da recuperação;
 - ✓ Detecção de casos irrelevantes: são casos que não têm mais valor, pois o seu conhecimento não é mais representativo;
 - ✓ Remoção de ruído: como muitos casos são manualmente inseridos na base de casos, casos diferentes podem conter a mesma solução e casos semelhantes podem conter soluções diferentes, logo, a remoção destes casos é necessária.

- ✓ Retorno dos usuários: em muitas situações do mundo real os usuários fornecem o seu retorno sobre o resultado da interação com sistemas de RBC, o que gera um montante de informações potenciais para melhorar o sistema em diversos pontos, como medidas de similaridade, técnicas de adaptação e modificação da estrutura de índices;
- ✓ Backup e recuperação: o conhecimento de um sistema de RBC deve ser resguardado através de procedimentos de backup e recuperação, assim ele estará disponível sempre que necessário, na eventualidade de um desastre ou acesso a uma informação mais antiga.

O modelo tradicional de RBC com seus processos de recuperação, reutilização, revisão e retenção é amplamente usado em aplicações e considera que os casos são formados por pelo menos duas partes bem definidas, problema e solução. Neste modelo, uma consulta é feita a um sistema de RBC na expectativa de recuperar casos que apresentem soluções em potencial para o caso apresentado. O caso mais similar ao caso apresentado é escolhido e, se necessário, uma adaptação é realizada. Esta abordagem leva em conta uma clara separação entre problema e solução, mas existem domínios que não possuem esta separação clara e naturalmente demandam soluções para seus casos. Com o objetivo de tratar este novo tipo de situação, o ciclo de RBC tradicional foi extrapolado com novos processos e técnicas baseados na utilidade (o quão útil um conhecimento é na resolução de um problema) (Stahl e Gabel 2006) (Stahl 2005).

2.3.4. Raciocínio Baseado em Casos e outros métodos de raciocínio

O RBC representa uma importante contribuição para a construção de sistemas ou máquinas inteligentes, pois armazena casos ou situações já vivenciadas, fornecendo experiências contextualizadas que possibilitam a solução de situações futuras semelhantes às armazenadas na sua base de casos. Desta maneira, o RBC contrasta com outros métodos usados, como abordagem baseada em regras e baseada em modelos. Apesar de cada uma delas tratar, de maneira peculiar, a solução de problemas, elas podem ser conjugadas de diferentes maneiras com o RBC para prover um sistema mais preciso e eficiente (Marling, Rissland e Aamodt 2005). A seguir são explicadas as abordagens historicamente mais ligadas ao RBC:

- Baseada em regras: no sistema de regras, o conhecimento é armazenado na forma de regras, SE A , ENTÃO B , onde A é uma condição (ou antecedente) e B é uma ação (ou consequente). Se A for verdade, B é executado. As condições também podem ser agrupadas para aumentar as restrições. Os sistemas baseados em regras também possuem uma máquina de inferência (Pal e Shiu 2004) que indica qual regra ou conjunto de regras deve ser colocado em ação para cada situação. Além desta característica determinante, estes sistemas tratam o conhecimento de maneira mais

compartimentada, não como um caso do RBC, mas em partes menores que possam ser processadas pelo conjunto de regras e trabalhadas em incrementos. Esta é uma grande diferença em relação ao RBC, que trata a situação a ser resolvida, ou caso, como uma parcela maior de conhecimento, formada pelo problema, solução e lição. Dessa maneira, um caso é bem maior que a parcela de conhecimento tratada pelas regras de um sistema baseado em regras e a solução é geralmente encontrada em menos etapas. Naturalmente, as regras trabalham com recuperações exatas de informação, já que obedecem a condições pré-determinadas, criadas durante a modelagem do domínio, e o RBC lida com aproximações, ou seja, uma solução não necessita ser exata, mas pode atender parcialmente os requisitos e ser aceitável para o caso em questão. O RBC também tem suas peculiaridades, pois o conhecimento está armazenado não somente nos casos, mas também em medidas de similaridade e índices. Assim, a diferença mais importante em relação ao RBC é o tempo e esforço na criação e manutenção de um sistema, pois no sistema de regras o domínio necessita ser muito bem compreendido para que elas sejam modeladas, tornando-se complexa a sua manutenção, caso o domínio de conhecimento apresente variações que necessitem ser refletidas no sistema. Como exemplo, um sistema usando regras demorou quatro meses para ser criado e, no RBC, apenas duas semanas (Kolodner 1993). Esta diferença na criação de sistemas não implica que as abordagens não trabalhem juntas, pois as regras podem inicialmente diminuir a amplitude de casos a ser pesquisada ou resolver tarefas mais simples, para depois passar os resultados ao módulo do RBC, que obterá a solução final;

- Baseada em modelos: este método é usado quando se conhece o modelo do domínio de tal modo que ele possa ser usado como meio para se chegar a conclusões ou soluções que partam de pontos em que já exista algum conhecimento que possa ser reaproveitado. O modelo aqui referido é a representação de como algum domínio ou dispositivo funciona, com as suas relações de causa e efeito. Neste ponto, a abordagem baseada em modelos é diferente da baseada em regras, onde a primeira lida com menos restrições, pois um modelo aborda os principais mecanismos de funcionamento e as regras são condições específicas para execução de uma ação. Neste sentido, a abordagem baseada em modelos tem mais interseções que diferenças em relação ao RBC, visto que o RBC lida com um conhecimento específico e contextualizado armazenado em uma base de casos, mas que eventualmente pode não oferecer uma solução adequada. Nesta situação, a abordagem baseada em modelos torna-se uma boa opção, pois provê, de modo geral o funcionamento de situações mais representativas, que apesar de não ser tão específico quanto o caso em questão, aumenta a possibilidade de fornecer uma solução adequada, se um caso não puder fornecê-la parcial ou completamente.

2.4. Exemplos de Aplicações

Nesta seção serão apresentadas aplicações de relevância histórica do RBC, tanto acadêmicas como implantadas no mundo real. As aplicações ilustram os processos do RBC, incluindo a similaridade, indexação e adaptação de casos para prover a solução mais útil ao problema ou situação em questão.

2.4.1. CYRUS

CYRUS é considerado o primeiro sistema construído em RBC e tem por objetivo armazenar e recuperar eventos da vida de Cyrus Vance, ex-Secretário de Estado dos Estados Unidos, na era Jimmy Carter (Watson e Marir 1994).

CYRUS faz uso dos MOPs (estruturas que armazenam conhecimento geral e organizam o conhecimento específico), de maneira modificada através dos Pacotes Organizadores de Memória de Episódios (*Episodic Memory Organization Packets* – E-MOPS). Nos E-MOPS, Kolodner vai além da estrutura original dos MOPs e propõe que os casos de E-MOPS estão organizados em uma estrutura hierárquica de especialização e generalização, ou seja, uma situação geral pode conter um ou mais casos específico ou outras situações gerais abaixo dela, criando assim uma teia de casos gerais e específicos interrelacionados. A memória de CYRUS foi organizada de modo a ser uma rede de discriminação redundante, na qual os índices foram criados através da técnica de *checklist*, já previamente apresentada.

CYRUS recuperava os casos percorrendo os E-MOPS até chegar à informação mais específica, esteja ela em um E-MOP ou em um caso. Se ele não tivesse todas as informações necessárias para tomar uma decisão, ele fazia inferências, usando as informações dos índices dos E-MOPS com o contexto ou objetivo da sua busca. CYRUS acrescentava casos à sua estrutura, quando eles se encaixavam abaixo da sua estrutura de índices. Se o caso ou E-MOP acrescentados sob um índice possuísem um vizinho, outro caso ou E-MOP, um E-MOP era automaticamente criado, desta maneira a memória sempre respeitava a especialização e generalização de casos, um funcionamento muito similar à memória humana.

CYRUS foi inspiração para muitos outros sistemas como MEDIATOR, CHEF, PERSUADER, CASEY e JULIA (Watson e Marir 1994).

Kolodner (1993) cita dezenas de aplicações acadêmicas e de uso prático implementadas em corporações.

2.4.2. Clavier

Clavier (Hennessy e Hinkle 1992) tem por objetivo escolher a melhor disposição para tratar termicamente peças de aeronaves em um autoclave na empresa Lockheed. Como entrada, ele recebe as peças que devem passar pelo tratamento e define suas diversas passagens pelo autoclave, de modo a

processar a maior quantidade de peças simultaneamente no menor tempo possível.

É importante ressaltar que não existe um modelo definido de funcionamento do autoclave, pois as peças mudam de tamanho, tipo de material e peso. Além disso, o autoclave funciona através da troca de calor por convecção e as correntes de ar, em conjunto com as peças, geram temperaturas diferentes em vários pontos do equipamento. Caso as peças não sejam tratadas adequadamente, elas são rejeitadas, o que desperdiça milhares de dólares.

Como o modelo do autoclave não é conhecido, os seus operadores criaram um pequeno acervo, onde catalogaram as principais disposições de peças e os resultados de cada tratamento. A cada nova operação eles consultavam este registro e faziam as adaptações necessárias, mas era demorado e dependia muito do conhecimento do operador. Na verdade, a equipe da Lockheed já usava o RBC, só que manualmente.

Um sistema de RBC foi construído e começou com vinte casos, chegando a 150 casos diferentes armazenados. Como cada peça é classificada de acordo com a sua prioridade, o processo de recuperação busca serialmente por disposições na base de casos que tenham o máximo de peças prioritárias na sua formação. Os casos são ordenados de acordo com a quantidade de peças prioritárias presentes em cada um deles. Clavier armazena as seguintes informações nos seus casos: as peças e suas posições relativas na mesa, a mesa e sua posição no autoclave, o tipo da disposição e estatísticas do processo (diferença entre maior e menor temperatura das peças e duração do processo).

É natural que, dependendo da situação apresentada, o caso recuperado necessite de adaptação. Esta adaptação se dá em duas etapas, a primeira é através de um contexto global, e em seguida, através de um contexto local, ambas usando estruturas de árvore de decisão. No primeiro tipo, os casos são buscados, levando em conta características gerais como tipo de molde, material e peças semelhantes no entorno da peça em questão. Em um segundo momento, uma busca mais refinada é executada, procura-se por informações para encontrar casos em que a disposição da peça a ser substituída esteja em região semelhante da mesa, no autoclave. O usuário pode optar pela escolha do sistema ou realizar uma adaptação manual para aperfeiçoar a solução.

Dessa maneira, Clavier solucionou 90% das situações sem necessidade de adaptação, permitindo um repasse mais rápido para que novos operadores inexperientes pudessem se inserir neste tipo de trabalho. Além destes ganhos operacionais, Clavier possibilitou a economia de milhões de dólares em desperdício de mão de obra e material.

2.4.3. SMART

Este sistema foi construído para prover um melhor serviço de atendimento aos usuários que adquiriram computadores da Compaq (atualmente uma divisão da HP). Os usuários ligavam para um número 0800 e eram atendidos por profissionais que usavam o sistema SMART (Cheetham e Watson 2005). O SMART é um sistema interativo, em que o atendente faz perguntas, cujas respostas ajudarão a recuperar os casos mais relevantes para resolver o problema do usuário. Como ele é um sistema interativo, leva-se em conta que os casos propostos, ou problemas, sejam repetitivos, o que é verdade em um sistema de atendimento a usuários deste tipo. A busca do SMART por casos era realizada de maneira incremental, à medida que as perguntas eram respondidas e os casos indexados pelas características vinculadas às perguntas eram resgatados. Os casos eram armazenados em um banco SQL relacional e o método de similaridade usado era o do vizinho mais próximo. Com estas características, 95% dos chamados de SMART eram resolvidos pelos profissionais de primeiro nível e 5% era repassado para profissionais mais experientes. SMART não fornecia adaptação e a inserção de novos casos na base era através de procedimentos específicos que garantiam a representatividade dos casos e, ao mesmo, tempo mantinham a base de casos em um tamanho aceitável.

2.4.4. FormTool

FormTool foi criada na divisão de plásticos da empresa General Electric (GE) e foi desenvolvida com o objetivo de criar plásticos que obedecessem às especificações de cor dos clientes. A maioria das técnicas usadas envolvia buscas exaustivas nas bases de dados para que os usuários dos sistemas escolhessem e tomassem a decisão final na escolha dos pigmentos formadores. Este processo está sujeito a erros humanos e também não leva em conta outros fatores importantes no processo de criação das cores (Cheetham e Graf 1997).

FormTool se baseia na decomposição dos fatores principais na formação da cor em plásticos, fornecendo limiares de aceitação para cada um deles e, depois, conjugando cada um destes fatores, de modo a emitir uma medida geral de similaridade entre a cor requisitada e a gerada pelo sistema. O FormTool trabalha com cinco características básicas: a semelhança entre cores, o volume total dos pigmentos, o custo financeiro dos pigmentos, densidade óptica (o quanto de luz passa pelo plástico) e a resistência à mudança da cor (os plásticos precisam ser aquecidos a altas temperaturas e isto pode gerar mudanças na sua cor).

A equiparação de cada atributo se dá através do uso de termos linguísticos, como ruim, médio, bom e excelente. Cada atributo tem seus valores, as diferenças entre eles são computadas através de variáveis linguísticas e, ao final do processo, é feita uma soma ponderada com seu peso, de forma a indicar alguma possível importância entre atributos. A soma é

dividida pela soma dos pesos e chega-se ao valor final, quando é escolhido o caso com a menor diferença em relação ao caso candidato.

A adaptação do FormTool é guiada pelo valor final do processo de equiparação citado acima. As decisões da adaptação são baseadas em uma teoria de cores específica, que fornece informações para decisão dos volumes de cores necessárias para se chegar a uma cor mais próxima à especificada pelo cliente, partindo do caso encontrado no processo de busca.

O uso do FormTool possibilitou à GE economizar milhões de dólares e também padronizou a maneira de se produzir plásticos coloridos em diversas subsidiárias da empresa.

2.4.5. Outros exemplos

Desde o desenvolvimento da primeira aplicação de RBC (Kolodner 1993) (Watson e Marir 1994) centenas de outras foram criadas nas mais diversas áreas, como:

- Detecção de diagnósticos de problemas (Goker, Howlett e Price 2005);
- Geração de planos (Cox, Muñoz-Avila e Bergmann 2005);
- Medicina (Holt, et al. 2005);
- Jurídica (Rissland, Ashley e Branting 2005);
- Educação (Kolodner, Cox e González-Calero 2005);
- Gerenciamento do conhecimento (Althoff e Weber 2005);
- Processamento de imagens (Perner, Holt e Richter 2005);
- Alocação de recursos em *grids* computacionais (Nassif, et al. 2005);
- Controle de processo siderúrgico (Jinsheng, et al. 2006) (Jinsheng, Jinchuan e Ru 2006);
- Recomendação (Bridge, et al. 2005);
- Reprodução de música e comportamento de personagens em jogos digitais (Lópes De Mántaras, Cunningham e Perner 2005).

2.5. Raciocínio Baseado em Casos e *Soft Computing*

Esta seção tem por objetivo relacionar *Soft Computing* com o RBC, mas para isto, é importante conceituar *Soft Computing* e introduzir algumas de suas principais técnicas.

De acordo com Zadeh (1994), *Soft Computing* é “uma abordagem emergente na computação que se assemelha à habilidade da mente humana no raciocínio e aprendizado em um ambiente de imprecisão e incerteza.” Segundo Pal e Shiu, *Soft Computing*:

“é de modo geral uma associação de técnicas e ferramentas de computação que compartilham disciplinas correlatas incluindo a lógica fuzzy, teoria das redes neurais, computação evolutiva e raciocínio probabilístico...”.

A composição do RBC com *Soft Computing* é chamada de *Soft CBR* (Cheetham, Shiu e Weber 2005).

2.5.1. Lógica Fuzzy

O ser humano interpreta informações incompletas e imprecisas através dos seus sentidos, tratando muitas vezes quantidades, sequências e ordens não por números, mas sim por grandezas mais genéricas expressas por variáveis linguísticas. A teoria dos conjuntos fuzzy oferece um arcabouço para lidar com este tipo de informação de maneira linguística e provê uma série de operações que usam estas informações linguísticas através de derivações de funções de pertinência (Jang, Sun e Mizutani 1997). A lógica fuzzy trabalha com a pertinência dos elementos de forma difusa em relação aos conjuntos, ou seja, um elemento não pertence necessariamente a um único conjunto mas, sim, a vários conjuntos com diferentes graus de pertinência.

A partir desta construção, a lógica fuzzy agrupa as variáveis linguísticas sob a forma de regras na forma SE-ENTÃO para criar o seu modelo de inferência, com o objetivo de aproximar esta representação do conhecimento humano.

As variáveis linguísticas podem assumir diversos valores como caro, barato, médio, pouco, muito, ruim, bom, excelente e assim por diante. Estes valores linguísticos são a representação de faixas ou valores quantitativos do que eles significam para as pessoas, como o preço de um carro ou a qualidade de um serviço, por exemplo.

Diversas aplicações no RBC (Cheetham, Shiu e Weber 2005) usam esta relação, como indexação, conversão de valores para facilitar a comparação e representação de atributos, o que faz da lógica fuzzy uma importante ferramenta no RBC, pois essas representações qualitativas são intrinsecamente fuzzy.

2.5.2. Redes Neurais Artificiais

As Redes Neurais Artificiais (RNAs) são sistemas de processamento inspirados no cérebro humano. As RNAs são formadas por modelos matemáticos dos neurônios (unidade de processamento) e suas ligações (conexões). Cada neurônio processa as informações que chegam até ele através de funções de ativação, onde as mais comuns são as lineares e radiais. As ligações ou conexões têm a função de conectar os dados de entrada da rede aos neurônios ou ligar camadas de neurônios. Além de estabelecer esta conexão, elas também possuem um peso atribuído, que realça a importância ou prioridade da conexão em relação a outras. No final do processo, as RNAs geralmente geram uma ou mais saídas, que são o resultado do problema resolvido pela rede.

As RNAs têm a vantagem de serem estruturas de processamento paralelo, pois elas são organizadas de tal modo que as entradas são distribuídas para diferentes neurônios. Ao mesmo tempo, elas são tolerantes a falhas, pois como a rede é formada por diversos neurônios, a falha pode ser minimizada aumentando o peso das conexões para os neurônios restantes e enfraquecendo a dos neurônios problemáticos.

As RNAs são muito eficazes na detecção de padrões, onde, por exemplo, fornecidos os dados ou um caso de entrada, um caso pode ser recuperado ou detectado de uma base de casos (Pal e Shiu 2004). Como as RNAs lidam com muitas entradas incompletas e imprecisas (ruídos), elas são de grande valia no RBC, em domínios onde estas características estejam presentes.

2.5.3. Computação Evolutiva

A inteligência natural é o resultado de um processo de milhões de anos de evolução, e a simulação desta evolução pode gerar resultados interessantes para se atingir graus de inteligência superiores (Jang, Sun e Mizutani 1997).

Segundo (Pal e Shiu 2004), “a computação evolucionária envolve técnicas usadas para resolver problemas de busca e otimização inspirados em princípios biológicos da seleção natural e genética.”

A Computação Evolutiva (CE) considera um indivíduo como sendo representado por um conjunto de atributos, onde um grupo de indivíduos compete por várias gerações, guiando-se por alguma função objetivo (Pal e Shiu 2004). Os melhores indivíduos de cada geração são selecionados e fazem parte da próxima geração seguindo, desta maneira, até que o objetivo seja atingido. O critério de parada se dá, geralmente, quando se encontra um grupo de indivíduos que represente uma solução adequada ao problema. Uma das aplicações mais comuns é encontrar pesos adequados dos atributos, de modo que eles reflitam o grau de importância de cada um na composição do indivíduo, ou caso e, desta maneira, melhore a precisão em sistema de RBC. Uma aplicação interessante da CE é a busca no espaço de buscas, quando um método de força bruta é inviável, devido à enorme massa de dados ou à dificuldade existente de se extrair conhecimento dos dados, de modo a reduzir

o espaço de buscas. Um algoritmo genético pode ser usado para efetuar a busca no espaço de buscas. Como esta busca utiliza métodos estocásticos e estes têm menor probabilidade de ater-se a mínimos locais, ela tem a vantagem de fornecer soluções em menor tempo, mas que talvez não sejam ótimas. Este comportamento é interessante no RBC, já que o RBC trabalha com aproximações que possivelmente serão adaptadas (Jang, Sun e Mizutani 1997).

Outro importante paradigma da computação inspirada na natureza, assim como a CE, é o Sistema Imune Artificial (SIA), que possui características interessantes em áreas como detecção de padrões, otimização, geração de diversidade, agrupamentos e outros tipos de problemas. Devido ao foco deste trabalho na utilização de SIA em RBC, SIA terá o Capítulo 3 dedicado à sua exposição.

Capítulo 3 - Sistema Imune Artificial

3.1. Introdução

O corpo humano frequentemente se depara com elementos estranhos que ele pode interpretar como ameaçadores ao seu equilíbrio, como vírus, bactérias, protozoários, fungos e compostos químicos, dentre outros.

O sistema imune é o sistema responsável, no corpo humano, por prover os mecanismos de defesa contra as ameaças. É formado por diversos componentes que realizam diferentes funções e interagem entre si. Para que o sistema imune execute sua função, de maneira geral, ele conta com mecanismos complexos que fazem o reconhecimento do corpo estranho e a ativação das células que eliminarão a ameaça. Para que seus efeitos sejam mais eficazes, ele conta com a manutenção de uma memória que salvaguarda características de reações específicas pregressas, objetivando reutilizá-las em situações futuras semelhantes. A memória é de especial interesse neste trabalho, especificamente a característica de endereçamento baseado no conteúdo, que tem aplicações relevantes no RBC (Hunt, Cooke e Holstein 1995).

O sistema imune é formado por órgãos, tecidos e células, estando todos eles distribuídos por diversas partes do corpo. Suas células existem em imensa quantidade e são produzidas pelos órgãos primários. Quando um agente externo penetra no organismo, mecanismos de proteção são acionados. Estes mecanismos possuem características muito interessantes, sob o ponto de vista computacional como, por exemplo, a detecção de padrões realizada pelas células do sistema imune ao identificar um agente externo, já que a diversidade destes agentes é imensa. Outra característica interessante é a atuação descentralizada e massiva de suas células, pois diversas delas podem ser usadas em diferentes locais para executar tarefas específicas, construindo assim um sistema distribuído que realiza atividades simultâneas. A memória é outro atributo do sistema imune que se dá na forma de armazenamento de um histórico de reações anteriores através da manutenção das células que combateram ameaças anteriores, tornando mais rápida uma segunda reação com estímulos semelhantes.

Inspirado nos mecanismos supracitados do sistema imune e em outros que serão aqui detalhados, surgiu o paradigma dos Sistemas Imunológicos Artificiais como uma abordagem computacional direcionada para resolução de

problemas, muitos deles complexos, como otimização, aprendizado de máquina, análise de dados e detecção de falhas.

Apresenta-se aqui o sistema imune dos mamíferos (aqui denominado como biológico para diferenciá-lo do artificial) como fonte de inspiração para o Sistema Imune Artificial, os conceitos e mecanismos que permeiam o sistema imune biológico e artificial, alguns exemplos de aplicações e sua relação com outros paradigmas computacionais.

A abordagem do Sistema Imune Artificial aqui descrita é, em parte, baseada nos trabalhos de de Castro (2001) e de Castro e Timmis (2002) devido à sua apresentação formal e abrangente do assunto.

3.2. Sistema Imune Biológico

Com a evolução no estudo da imunologia, a sua definição mudou sensivelmente ao longo do tempo, podendo atualmente ser expressa como um ramo da biologia que trata das reações de defesa que geram resistência às enfermidades (Klein 1990) ou um sistema que protege o indivíduo contra a ameaça de microorganismos (Tizard 1995). A este sistema damos o nome de sistema imunológico, ou sistema imune.

O sistema imune biológico é responsável pela defesa rápida e efetiva do organismo contra ameaças. O sistema imune reage de diversas formas às ameaças e possui grande capacidade de adaptação, ambas as características serão aqui explicitadas através do esclarecimento do funcionamento de seus mecanismos, formação e organização.

3.2.1. Breve Histórico

A origem da imunologia é atribuída a Edward Jenner, em 1796, pela criação da vacina da varíola a partir da injeção de amostras da varíola bovina em seres humanos saudáveis. Apesar de criar a vacina, Jenner não sabia de fato como operava a infecção e o sistema imune humano. No século XIX, Robert Koch provou que as doenças infecciosas eram causadas por microorganismos, ou patógenos, formados por vírus, bactérias, parasitas e fungos. O famoso químico francês Louis Pasteur deu grandes contribuições através da produção de vacinas, como a da raiva, mas também não compreendia corretamente o mecanismo imune humano. Foi em 1890 que Emil Von Behring e Shibasaburo Kitasato descobriram que a proteção da vacina era devida ao surgimento de agentes específicos que se ligavam aos elementos infecciosos. A estes agentes foi dado o nome de anticorpos.

Em 1882, Elie Metchnikoff realizou experimentos que resultaram na descoberta de células que eliminavam os microorganismos, denominadas fagócitos, mas Metchnikoff atribuiu pouca importância no seu processo aos anticorpos descobertos por Von Behring e Kitasato. Na década de 1890, Paul Ehrlich criou a teoria da cadeia lateral, a qual postulava que as células brancas, ou leucócitos, como as células B, possuíam diversas cadeias, ou receptores,

no seu entorno. Estes receptores se ligavam aos microorganismos estranhos, gerando um grande aumento de anticorpos. Desta maneira a teoria sugere que os próprios antígenos (moléculas reconhecidas pelo sistema imune e capazes de induzir uma resposta imune) eram responsáveis pela ativação da resposta imunológica através da seleção dos receptores que os reconheciam. Entre 1914 e 1955 pensou-se que a formação dos anticorpos, proposta por Ehrlich, era inadequada e os esforços foram direcionados na biosíntese de moléculas de anticorpo produzidas pelas células, onde a idéia básica era que o antígeno forneceria as informações necessárias para a ligação com o anticorpo. A esta abordagem deu-se o nome de instrucionista.

No início da década de 50, Jerne retomou a proposta dos anticorpos de Ehrlich, e considerou que existe uma diversidade natural de anticorpos e que a reação ao antígeno se dá pela ligação entre o receptor do anticorpo e o antígeno. Burnet trabalhou com o princípio resgatado por Jerne e afirmou que cada célula do sistema imune produz um único tipo de anticorpo que, em contato com um antígeno específico, gera a multiplicação de anticorpos. Segundo Burnet, a geração da variedade dos receptores dos anticorpos se dá de maneira aleatória e inicia-se na vida pré-natal. A esta abordagem geradora denominou-se teoria da seleção clonal. Como a variedade obedece a um processo aleatório, existem anticorpos que podem se ligar a células do organismo, algo indesejado, sendo então descartadas durante o processo de adaptação. Em 1971 Jerne, propôs que a eliminação das células auto-reativas era regulada pelo mecanismo de seleção negativa, o que afetava a diversidade e, conseqüentemente, a resposta do sistema imune. Através desta consideração Jerne propôs a teoria da rede imunológica. Em 1983, Susumo Tonegawa sistematizou a sua proposta do mecanismo gerador da diversidade das moléculas do anticorpo.

O conhecimento atual do sistema imune é resultado de aproximadamente 200 anos de descobertas e pesquisas que renderam diversos prêmios Nobel a cientistas como Emil Von Behring, Ehrlich e Metchnikoff, Jules Bordet, Karl Landsteiner, Medawar e Burnet, Jerne e Susumo Tonegawa.

3.2.2. Componentes

O sistema imune é formado por diversas partes, como tecidos (grupos de células que realizam juntas a mesma função), células e órgãos. O sistema imune consiste de várias linhas de defesa e estas são fortemente dependentes das células brancas, ou leucócitos. Os leucócitos executam diversas atividades e são divididos de acordo com os mecanismos de defesa dos quais participam. Os níveis de defesa ilustrados na Figura 3.1 podem ser classificados como (Janeway, et al. 2000) (Rensberger 1996) (Hofmeyr e Forrest 2000):

- Barreiras físicas: a pele, além de ser uma barreira mecânica (ela descama e assim contribui no descarte de microorganismos), libera secreções como o ácido láctico e lisozima que impedem a instalação de alguns microorganismos;

- Barreiras bioquímicas: os ácidos no estômago, a saliva, as lágrimas e o suor também contribuem para eliminar diversos microorganismos;
- Sistema inato: é considerado o primeiro nível de defesa contra muitos microorganismos. Este sistema atua direta e prontamente na eliminação de antígenos e também na ativação do sistema adaptativo, que é de suma importância em reações mais específicas. As células do sistema inato são formadas basicamente pelos macrófagos e granulócitos;
- Sistema adaptativo: é formado pelos linfócitos e os principais são as células B e T. O sistema adaptativo interage de maneira única com os antígenos, reconhecendo-os e adaptando-se a eles, mesmo sem nunca ter tido contato prévio. Este comportamento é vital para a manutenção da integridade do indivíduo, pois o sistema inato não é capaz de lidar sozinho com a imensa possibilidade de ameaças existentes, mas é crucial no combate aos agentes externos conhecidos por ele, enquanto a resposta adaptativa é formada, o que pode levar dias. Devido a sua importância neste trabalho o sistema adaptativo terá uma seção exclusiva.

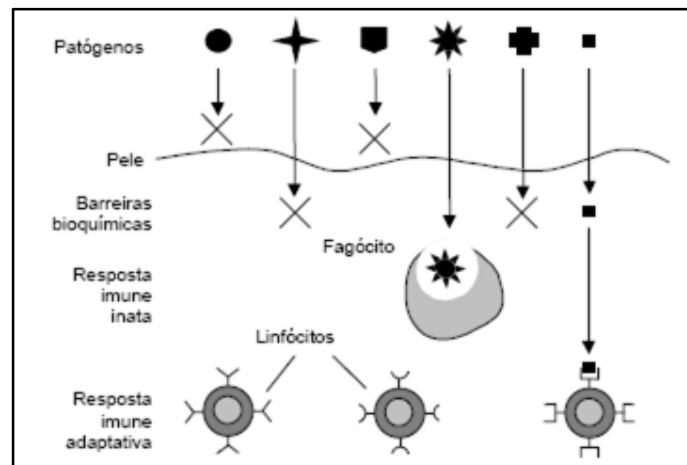


Figura 3.1 – Níveis de defesa do sistema imune (de Castro 2001).

3.2.2.1. Células

As células e órgãos do sistema imune funcionam juntos de maneira a proteger o indivíduo contra o que ele considera uma ameaça.

As células são divididas basicamente em três grandes grupos, os granulócitos formados pelos neutrófilos, eosinófilos e basófilos; os macrófagos, que são membros do sistema inato e os linfócitos (células T e B), que pertencem ao sistema adaptativo. Estas células, muitas vezes, trabalham em conjunto para prover uma reação imune mais efetiva. Os macrófagos e os neutrófilos são células consideradas essenciais na imunidade inata (Calich e Vaz 2001). Os macrófagos e neutrófilos realizam o processo da fagocitose dos microorganismos, que são internalizados por essas células. Os macrófagos, dentre outras células, também são responsáveis pela apresentação de

antígenos a outras células, como as células T, iniciando assim a ativação da reação adaptativa. O nome dado às células que apresentam antígenos é de células apresentadoras de antígenos (*Antigen Presenting Cell* - APC). Os eosinófilos e basófilos estão presentes, em baixas quantidades, se comparados com os outros tipos celulares, onde respectivamente o primeiro tem sua atuação ligada à eliminação de parasitas e, o último, a processos alérgicos.

As células T e B são centrais no processo da resposta adaptativa. As células T possuem receptores que permitem a identificação de fragmentos de antígenos apresentados por outras células, as APCs. As células T contribuem no processo imunológico através de duas funções: regulação da resposta imune e pelo ataque direto a células infectadas ou cancerígenas (U.S. National Institutes of Health 2003). Nestas funções, as células T separam-se em dois tipos, Th e Tk. A Th (*T helper*), ou auxiliar, que exerce a primeira função e a Tk (*T killer*), responde por T citotóxica, que exerce a segunda.

A célula B tem a característica de reconhecer partes solúveis de antígenos específicos, logo elas não dependem exclusivamente das APCs para ativação, como as células T. Quando ativadas pelas substâncias liberadas pelas células T, as células B se reproduzem e secretam anticorpos em grandes quantidades. Estes anticorpos irão se ligar aos antígenos, que serão posteriormente eliminados por outras células, como as células T citotóxicas.

As células T, B e os macrófagos são células ativadas por dois sinais, um primário e outro secundário, ou co-estimulatório. A ativação em duas etapas é um mecanismo menos sujeito a falhas, especialmente porque a decisão não é baseada em uma única célula e, em geral, um dos sinais é gerado por uma célula de tipo diferente da célula ativada (Freitas e Timmis 2007).

3.2.2.2. Órgãos

As células citadas anteriormente são produzidas, desenvolvidas e apresentadas aos antígenos em diferentes órgãos e tecidos do corpo humano. Seguem os órgãos mais importantes organizados pelas suas funções:

- Órgãos linfóides primários: responsáveis pela produção e desenvolvimento dos linfócitos.
 - ✓ Medula óssea: é um tecido esponjoso localizado no interior dos ossos longos onde são produzidas as células sanguíneas, incluindo as destinadas a atuar no sistema imune. Todas as células imunes começam como células tronco. Como estas células ainda não se especializaram em nenhuma função, elas são objeto de estudo no tratamento de algumas doenças imunes;
 - ✓ Timo: nele ocorre a proliferação e diferenciação de células precursoras em linfócitos T.
- Órgãos linfóides secundários: locais onde os linfócitos interagem com os antígenos para iniciar sua resposta imune.

- ✓ Linfonodos: também chamados de gânglios linfáticos, estão distribuídos por diversas partes do organismo e são ligados pelos vasos linfáticos. O transporte das células pelos vasos linfáticos é feito pela linfa, um líquido reabsorvido nos tecidos conjuntivos. Juntos, os linfonodos, os vasos, a linfa e outros componentes formam o sistema linfático, que é parte importante do sistema imune. Como os vasos linfáticos se conectam a diversos tecidos do corpo (sempre no sentido tecido/órgão para o linfonodo), antígenos presentes nestes locais são transportados e acabam passando pelos linfonodos, onde encontram células do sistema imune. Estas células circulam constantemente pelos linfonodos, como os linfócitos e macrófagos, onde iniciarão uma resposta imune (Calich e Vaz 2001);
- ✓ Baço: é o único órgão linfático interposto na corrente sanguínea, sendo importante na ação contra antígenos presentes na circulação. O baço não é vital para a sobrevivência mas, caso ele apresente problemas ou seja removido, o indivíduo estará mais suscetível a infecções;
- ✓ Outros órgãos: existem tecidos linfóides distribuídos por mais partes do corpo, como as placas de Peyer e as tonsilas palatinas, dentre outros, mas não serão aqui detalhados.

3.2.3. Sistema Imune Adaptativo

O sistema imune possui diversas propriedades, mas

as características mais importantes do sistema imune são especificidade e memória. A especificidade é a capacidade de reconhecer e reagir a determinada molécula e a memória é a capacidade de voltar a reconhecer e reagir rapidamente a esta molécula, quando esta for reintroduzida no organismo. As moléculas que são reconhecidas pelo sistema imune são chamadas de antígenos (Calich e Vaz 2001).

O sistema imune adaptativo é mediado por células altamente especializadas que eliminam ou previnem ameaças, conhecidas como linfócitos. O sistema inato é considerado não específico, pois ele não se adapta a um ataque de novos antígenos, mas possui participação ativa neste processo adaptativo, por apresentar os fragmentos antigênicos aos linfócitos T. O sistema adaptativo tem este nome porque se adapta aos estímulos antigênicos. Esta adaptação é realizada através de hipermutações somáticas nas regiões variáveis (regiões-V) dos receptores, um processo que opera mudanças nos receptores com altas taxas de diversificação e recombinação de genes que codificam proteínas usadas para formar os repertórios dos receptores das células T e imunoglobulinas, ou anticorpos (Oprea 1999).

As células B e T possuem receptores chamados BCR (*B-cell receptor*) e TCR (*T-cell receptor*), respectivamente. O TCR tem a função de reconhecer fragmentos antigênicos sob a forma de peptídeos que são apresentados pelas APCs, que inicialmente ingeriram e digeriram antígenos apresentados ao TCR.

Mais especificamente, os TCRs reconhecem os antígenos através da apresentação destes pelas moléculas do complexo de histocompatibilidade principal (*Major Histocompatibility Complex* - MHC) das APCs. Após sua ativação, as células T se multiplicam e secretam citocinas, substâncias que estimulam outros elementos do sistema imune, como as células B. Através do BCR, que corresponde ao anticorpo, estas células já estimuladas reconhecem antígenos solúveis e são ativadas. Em seguida se reproduzem e diferenciam em plasmócitos, que liberam grande quantidade de anticorpos usados para conectarem-se às partes reconhecíveis do antígeno, chamadas de epítomos.

3.2.3.1. Diversidade dos Receptores e o Reconhecimento de Padrões

Para entender a diversidade dos receptores dos linfócitos, especificamente dos linfócitos B, representados pelo anticorpo, é importante conhecer melhor a sua composição. Os anticorpos são basicamente formados por dois tipos de cadeias polipeptídicas, uma variável (com muitas variações na sua composição), de composição aminoterminal, usada no reconhecimento antigênico e outra constante (poucas variações na sua composição), de composição carboxiterminal, com funções efetoras que estimulam a destruição do antígeno pelo mecanismo do sistema complemento, como na Figura 3.2a. Os receptores das células T possuem semelhanças com os anticorpos, como as cadeias leves e pesadas, mas a sua estrutura é diferente.

Estudos mostraram que as cadeias polipeptídicas que formam os receptores BCR e TCR são codificadas por múltiplos segmentos gênicos do genoma. Estes diversos segmentos são combinados e codificados para formar diferentes anticorpos. Além do fator combinatório dos segmentos, ocorre a hipermutação somática, gerando mutação nos genes e contribuindo mais ainda para o aumento da diversidade dos anticorpos. É importante ressaltar que os leucócitos são monoespecíficos, ou seja, possuem somente um tipo de anticorpo na sua superfície, diferentes dos antígenos, que podem possuir diversos epítomos, como ilustrado na Figura 3.2b.

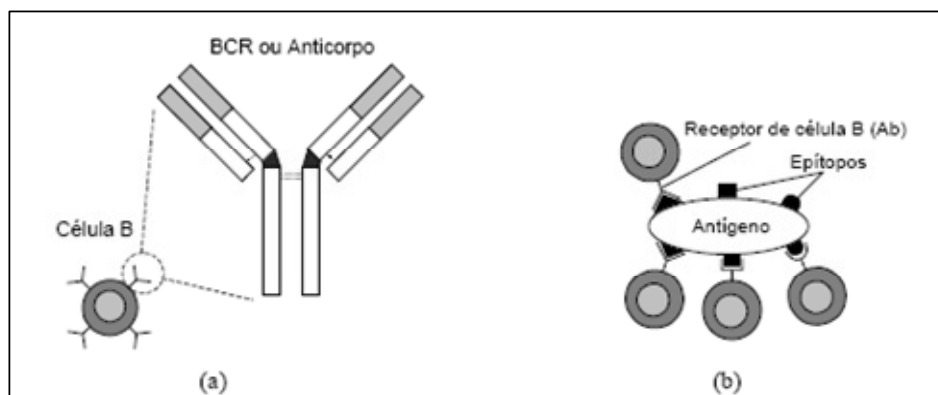


Figura 3.2– (a) A célula B possui todos os anticorpos iguais. (b) O antígeno possui diversidade de epítomos, reconhecidos por vários anticorpos (de Castro 2001).

Na fase de desenvolvimento do linfócito, ocorre a recombinação gênica que gera a diversidade inicial dos anticorpos. Logo após ativação da célula B

pelo contato do antígeno com um anticorpo específico, são gerados novos indivíduos, ou clones. Durante este processo de proliferação, ocorre a hipermutação somática, efetuando trocas nucleotídicas. O resultado final é um crescimento na capacidade de detectar mais antígenos e no refinamento da detecção de outros previamente encontrados. Para se ter uma idéia da ordem de grandeza, na ausência de estímulo antigênico, o corpo humano é capaz de produzir um trilhão de anticorpos diferentes. Caso o indivíduo não possuísse estes mecanismos de produção de diversidade, ele seria incapaz de se preparar contra a enorme diversidade de antígenos existentes (Jeneway, et al. 2005).

A detecção de um antígeno por um anticorpo não necessita ser exata, mas sim próxima o suficiente para gerar a ativação, o que insere uma grande capacidade de generalização no sistema, aumentando assim o potencial de detecção de padrões dos antígenos pelos anticorpos, ou seja, um anticorpo pode detectar mais de um antígeno.

3.2.3.2. O Princípio da Seleção Clonal

Como descrito anteriormente, o conjunto de linfócitos presentes em um indivíduo já tem uma considerável diversidade, presente desde a sua geração. Para que a ação dos anticorpos seja efetiva contra antígenos, como no caso de uma agressão infecciosa, por exemplo, eles devem ser específicos em relação à ameaça, aumentando assim as chances de ativação e a consequente eliminação dos antígenos.

O princípio da seleção clonal postula que a proliferação das células será atingida somente por aquelas capazes de reconhecer um estímulo antigênico, sendo assim selecionados para reprodução. Além do reconhecimento antigênico, as células T devem liberar substâncias chamadas citocinas para que os linfócitos B possam se proliferar e diferenciar. Após este segundo evento, as células B se multiplicarão e gerarão clones e estes clones se diferenciarão em plasmócitos (células efectoras ou terminais que secretam anticorpos em altas taxas) e células de memória (Figura 3.3). A diversidade advém justamente da recombinação dos segmentos gênicos e da hipermutação somática. Os plasmócitos são células com morfologia diferente dos linfócitos e podem secretar milhares de anticorpos por minuto durante vários dias (Encyclopaedia Britannica s.d.).

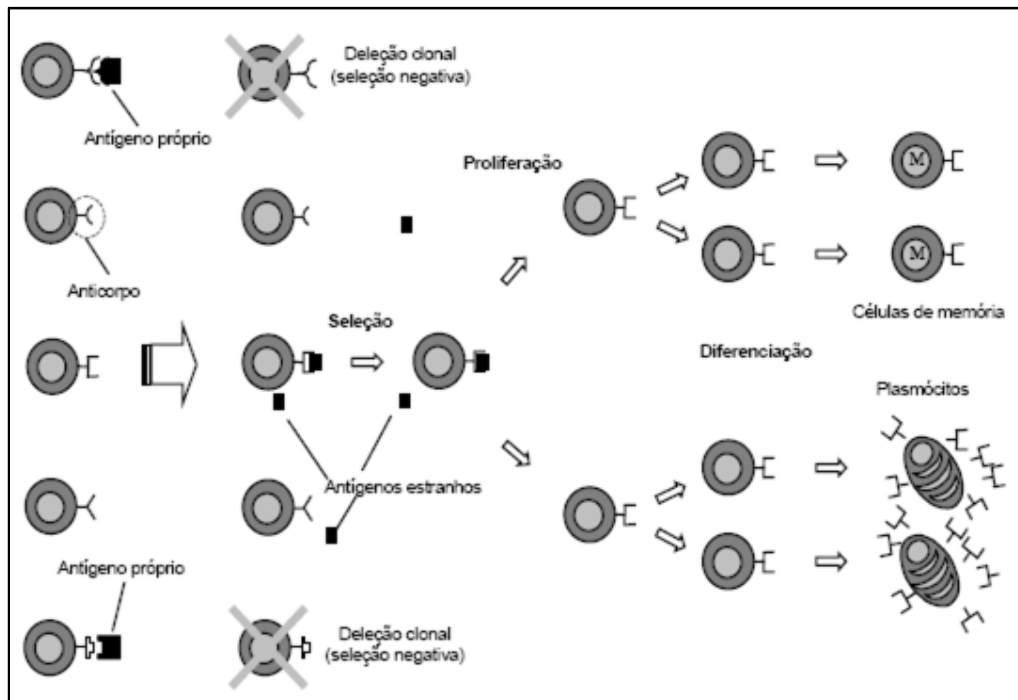


Figura 3.3 – Seleção clonal mostrando os possíveis desdobramentos da apresentação do antígeno ao anticorpo. Ao final ocorre a proliferação e diferenciação em células de memória e plasmócitos, que secretam anticorpos em grandes quantidades (de Castro 2001).

É interessante colocar que, no processo de geração de diversidade, existe a possibilidade da geração de resultados indesejados, nos quais os anticorpos reconheçam o próprio, ou seja, partes do indivíduo que não deveriam ser reconhecidas como um antígeno, iniciando-se um ataque às próprias células do indivíduo. Normalmente as células geradoras destes anticorpos são eliminadas, mas podem escapar deste processo e causar doenças, como artrite reumatóide, diabetes e lúpus. O nome do conjunto destas doenças causadas pelo sistema imune é doenças *auto-imunes*.

3.2.3.3. Processo Seletivo

A capacidade de reconhecimento dos anticorpos é tida como praticamente ilimitada, devido às diversas combinações possíveis na geração de receptores a partir de um número limitado de genes. Com esta imensa capacidade de reconhecimento, até mesmo as moléculas do próprio indivíduo podem ser detectadas como antígenos, o que leva à necessidade de distinguir entre o próprio, moléculas do indivíduo e o não próprio, os agentes externos. A partir desta diferenciação entre o próprio e o não próprio, torna-se relevante conhecer os principais desdobramentos dos encontros entre os antígenos e os receptores dos linfócitos:

- Expansão clonal: a proliferação das células será atingida somente por aquelas capazes de reconhecer um estímulo antigênico;
- Seleção positiva: trata da seleção das células B e T cujos receptores possam reconhecer os MHCs a partir das APCs próprias e também do

reconhecimento dos antígenos externos pelas células B. Estas condições dão início ao já apresentado princípio da expansão clonal;

- Seleção negativa: lida com a regulação das reações entre os receptores das células T ou B que se ligam ao próprio. Como as células B e T são geradas nos órgão linfóides primários e estes são projetados para manter somente antígenos próprios, a ativação de uma célula T ou B a estes tipos de antígenos sinaliza uma reação indevida. Geralmente as consequências desta ativação nas células T são a eliminação ou inativação (anergia) e para as células B são a eliminação, inativação ou recombinação de genes para aumentar a especificidade. A recombinação dos genes de um linfócito B para evitar seu descarte, ou inativação, faz parte de uma atuação mais ampla do sistema imune, chamada de maturação de afinidade. Na maturação de afinidade, não somente as células mais aptas terão a sua especificidade refinada e serão selecionadas para atuar como células de memória, mas algumas células que reagem ao próprio podem ter seus genes auto-reativos removidos e substituídos por outros resultantes de uma nova combinação de genes. Desta maneira, estas células aumentam as suas chances de gerar anticorpos mais específicos a antígenos externos e evitam o descarte excessivo de células improdutivas sob o ponto de vista imunológico;
- Ignorância Clonal: representa a não reação de um linfócito ao estímulo antigênico, podendo ser causada pela baixa concentração do antígeno ou fraca afinidade entre o receptor e o antígeno.

3.2.3.4. Memória e Aprendizado

A primeira reação do sistema imune a um antígeno é chamada de resposta primária. Considerando que o sistema imune de um indivíduo nunca foi exposto a um estímulo antigênico específico, não deve existir uma grande diversidade e quantidade de linfócitos B com anticorpos característicos para lidar com a ameaça. Com a presença do antígeno, ocorre o processo da seleção clonal, quando acontece a proliferação dos linfócitos ativados pelo antígeno e a sua posterior diferenciação em dois tipos. O primeiro tipo, os *plasmócitos*, que secretam grandes quantidades de anticorpos, e o segundo, as *células de memória*, que ficarão armazenadas no organismo do indivíduo esperando o estímulo apropriado para agir. Ao longo deste processo ocorre uma grande geração de anticorpos específicos para o antígeno, uma consequente diminuição da concentração de antígenos, devido à sua eliminação, e a manutenção de células de memória. As células de memória são células de alta especificidade e que podem sobreviver em estado de latência por anos até encontrar o estímulo adequado, quando se diferenciam em plasmócitos e passam a secretar anticorpos novamente (Calich e Vaz 2001).

As células de memória são fundamentais no processo de combate aos antígenos, pois elas fornecem uma resposta imunológica mais rápida e eficiente a ameaças semelhantes ocorridas no passado. A sucessão destas ameaças é bem provável ao longo da vida de um indivíduo, pois ele pode ser

exposto aos mesmos agentes infecciosos, ou similares, existentes no seu meio ambiente.

A resposta das células de memória a um antígeno semelhante, já previamente encontrado, gera uma grande produção de anticorpos em um tempo muito menor, pois as células de memória com anticorpos específicos estão presentes no organismo em maior quantidade, se comparadas aos linfócitos “inocentes” ou “virgens” que deram origem à resposta primária. Uma consequência deste comportamento é que o tempo de reação é sensivelmente menor em relação à resposta primária, o que vem a ser um fator relevante para a integridade do indivíduo em um cenário de ameaças constantes. A esta reação das células de memória dá-se o nome de resposta secundária.

As respostas primárias e secundárias revelam um comportamento extremamente importante do sistema imune, que está frequentemente aprendendo e memorizando suas reações através da interação constante com seu meio ambiente (de Castro e Timmis 2002). Seguem situações relevantes que fomentam o aprendizado e a memória do sistema imune (de Castro 2001):

- Aprendizado:
 - ✓ Contato contínuo com antígenos;
 - ✓ Crescimento da população de clones de maior afinidade;
 - ✓ Refinamento da especificidade através da maturação por afinidade.
- Memória:
 - ✓ Presença de células de memória com alta especificidade e longos períodos de vida que podem ser ativadas e responder prontamente ao estímulo antigênico;
 - ✓ Reconhecimento de antígenos similares e não necessariamente iguais em relação ao anticorpo;
 - ✓ Contato constante com antígenos.

3.2.3.5. Teoria da Rede Imunológica

O princípio da seleção clonal descreve, em linhas gerais, os mecanismos do sistema imune adaptativo que atuam, quando este sistema é estimulado e considera que os clones do sistema imune estão em repouso, caso nenhum estímulo antigênico externo se apresente.

Com uma abordagem diferente em relação ao princípio da seleção clonal e uma proposta interessante sob o ponto de vista da tolerância natural (Coutinho 2003), o paradigma da rede imunológica renasce, após um longo período de dominação no cenário imunológico e posterior ostracismo, devido ao sucesso da imunobiologia molecular (Behn 2007).

A teoria da rede imunológica proposta por Jerne (1974) oferece uma abordagem linfocitária diferente em relação ao princípio da seleção clonal. Na

teoria da rede imunológica Jerne sugere que “o sistema imune é composto por uma rede regulada de células e moléculas que se reconhecem mesmo na ausência de antígenos” (de Castro 2001). Como já se sabe, o processo de geração da diversidade de anticorpos é aleatório, podendo-se esperar por anticorpos que reajam tanto ao próprio como ao não próprio. Indo mais além, os anticorpos podem também ser estimulados por outros anticorpos, sendo estes tratados como antígenos por aqueles que os detectam. A partir desta possibilidade, uma nova nomenclatura foi cunhada, nomeando-se a região de ligação do anticorpo de *paratopo* e a parcela detectável do antígeno de *epítopo*. Estudos confirmaram que os anticorpos possuem epítomos, como se tivessem um componente antigênico que permite a sua detecção por anticorpos específicos. O nome de *idiotopo* foi dado a um epítopo de anticorpo e *idiotipo* ao conjunto de epítomos de anticorpo presentes na região-V.

Como consequência deste modelo, foi proposta outra definição do sistema imune, na qual o sistema imune é

constituído por uma rede enorme e complexa de paratopos que reconhecem conjuntos de idiotopos, e de idiotopos que são reconhecidos por conjuntos de paratopos, assim, cada elemento pode reconhecer e ser reconhecido (Jerne 1974).

A partir do estímulo entre o idiotopo e o epítopo, uma resposta positiva ou negativa pode ser formada. A resposta positiva configura a ativação, proliferação e secreção de anticorpos, enquanto a resposta negativa seria a condução à supressão (Figura 3.4).

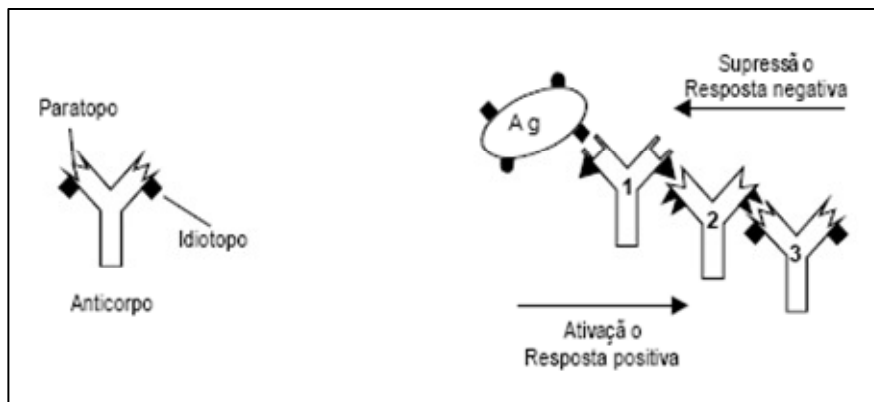


Figura 3.4 – Respostas entre anticorpo-antígeno e anticorpo-anticorpo da rede imunológica (de Castro 2001).

Pela teoria da rede imunológica, os anticorpos podem reconhecer e serem reconhecidos na ausência de estímulos antigênicos externos, sendo assim o sistema imune apresenta certo comportamento cognitivo derivado desta interação. Este comportamento leva a um equilíbrio dinâmico entre as partes, aumentando e diminuindo a concentração e diversidade de seus elementos à medida que os estímulos acontecem. Segundo de Castro (2001):

a teoria de rede é particularmente interessante para o desenvolvimento de ferramentas computacionais, pois ela fornece uma medida aproximada de propriedades emergentes como aprendizagem e memória, tolerância ao próprio, tamanho e diversidade de populações celulares.

3.2.3.6. Teoria do Perigo

Tradicionalmente a imunologia trata o reconhecimento dos antígenos através do conhecimento do que é próprio e do que não é próprio, ou seja, o que é nativo ao indivíduo e o que é externo, agindo para eliminar o que é não próprio, considerado ameaçador. Este princípio está em uso há aproximadamente 50 anos e tem servido para explicar razoavelmente bem diversos comportamentos imunológicos, mas existem várias situações que não possuem uma boa explicação neste modelo, assim como também não é bem conhecida a diferenciação do próprio e do não próprio pelo indivíduo, mesmo considerando o processo da seleção negativa (Matzinger 1994) (Matzinger 2002).

Dentre as questões que suscitaram uma explicação mais consistente do modelo vigente estão (Aickelin e Cayzer 2002):

- Por que as mães não rejeitam os seus fetos, já que são novos indivíduos? Por que os fetos não são atacados e os órgãos transplantados geralmente são atacados pelo sistema imune?
- Por que não há uma reação imunológica para com as bactérias do nosso intestino?
- O que acontece quando o próprio muda, como resultado da puberdade, gravidez e envelhecimento? Por que tecidos alterados não são atacados?
- Por que algumas células ou órgãos são atacados mesmo sendo originados do próprio indivíduo?

As respostas a estas perguntas podem ser obtidas através de um modelo mais abrangente, que se apóia no conceito do perigo, ou seja, o não próprio pode ou não representar uma ameaça à integridade do indivíduo. Através desta abordagem, as questões acima colocadas podem ter uma resposta mais aceitável, pois extrapolam o conceito do próprio e não próprio através de um senso real de perigo, o que leva à necessidade de compreender melhor a percepção do que representa e não representa perigo (Matzinger 1994).

A teoria do perigo postula, com base em diversas observações, que nem tudo o que é externo é ameaçador. O perigo é caracterizado pelos danos às células, percebidos por sinais emitidos por estas células quando sofrem uma morte não natural (Matzinger 2002). Quando estas células enviam os sinais de auxílio, as APCs representadas por macrófagos percebem estes sinais, capturam os antígenos e os apresentam aos linfócitos, que por sua vez se reproduzem e atacam os antígenos para os quais possuem especificidade. Os sinais de auxílio são emitidos e enviados no entorno onde as células estão

sendo atacadas, e assim como alguns linfócitos reagem ao não próprio, outros podem ser ativados pelo próprio e iniciar uma resposta auto-imune. Com o passar do tempo, estas células auto-reativas são eliminadas, pois segundo a teoria do perigo, foram ativadas sem um sinal co-estimulatório, necessário para ativação, colocando-se então como células imaturas e passíveis de eliminação por reagirem ao próprio. Ao final deste processo a tolerância ao próprio é restabelecida (Aickelin e Cayzer 2002).

Apesar de inovadora, a teoria do perigo apresenta lacunas a serem preenchidas em alguns pontos, especificamente em como se dão as percepções das reais situações de perigo, se são positivas (liberação de proteínas e calor) ou negativas (ausência de contato sináptico com uma célula dendrítica); as situações nas quais o perigo não deve ser detectado (transplantes) e como conciliar o modelo com as doenças auto-imunes (Aickelin e Cayzer 2002).

É importante ressaltar que a teoria do perigo é considerada controversa e não totalmente aceita pela comunidade científica, mas constitui uma relevante inspiração para a construção de sistemas imunes artificiais devido à sua abordagem alternativa e pragmática na caracterização das ameaças (Aickelin e Cayzer 2002).

3.3. Sistema Imune Artificial

A natureza é fonte inspiradora de diversos paradigmas computacionais para resolução de problemas na área de inteligência artificial (IA). Dentre as fontes de inspiração mais conhecidas estão o funcionamento do cérebro humano, que originou as redes neurais artificiais e os processos evolutivos que originaram a computação evolutiva. A lógica fuzzy, ou nebulosa, também figura como um importante paradigma, mas possui inspiração na teoria dos conjuntos e é bastante utilizada no tratamento de grandezas inexatas.

O sistema imune biológico contém características que tratam as complexas e constantes situações de ameaças a que um indivíduo está sujeito. Dentre as suas características, as mais marcantes são a detecção de padrões, geração de memória, capacidade de aprendizado, diversidade de indivíduos, descentralização e resiliência, atributos estes primordiais para que o sistema imune biológico mantenha o seu equilíbrio e consiga operar as funções essenciais à manutenção da vida. Sendo assim, um Sistema Imune Artificial, ou SIA, representa basicamente a transposição de aspectos relevantes do modelo imunológico biológico, objetivando adaptá-los e utilizá-los na resolução de problemas tratáveis computacionalmente que demandem abordagens com características deste tipo.

Os diversos modelos iniciais usados para explicar o comportamento imunológico eram praticamente desprovidos de tratamento matemático. A inserção da matemática forneceu importante apoio à imunologia, como a melhora na descrição quantitativa do sistema imune, apoio a previsões e planejamento de experimentos e análise de hipóteses e resultados (de Castro e Timmis 2002). A consideração da matemática facilita a modelagem de certos

princípios ou teorias imunológicas no SIA, já que a expressão computacional é essencialmente matemática. Torna-se então importante definir SIA a partir deste contexto, representado aqui por dois exemplos. No primeiro, “os sistemas imunes artificiais são compostos por metodologias inteligentes, inspiradas no sistema imune biológico, para a solução de problemas de mundo real” (Dasgupta 1999) e no segundo é “um Sistema Imune Artificial é um sistema computacional baseado em metáforas do sistema imune natural” (Timmis 2000). Outra definição interessante atesta que:

os sistemas imunes artificiais (SIA) são sistemas adaptativos inspirados na imunologia teórica e nas observações das funções imunes, com seus princípios e modelos para aplicação na resolução de problemas (de Castro e Timmis 2002).

Nesta última definição, o termo imunologia teórica refere-se a todos os fundamentos matemáticos e não matemáticos, teorias e princípios que descrevem o sistema imune.

Com um conjunto relevante de propriedades e processos, o SIA tem sido fruto de estudo há aproximadamente vinte anos desde os seus trabalhos precursores (Farmer, Packard e Perelson 1986) (Hoffmann 1986) (Hoffmann, Benson, et al. 1986), mas foi somente em 1996 que tomou forma como uma linha de pesquisa, quando foi realizado no Japão o primeiro workshop sobre sistemas baseados em imunologia. A partir desta data houve um aumento significativo de publicações, livros, palestras e participação em workshops e congressos. SIA conta atualmente com um congresso internacional, o ICARIS. Para maiores informações sobre o histórico dos principais eventos de SIA, consultar (de Castro 2001).

É interessante ressaltar que os trabalhos nesse período não chegaram a realizar uma unificação da formalização dos conceitos e ferramentas de SIA. A partir desta percepção de necessidade, criou-se a Engenharia Imunológica (EI), um processo de meta-síntese na definição de uma ferramenta baseada no modelo imune biológico, com objetivo de resolver problemas computacionais através dos atributos essenciais deste modelo (de Castro 2001). Baseadas na EI serão fornecidas representações dos componentes de um sistema, os mecanismos de interação entre seus elementos constituintes e os procedimentos que regulam a sua dinâmica.

3.3.1. O Espaço de Formas

Para que haja estímulo do anticorpo com um antígeno, estes devem ter um grau mínimo de acoplamento, ou afinidade, como peças de um quebra-cabeça com algumas folgas. A relação entre o receptor, ou paratopo e o epítopo (ou idiotopo) não exige um “encaixe” perfeito nas regiões complementares (Figura 3.5a), mas aproximado o suficiente para que o estímulo antigênico possa ocorrer. A interação entre estes pares é, na verdade, baseada em grandezas como cargas eletrostáticas, ligações de hidrogênio e interações de van der Waals, além de outras. Às diversas variações destas grandezas dá-se o nome de forma generalizada de uma molécula. Esta forma generalizada pode ser

expressa em um conjunto L de parâmetros, como largura, carga, profundidade, etc, de modo que representem um ponto em um espaço S de formas multidimensional e sejam passíveis de tratamento computacional. Como geralmente existe uma faixa de valores que podem ser assumidos pelos atributos formadores dos antígenos e anticorpos em uma relação complementar, uma região V (Figura 3.5b) será delimitada por estes valores, onde serão analisadas as afinidades entre as partes. Partindo-se da idéia da ativação pela afinidade de relações complementares e que esta afinidade não necessita ser exata, mas aproximada, é pertinente que exista uma sub-região do espaço V em torno de um determinado anticorpo na qual os antígenos presentes nesta região sejam reconhecidos por este anticorpo. A esta região próxima do anticorpo e delimitada por um limiar ε , dá-se o nome de região de reconhecimento, ou V_ε .

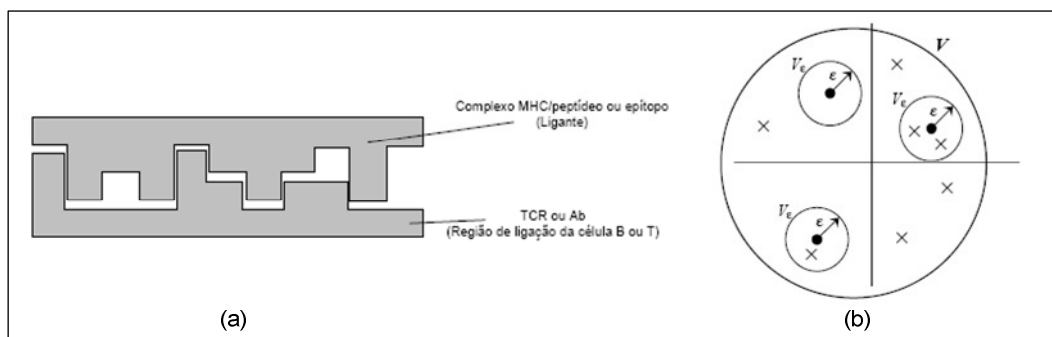


Figura 3.5 – (a) Reconhecimento por complementaridade. (b) Espaço de formas S com região V onde um paratopo (ponto) pode reconhecer um epítipo (cruz) (de Castro 2001).

3.3.2. Representações e Medidas de Afinidade

A maneira de representar os anticorpos e antígenos no espaço de formas como cadeias de atributos está fortemente vinculada ao tipo e domínio do problema a ser solucionado por SIA. Os espaços de formas mais comuns derivados dos tipos de atributos são:

- Hamming: originalmente é uma representação da distância entre cadeias de símbolos, sendo estes pertencentes a um conjunto finito de elementos e que a diferença, ou distância, entre as cadeias corresponde à quantidade de atributos diferentes entre as cadeias. Hamming é muito aplicada para tratar valores binários e também é a representação mais utilizada nas aplicações de SIA e redes imunológicas biológicas;
- Inteiro: os atributos são formados por valores inteiros. As diferenças entre valores inteiros podem ser expressas também sob a forma da distância de Hamming e até mesmo Manhattan, dependendo da aplicação;
- Real: é usada para expressar valores reais. A medida de distância mais comum utilizada para expressar a diferença é a euclidiana, mas a Manhattan pode ser usada, dependendo da aplicação.

A partir das representações como uma cadeia de anticorpo $Ab = (Ab_1, Ab_2, Ab_3, \dots, Ab_L)$ e de antígeno $Ag = (Ag_1, Ag_2, Ag_3, \dots, Ag_L)$ e da necessidade de extrair-se a afinidade como medida quantitativa do reconhecimento, é importante ter uma visão pragmática das representações computacionais dos linfócitos e da medida de afinidade.

- Representação das células: a maioria das células do sistema imune representadas no SIA são as células B, pois são elas que secretam os anticorpos que se ligarão aos antígenos e diversas aplicações não necessitam de modelagens de maior complexidade que esta. A representação comum destas células é expressa como uma cadeia de atributos, desconsiderando-se o restante, como as suas características de APC e seu idiotipo. Em estruturas com mecanismos mais elaborados, como as redes imunológicas, a representação das células B pode ir além da expressão do seu receptor. Existem aplicações, que consideram também a representação de outros agentes do sistema imune, como a célula T, que regula a resposta imunológica das células B e também tem ação supressora. Para maiores detalhes sobre as células T e suas aplicações em SIA, consultar (Guzella, Mota-Santos e Caminhas 2007);
- Complemento e similaridade: a afinidade pode ser representada basicamente de duas maneiras, pelo complemento e pela similaridade. O complemento é uma maneira de calcular a afinidade, e se dá na medida em que, quanto menor a distância entre a cadeia de atributos do anticorpo e o complemento da cadeia de atributos do antígeno, maior será a afinidade. Outra maneira de se tratar a afinidade é através da similaridade, ou seja, as cadeias de atributos que apresentem as distâncias mais próximas, sem envolver o complemento, têm a maior afinidade. A escolha de um dos dois métodos é baseada na aplicação;
- Limiar e proximidade: a afinidade traduz o reconhecimento entre as cadeias de atributos dos anticorpos e antígenos com base no objetivo estabelecido, seja ele de semelhança ou diferença. Como muitas vezes a afinidade é um valor numérico, a relação Ag-Ab que atende ao critério definido para ativação pode estar restrita a um intervalo ou limiar de afinidades, como também pode ser expressa pelos antígenos mais próximos do anticorpo, sem considerar um determinado intervalo ou limiar. Mais uma vez, a aplicação é que determinará a melhor maneira de se trabalhar com a afinidade.

As Equações (2.1), (2.2) e (2.4) representam as medidas usadas para calcular a distância Euclidiana, Manhattan e Hamming, respectivamente. A Figura mostra exemplos das medidas no espaço de formas de Hamming e algumas de suas variações.

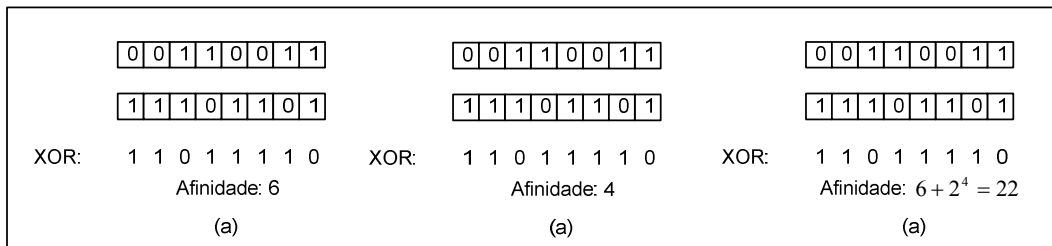


Figura 3.6 – Variações de medidas no espaço de formas de Hamming. (a) Quantidade total de bits complementares. (b) Quantidade de r -bits complementares consecutivos. (c) Medida de Hunt, Equação (2.8) (de Castro 2001).

3.3.3. Processos Imunes e Seus Algoritmos

Após a apresentação da representação e tratamento matemático da afinidade entre os anticorpos e antígenos, é necessário conjugar estes elementos para que possam modelar os processos imunes mais importantes de um SIA.

Os processos imunológicos aqui apresentados são os da medula óssea, seleção negativa, seleção clonal e a rede imunológica.

3.3.3.1. Medula Óssea

A medula óssea gera um conjunto de anticorpos que serão utilizados na detecção dos antígenos no organismo do indivíduo. Existem diversas maneiras de se gerar estes anticorpos através do SIA e a definição de como gerá-los é tomada com base no problema a ser resolvido. O modo mais simples é gerar os atributos da cadeia de comprimento L de cada anticorpo de maneira pseudo-aleatória. Se o espaço de formas for um espaço binário, então os valores dos atributos ficarão contidos em $\{0,1\}$ e caso seja um espaço real, os valores devem estar contidos em $[0,1]^L$. No caso de valores inteiros, uma permutação de pontos pode ser suficiente, como vértices de um grafo que ligados após este processo geram diferentes caminhos. A seguir está representado um algoritmo genérico da medula óssea:

Algoritmo que gera anticorpos aleatórios

1. Definir número N de anticorpos aleatórios a serem gerados;
2. Definir tipo e número de atributos L dos anticorpos;
3. Gerar um valor pseudo-aleatório para cada atributo de cada anticorpo;
4. Retornar o conjunto de anticorpos gerados;

Algoritmo 3.1 – Gera anticorpos de comprimento L .

3.3.3.2. Seleção Negativa

A seleção negativa é o processo responsável por remover os linfócitos T e B que atacam as próprias células do organismo, diminuindo assim a probabilidade de ocorrências de doenças auto-imunes. O algoritmo apresentado a seguir é uma versão mais geral da seleção negativa proposta de Forrest *et. al*, podendo ser aplicada a qualquer tipo de dados e representação

de afinidade. Originalmente ele foi concebido para ser aplicado na área de segurança de computadores, especificamente na detecção de vírus de computadores (Forrest, et al. 1994). A seguir está representado um algoritmo de seleção negativa:

Algoritmo que realiza a seleção negativa

1. Gerar anticorpos aleatórios - Algoritmo 3.1;
2. Avaliar a afinidade de cada anticorpo com todos os elementos próprios apresentados;
3. Eliminar o anticorpo ativado do repertório, caso ocorra a ativação deste com o próprio.

Algoritmo 3.2 – Seleção negativa para eliminar anticorpos.

Este algoritmo apresenta uma limitação para aplicação em ambiente de grande escala, pois devido ao grande número de antígenos e anticorpos, fica cada vez mais difícil encontrar antígenos e anticorpos compatíveis e manter a eficiência do processo (Kim e Bentley 2001).

Em Zhou & Dasgupta (2007) é feita uma análise mais aprofundada das variações de algoritmos de seleção negativa, levando em consideração a representação dos dados, representação dos detectores, medidas de afinidade, o mecanismo de geração de detectores e a sua eliminação, objetivando apresentar um panorama mais amplo que possibilite melhores escolhas na criação de algoritmos de seleção negativa.

Em Ayara *et. al* (2002) os autores fazem comparações entre o algoritmo tradicional de seleção negativa e outras variações para mostrar que a escolha do algoritmo de seleção negativa envolve considerações de complexidade espacial.

3.3.3.3. Seleção Clonal

A seleção clonal é um processo que seleciona as células B e T com maiores afinidades aos antígenos apresentados. Através do processo de proliferação e reprodução, as células B sofrem mutação em altas taxas (hipermutação somática) que, conjugadas com a pressão seletiva, geram anticorpos mais específicos no combate dos antígenos. Um aspecto interessante da seleção clonal utilizado em SIA e que, ao mesmo tempo a diferencia em relação a algoritmos evolucionários é a reprodução proporcional à afinidade, na qual as células cujos anticorpos possuem maior afinidade geram uma quantidade maior de descendentes. Outra característica é que células cujas afinidades são baixas sofrem mutação a uma taxa maior em relação às células de maior afinidade, tendendo a gerar células com anticorpos mais afins e preservar aquelas com melhor afinidade. O algoritmo que expressa estas características é o CLONALG (de Castro e Von Zuben 2000) e está apresentado a seguir:

1. Gerar anticorpos aleatórios para criar uma população **P** – Algoritmo 3.1;
2. **Enquanto** critério de parada não atingido (geralmente é o número de gerações da população de anticorpos ou a precisão da afinidade)
 - 2.1. **Para** cada antígeno
 - 2.1.1. Verificar a afinidade do antígeno em relação a cada anticorpo da população **P**;
 - 2.1.2. Fazer a geração proporcional de clones para os n_1 anticorpos, da população **P** que apresentam maior afinidade em relação ao antígeno apresentado, ou seja, quanto maior a afinidade, maior o número de clones;
 - 2.1.3. Realizar a mutação de cada um dos clones de modo inversamente proporcional à sua afinidade, ou seja, quanto maior a afinidade entre o clone e o antígeno, menor deve ser a taxa de mutação. Após a realização das mutações, estes elementos devem ser reinseridos na população **P**. O mais apto desta população deve ser selecionado como célula de memória **m** para o antígeno apresentado;
 - 2.1.4. Substituir os n_2 anticorpos menos aptos da população pelos n_1 gerados no processo de mutação;
 - 2.2. **Fim Para**
3. **Fim Enquanto**

Algoritmo 3.3 – Seleção de anticorpos mais aptos aos antígenos.

O CLONALG é um algoritmo baseado em aprendizado não supervisionado utilizado para classificação de dados, onde os antígenos ou padrões de dados são apresentados aos anticorpos. Como uma alternativa de aprendizado supervisionado, o algoritmo Sistema de Reconhecimento Imune Artificial (*Artificial Immune Recognition System* - AIRS) foi proposto (Watkins, Timmis e Boggess 2004) e se diferencia basicamente em relação ao CLONALG por trabalhar com o conceito de Bolas de Reconhecimento Artificial (*Artificial Recognition Balls* - ARBs), que tem a função de representar células idênticas, de maneira única com objetivo de evitar a duplicação e limitar o crescimento da população através de um mecanismo de distribuição de recursos finitos. A mutação também é operada de modo diferenciado, onde esta não ocorre de maneira inversamente proporcional à afinidade, mas sim de maneira probabilística, independente da afinidade.

3.3.3.4. Mutação

A mutação é o processo gerador da diversidade na população de anticorpos e este pode ser executado através de diversos mecanismos de seleção e da mutação propriamente dita (de Castro e Timmis 2002). Dependendo dos tipos de dados e das restrições da aplicação, diferentes métodos de seleção e mutação podem ser empregados. Para ilustrar a mutação, serão aqui exemplificados processos baseados no espaço de Hamming, números inteiros e reais.

Na Figura 3.7a, o espaço de Hamming contém um anticorpo representado por uma cadeia binária e o atributo número cinco sofre mutação. Além da mutação em um único atributo, o anticorpo também pode sofrer mutação em múltiplos atributos simultaneamente, como ilustrado na figura Figura 3.7b. Os atributos inteiros também podem sofrer mutação da mesma maneira que os binários anteriormente apresentados e serem representados no espaço de formas de Hamming, lembrando que é importante a existência de conjunto que explicita quais valores inteiros, ou intervalo, podem ser assumidos pelos atributos. Além da mutação direta, os atributos inteiros podem sofrer diferentes tipos de mutação, como a mutação inversa, que troca valores de atributos de posição, sem acrescentar nenhum novo valor na cadeia de atributos. Esta situação pode ocorrer devido a restrições do problema a ser resolvido, que exige a permanência dos mesmos atributos na cadeia de atributos.

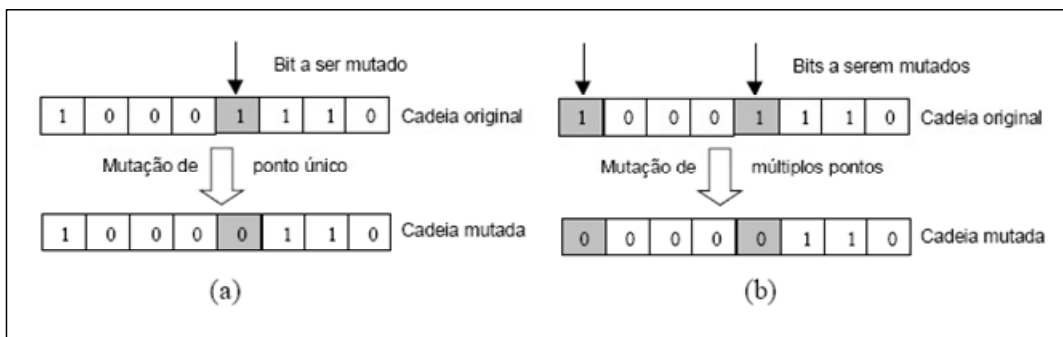


Figura 3.7 – (a) Mutação de cadeia binária em ponto único. (b) Mutação de cadeia binária pontos múltiplos (de Castro 2001).

No caso dos atributos reais, os mecanismos de mutação supracitados também podem ser empregados, atendo-se aos limites inferiores e superiores de cada atributo e à afinidade proporcional, onde o anticorpo com a maior afinidade terá uma baixa ou nula mutação para que a sua especificidade não seja perdida. Além das técnicas de substituição e troca, pode ser utilizada a mutação indutiva, consistindo da obtenção de um novo valor do atributo através de uma pequena alteração no seu valor atual (de Castro e Timmis 2002). Assim,

$$m' = m + \alpha(D)N(0, \sigma) \quad (3.1)$$

Onde $m = (m_1, m_2, m_3, \dots, m_L)$ é uma cadeia de atributos de um anticorpo que sofrerá mutação, $\alpha(D)$ é uma função que representa um fator de mutação inversamente proporcional à afinidade e $N(0, \sigma)$ é um vetor de variáveis Gaussianas aleatórias independentes com média zero e desvio padrão σ a ser ajustado por α . Para que α seja calculado, deve-se obter o valor normalizado da afinidade D em relação a um D_{max} , expressado por

$$D^* = \frac{D}{D_{max}} \quad (3.2)$$

Com D^* calcula-se α , expresso por $\alpha(D^*) = \exp(-\rho D^*)$, onde ρ é um fator de amortização da função exponencial α e também maior que zero.

3.3.3.5. Rede Imunológica

A teoria de rede imunológica se baseia na premissa de que os anticorpos reconhecem e são reconhecidos pelos seus pares, assim como reconhecem os antígenos. Esta capacidade de reconhecimento mútuo dos anticorpos é explicada pela existência dos idiotopos, que são epítomos idiotípicos que podem ser reconhecidos por outros anticorpos. Através deste mecanismo estabelece-se um comportamento dinâmico pois, segundo esta teoria, o sistema imune não está em estado latente à espera de um estímulo externo, o próprio sistema imune passa a ser componente importante na sua estimulação.

Os modelos de rede imunológicas se dividem em contínuos e discretos (de Castro e Timmis 2002), onde os contínuos se baseiam em equações diferenciais ordinárias e foram os primeiros a serem propostos. Nestes modelos, a variação da população se dá de maneira genérica pela expressão (Perelson 1989):

$$\text{Taxa de variação da população} = \text{Estímulo da rede} - \text{Supressão da rede} + \text{Entrada de novos elementos} - \text{Morte de elementos não estimulados} \quad (3.3)$$

Dentre os modelos contínuos, os mais relevantes são os de Jerne (1974), Farmer *et. al* (1986) (1987) e Varela & Coutinho (1991). Este trabalho focará no modelo discreto, especificamente o modelo de de Castro & Von Zuben (2000), a Rede Imune Artificial (*Artificial Immune NETwork* - aiNet). A escolha do modelo discreto se dá por algumas razões: primeiro, porque nem sempre é possível encontrar uma solução analítica para o problema; segundo, porque o modelo discreto trata a rede imunológica como constituída também pela relação entre antígenos e anticorpos, não exclusivamente entre anticorpos e, por último, o modelo discreto é mais permissivo quanto à diversidade e quantidade das células, contribuindo assim para aumentar a sua especificidade e afinidade. A seleção da rede discreta aiNet se deve à relevância de suas características, como a “capacidade de descrever a estrutura interna dos antígenos (dados de treinamento), seu perfil de distribuição de probabilidade, relações de vizinhança (grupos)” e compressão de dados, atributos com grande potencial de aplicação no Raciocínio Baseado em Casos (de Castro 2001).

A aiNet é inspirada no modelo de Jerne (1974) e foi concebida para uso com valores reais no espaço de formas Euclidiano, mas nada impede que sejam usados outros espaços de formas. A aiNet lida com a afinidade através da similaridade, avaliando o seu grau entre os anticorpos que constituem a rede e os antígenos no meio ambiente.

Na aiNet, cada elemento da rede é formado por uma molécula de anticorpo, ou seja, uma cadeia de atributos, gerados aleatoriamente, como já citado em outros processos imunes artificiais. Em seguida, os anticorpos são apresentados a um antígeno de cada vez, onde a afinidade será calculada pela distância euclidiana, como na Equação (2.1). Os anticorpos com maior valor de afinidade gerarão mais clones, que por sua vez passarão por um processo de

mutação, lembrando que a mutação se dá de maneira inversamente proporcional à afinidade. Os anticorpos com a maior afinidade são mantidos como células de memória e os de menor afinidade, ou abaixo de determinado limiar, são descartados, sendo substituídos por outros gerados no processo de expansão clonal e mutação. Após esta consolidação, os anticorpos formam uma rede, onde a sua afinidade em relação a outros anticorpos é calculada e os anticorpos com maior afinidade entre si (anticorpos muito semelhantes) são removidos da rede, promovendo assim a diversidade entre eles.

Algoritmo de aprendizado da aiNet

1. Gerar anticorpos aleatórios para criação de uma população – Algoritmo 3.1;
2. **Enquanto** critério de parada não atendido
 - 2.1. **Para** cada antígeno
 - 2.1.1. Determinar a afinidade de cada anticorpo ao antígeno apresentado e selecione os de maior afinidade para uma reprodução em quantidade proporcional à sua afinidade;
 - 2.1.2. Efetuar a mutação de cada clone de tal modo que seja inversamente proporcional à sua afinidade. Selecionar o número de clones com maior afinidade para fazer parte do repertório das células de memória;
 - 2.1.3. Eliminar os clones de memória cuja afinidade com o antígeno seja menor que um limiar específico, desta maneira, os clones de maior afinidade permanecerão no repertório;
 - 2.1.4. Calcular a afinidade entre todos os clones de memória;
 - 2.1.5. Eliminar os clones que tiverem grande afinidade entre si, assim anticorpos de memória muito semelhantes são eliminados;
 - 2.1.6. Inserir os clones de memória referentes ao antígeno apresentado junto aos demais clones da rede;
 - 2.2. **Fim Para.**
3. Determinar a afinidade entre todos os anticorpos da rede;
4. Eliminar os anticorpos cuja afinidade seja menor que determinado limiar;
5. Inserir novos anticorpos gerados aleatoriamente na rede;
6. **Fim Enquanto.**

Algoritmo 3.4 – Aprendizado da aiNet com a seleção de anticorpos a partir da afinidade entre antígenos e outros anticorpos.

Além da aiNet, o modelo de redes discretas conta com propostas diferentes em relação à criação da rede e seu crescimento. Uma destas propostas está ligada ao controle de recursos na rede e uso de ARBs, denominado Sistema Imune Artificial de Recurso Limitado (*Resource Limited Artificial Immune System - RLAIS*) (Timmis e Neal 2001). Este último trabalho é uma melhoria em relação ao trabalho realizado anteriormente (Timmis, Neal e Hunt 2000) que carecia de um controle populacional efetivo e apresentava problemas para criação de vínculos entre os elementos da rede imunológica. No RLAIS, os anticorpos iniciais não são gerados por um processo aleatório independente, como na aiNet, mas escolhidos a partir do repertório de antígenos. A partir deste ponto, todos os antígenos do repertório são apresentados aos ARBs e o estímulo de cada um é calculado. O objetivo do estímulo é mapear a distribuição dos recursos finitos na rede e com isto tomar

a decisão de quais ARBs poderão ter mais células armazenadas e quais serão eliminados. Os vínculos entre ARBs que formam a rede imunológica são criados a partir da proximidade entre os novos ARBs e seus vizinhos.

A partir dos modelos e propostas de redes apresentadas, torna-se importante a escolha da que melhor se adapta ao problema, dependendo dos objetivos, restrições e características dos dados e suas representações no espaço de formas. Deve-se ater em especial à escolha do espaço de formas e medidas de afinidade, pois o estudo realizado em (Hart, Bersini e Santos 2007) indica que diferentes redes surgem a partir de variações das suas representações. É interessante ressaltar que, dependendo de cada modelo utilizado, o grau de coesão da rede é consequência da criação de conexões e também da impossibilidade da criação destas conexões, devido a fatores como variação do limiar de ativação e concentração de antígenos e anticorpos.

3.4. Exemplos de Domínios de Aplicação

Desde o primeiro workshop dedicado ao SIA em 1996, foram abordados diversos de seus princípios e processos em problemas dos mais variados tipos, seja em simulações ou no mundo real.

As categorias das aplicações em SIA podem ser divididas de maneira geral em três grandes áreas: aprendizado, que inclui agrupamento, classificação, reconhecimento de padrões, robótica e sistemas de controle; detecção de anomalias, que engloba detecção de falhas e segurança de redes e computadores e, por último, otimização, que representa problemas combinatórios e otimização de funções numéricas. Partindo destas três divisões, serão apresentadas algumas aplicações de cada uma delas (Hart e Timmis 2008).

3.4.1. Aprendizado

3.4.1.1. SPAM

O SPAM, ou e-mail indesejado, é um problema que recebe muita atenção da comunidade científica e empresarial, pois é responsável por gerar improdutividade de usuários e desperdícios de recursos computacionais que poderiam ser utilizados para fins legítimos. O SPAM é um problema em franca mudança de comportamento, pois necessita alterar constantemente suas técnicas de divulgação para que os algoritmos e os sistemas existentes não sejam capazes de detectar as mensagens indesejadas. Para tratar este problema, uma abordagem imunológica muito interessante faz uso do sistema imune inato e adaptativo, proposto por Guzella (2008). Nesta proposta, as mensagens são representadas como microorganismos, que estes são divididos em três partes, o título e conteúdo, considerados antígenos e o remetente, que é o padrão molecular, ou MHC. A partir deste formato, os componentes do sistema inato e adaptativo são modelados, criando assim um sistema de proteção em duas camadas. Os e-mails são representados como vetores de palavras e, por sua vez, cada palavra é formada por uma sequência de

caracteres com codificação de seis bits, de tal modo que a detecção de palavras semelhantes, como *sex* e *zex*, ou *free* e *fr33*, seja facilitada através de uma medida de afinidade que avalie seqüências de bits.

Fazendo a analogia com o modelo imunológico biológico, os macrófagos, representando o sistema inato, se ligam ao padrão molecular do microorganismo, reconhecendo assim um conjunto fixo de elementos. Por sua vez as células T auxiliares, reguladoras e as células B se ligam às partes do microorganismo compostas pelo antígeno, criando a capacidade de reconhecimento adaptativo. As células T e B sofrem seleção negativa e aquelas que se ligam ao próprio, representados pelos e-mails legítimos do conjunto de treinamento, são eliminadas. Nesta proposta, a estimulação de uma célula B por um antígeno passa ao estágio de ativação somente se o número de vezes que as células T auxiliares ativadas for maior que o número de células T reguladoras ativadas. Este mecanismo é interessante, pois a ativação se dá somente após a constatação pelo sistema que a mensagem é considerada SPAM e também não é detectada como legítima. Neste classificador, o seu desempenho é melhorado por um processo de mutação e também através de um processo de retroalimentação, onde os resultados de falsos positivos e falsos negativos resultam em correções que irão melhorar as classificações futuras. Ao final de cada ciclo, os anticorpos que não foram ativados são penalizados, perdendo pontos, e os ativados ganham pontos, eliminando ao final aqueles que não possuem mais pontos. Os resultados foram comparados com o método *naive* Bayes e mostraram que foi possível detectar mais de 99% das mensagens legítimas e SPAM através do uso deste classificador, concebido a partir de um SIA, enquanto o *naive* Bayes apresentou taxa de acerto acima de 99,5% para mensagens legítimas e entre 96,4% e 98,2% para SPAM.

Existem outros trabalhos com resultados interessantes na área de SPAM e classificação de e-mail utilizando SIA que abordam técnicas diferentes da apresentada nesta seção (Yue, et al. 2007) (Secker, Freitas e Timmis 2003) (Bezerra, et al. 2006).

3.4.1.2. Robótica

Aplicações de aprendizado em SIA também são utilizadas na robótica, onde redes idiotípicas modelam a navegação de robôs autônomos através de ações representadas por anticorpos e situações a serem resolvidas ou tratadas, representadas pelos antígenos (Whitbrook, Aickelin e Garibaldi 2007).

3.4.1.3. Máquinas de Busca

A busca por informações relevantes, positivamente inesperadas ou surpreendentes na Web constitui um grande desafio para as máquinas de busca atuais. A proposta de um sistema para descoberta de informação (*Artificial Immune System for Interesting Information Discovery* - AISIID) (Secker, Freitas e Timmis 2008) oferece uma abordagem diferente, onde o usuário inicialmente fornece páginas de seu interesse para que uma busca posterior mais aprofundada seja realizada. A partir do fornecimento das

páginas de interesse do usuário, o AISIID realiza um processo de busca intenso pela Web com o objetivo de encontrar páginas que contenham informações inesperadas ou surpreendentes ligadas ao interesse do usuário. No AISIID, os antígenos são as páginas Web e os anticorpos são gerados a partir das páginas fornecidas pelo usuário, sendo estes formados por um resumo dos interesses do usuário, uma noção do que o usuário pode achar interessante, uma URL de localização na Web e o estímulo do anticorpo em relação ao antígeno. Como o AISIID realiza uma busca bem mais detalhada na Web, se comparada com outras máquinas de buscas como Yahoo! e Google, o seu experimento registrou duração de 5 a 6 horas. O resultado da busca retornou páginas com conteúdo considerado interessante pelos usuários, incluindo posturas antagônicas e informações desconhecidas a respeito dos assuntos correlatos. É importante ressaltar que este processo deve ser utilizado em situações que necessitem de informações diferenciadas e onde não exista a necessidade de respostas em tempo real.

3.4.1.4. Redução de Instâncias para Melhora de Classificação

A classificação de elementos através da técnica do vizinho mais próximo é muito utilizada no meio acadêmico e em ferramentas de mercado, devido à sua simplicidade e resultados aceitáveis para classes com média ou baixa sobreposição. Como muitas vezes o desempenho e a precisão são comprometidos pela imensa quantidade de amostras a serem classificadas e à presença de ruídos, é muito importante que a massa de dados seja previamente tratada.

Para executar este pré-processamento existem diversos algoritmos (Wilson e Martinez 97) (Marchiori 2008). Pelo fato do SIA apresentar características interessantes sob o ponto de vista de detecção de padrões, Garain (2008) propôs uma abordagem inspirada no sistema imune que trate as necessidades supracitadas. Nesta abordagem, os antígenos são representados como vetores de valores binários de instâncias da base de dados e os anticorpos são vetores gerados para se ligarem aos antígenos. Logo após a geração dos anticorpos, a afinidade é calculada através da distância de Hamming. No decorrer do processo, os anticorpos mais aptos se multiplicam, sofrem mutação e são selecionados como células de memória, em um processo muito semelhante ao algoritmo CLONALG. Os anticorpos de maior afinidade formam uma população representativa da base de dados, os quais serão usados na execução da classificação. Ao final, o método é comparado com outros métodos como uma modificação do *Condensed Nearest Neighbor* (CNN) e os mapas auto-organizáveis (Devi e Murty 2002). Os resultados utilizando o SIA apresentaram a melhor precisão dentre os métodos, equiparando-se com a melhor taxa de redução da base de dados obtida por outro método.

3.4.2. Detecção de Anomalias

3.4.2.1. Detecção de Falhas

A detecção de falhas é um tipo de detecção de anomalias que deve ser evitada para que um equipamento ou processo não seja danificado ou tenha sua eficiência diminuída, o que poderia gerar prejuízos financeiros ou até mesmo danos à integridade humana. Através do algoritmo de seleção negativa, o SIA provê uma opção interessante para abordar este problema e é com esta proposta que foi criado um protótipo que detecta falhas de máquinas de usinagem em uma planta fabril (Dasgupta e Forrest 1999). Através do conhecimento do processo fabril da usinagem de uma peça, este é decomposto em grandezas como temperatura, força de corte, vibração, torque e emissão acústica. Estas grandezas são normalizadas, discretizadas em um *grid* e em seguida representadas no espaço de formas binário. Os antígenos são obtidos através da coleta de parâmetros em um regime normal de funcionamento do processo e os anticorpos, ou detectores, são gerados de maneira mais eficaz que a tradicional geração aleatória de vetores de atributos, evitando assim o excesso de detectores inválidos. A afinidade entre os antígenos e anticorpos se dá através da similaridade de r bits contíguos de uma cadeia de comprimento l . Ao ultrapassar um determinado limiar, a detecção é realizada e um alarme enviado ao operador da máquina indicando a iminência da falha.

Na linha da detecção de falhas, outra interessante aplicação (de Lemos, et al. 2007) baseada em SIA foi desenvolvida para detectar falhas em caixas automáticas bancários (*Automated Teller Machine* - ATM). As ATMs são utilizadas por bancos para prover vários serviços aos clientes, como saques, consultas a extratos, saldos e até mesmo financiamentos. Como estas máquinas devem ficar disponíveis por muitas horas para prover serviço aos usuários, uma falha fatal que as remova de operação deve ser evitada. O SIA foi escolhido como o método para detectar e prevenir indisponibilidade neste ambiente, devido às suas características chave como retenção de memória, aprendizado e capacidade de adaptação. A modelagem se dá através da extração dos arquivos de operação destas máquinas por módulos específicos, como o de saque, por exemplo, onde os registros de falhas são coletados e rastreados até que se encontre uma falha fatal, o que resultaria na inoperância do equipamento. Com base nesta estratégia, os antígenos e anticorpos são modelados, levando em conta a criação de uma sequência que termine com uma falha fatal, com o objetivo de detectar, de maneira antecipada a indisponibilidade da ATM. Os anticorpos são criados, levando em conta variações das sequências de erros previamente coletados das ATMs, para posteriormente serem usados na detecção de sequências que possam resultar em falhas durante o regime de execução dos equipamentos. Neste modelo, os antígenos são as sequências de status que surgem durante o funcionamento das ATMs e uma situação fatal é detectada quando um anticorpo é compatível em pelo menos r bits contíguos de uma sequência que está em execução. Como resultado, a precisão na detecção de um erro fatal variou entre 86,52% a 92,98%, dependendo das falhas detectadas. O tempo médio de antecipação da

falha foi de 12 horas, período suficiente para deslocar um técnico que atuasse preventivamente na falha do equipamento.

3.4.2.2. Sistemas de Detecção de Intrusos

Os sistemas de detecção de intrusos (*Intrusion Detection System* - IDS) são, segundo Aickelin *et al.* (2003), sistemas projetados com a finalidade de “detectar o uso não autorizado, indevido e o abuso de sistemas computacionais por intrusos externos e internos”. Os IDSs investigam a detecção de ameaças em diversas instâncias, desde a percepção de tráfego suspeito de rede, passando pela interrupção abrupta de processos até o acesso não autorizado de arquivos, dentre outros.

O IDS tem, na sua essência, a necessidade de percepção do que é confiável e não confiável, o que pode ou não representar uma ameaça. Desta maneira, ele atua para detectar comportamentos indesejados e evitar prejuízos financeiros, de infra-estrutura e imagem corporativa causados por invasões internas e externas, muitas vezes originadas por funcionários insatisfeitos ou hackers. Entendendo que um IDS lida com a detecção da intrusão, o SIA torna-se uma opção natural para modelá-lo, pois trata-se de um paradigma inspirado na capacidade do sistema imune de identificar o próprio do não próprio e agir de acordo com esta diferenciação, eliminando os elementos considerados estranhos, ou antígenos, que são as ameaças à integridade do indivíduo.

A teoria do perigo proposta por Matzinger (1994) pode ser vista como uma abordagem mais representativa para o desenho de um IDS, pois além de lidar com a percepção do próprio e não próprio, ela discrimina o que pode ou não caracterizar o perigo, onde conceito de perigo é mais amplo que o do próprio e não próprio, pois o perigo pode estar tanto no indivíduo como fora dele, como já colocado anteriormente.

Em Aickelin *et al.* (2003) foi proposta uma abordagem de projeto de um IDS baseado na teoria do perigo, na qual a combinação de sinais emitidos por células que têm morte natural e anormal foi a inspiração para propor a correlação entre alertas menos críticos e outros mais críticos para identificar um padrão mais amplo de intrusão. Outra abordagem interessante utilizada foi a da zona de perigo, na qual as APCs próximas do estímulo capturam os antígenos e os apresentam aos linfócitos como as células B, que próximas ao evento também podem reagir aos estímulos, tornando-se assim ativadas. A transposição deste processo para o IDS se dá na ativação de diversos sensores que tenham uma proximidade temporal, espacial ou funcional relacionada com o sinal inicial de perigo como, por exemplo, o bloqueio do tráfego de rede de um IP suspeito a um servidor de correio eletrônico pode gerar o bloqueio deste IP para todos os demais servidores de correio eletrônico da corporação.

Em outra abordagem (Aickelin e Greensmith 2007) são trazidos conceitos mais elaborados da teoria do perigo, que apresenta algoritmos do sistema imune inato, como o de células dendríticas (*Dendritic Cell Algorithm* - DCA), com objetivo de aproximar a modelagem de um IDS de outros princípios do

sistema imune biológico e que, segundo os autores apresentou resultados estimulantes.

3.4.2.3. Otimização

Otimização é um processo que objetiva encontrar a melhor solução ou ponto de operação de um sistema da maneira mais eficiente possível, levando em conta as restrições existentes (Wang, Gao e Ovaska 2004). O SIA possui diversos atributos relevantes para uso nesta tarefa, como reconhecimento de padrões, conhecimento do próprio e capacidade de aprendizado, dentre outros.

Como exemplo, uma abordagem híbrida entre o modelo imunológico e a computação evolucionária foi aplicada na resolução de problemas de otimização na área de desenho estrutural (Yoo e Hajela 1999) . Esta combinação de métodos foi aplicada em diversas tarefas de engenharia, em especial na escolha de elementos estruturais, como treliças e vigas que apresentem propriedades como baixo peso, grande resistência à tração e à compressão. Nesta aplicação, os anticorpos correspondem a opções de desenho e os antígenos a um conjunto de pesos que serão utilizados para avaliar a real utilidade dos anticorpos a uma função multicritério. O espaço de formas utilizado é um espaço de formas binário de Hamming, onde a adaptação dos anticorpos é realizada através dos processos de mutação e *crossover* para que estes se tornem cada vez mais aptos e atendam aos objetivos estabelecidos.

Existe um grande leque de aplicações, bem maior do que as foram aqui expostas que incluem áreas como segurança computacional, controle de tarefas e prazos, análises médicas, redes de sensores e servidores Web, dentre outros (de Castro e Timmis 2002) (Hart e Timmis 2008).

3.5. Inteligência Artificial, Computacional e o Sistema Imunológico Artificial

Desde a criação do termo Inteligência Artificial (IA) cunhado no Dartmouth College, em 1956, por Jonh MacCarthy (McCarthy 1995), o interesse por esta área que se relaciona com tantas outras áreas do conhecimento, como psicologia, biologia, sociologia, matemática, filosofia e computação continua em franca expansão. Em 1996, o Conselho de Redes Neurais do IEEE delineou o objetivo da IA como "...criar um computador que faça coisas que atualmente os humanos fazem melhor". Esta definição é muito ampla e diversas abordagens fazem referência a situações mais específicas, como a manipulação de símbolos, pensamento, criatividade e imaginação (Fulcher e Jain 2008). John MacCarthy definiu IA como sendo "a ciência e a engenharia de criar máquinas inteligentes, em especial programas inteligentes."

Como pode ser percebida, a conceituação de IA tem um amplo espectro de abrangência, mas tendo sempre em mente criação de máquinas ou computadores que reproduzam a capacidade da inteligência humana para os

mais diversos fins, seja para a resolução de problemas ou para o próprio estudo da inteligência.

A abordagem tradicional da IA à resolução de problemas é feita com base na representação simbólica das instâncias dos problemas e na construção de regras que guiam o processo por diversas transições até se chegar à solução (Amit 2005). As situações são constantemente avaliadas por regras do tipo SE-ENTÃO e, caso obedecerem às condições, a regra é disparada, ou executada. Este modelo é concebido para ser utilizado no sentido geral-específico, sendo complexo e de manutenção custosa, pois cada mudança, devido às alterações do problema ou dos objetivos, torna necessário realizar alterações nos seus componentes, demandando tempo e conhecimento especializado.

Os problemas iniciais tratados pela IA tradicional eram baseados em uma série de passos que os humanos usavam para resolver quebra-cabeças, jogos de tabuleiro ou fazer deduções lógicas. Estes métodos evoluíram bastante na década de 80 e 90, mas ainda sofriam do consumo excessivo de recursos computacionais (Norvig e Russel 2003).

Para lidar com problemas de difícil modelagem e tratamento simbólico, constituiu-se, como alternativa, um grupo de abordagens específicas não simbólicas com características ou comportamentos inteligentes. Estas abordagens, apesar de diferentes entre si, compartilham um ponto em comum, o tratamento da maioria dos problemas no sentido específico-geral, onde a solução emerge de um início aparentemente desestruturado e não de conhecimento formal previamente (Craenen e Eiben 2003).

O nome dado ao conjunto de abordagens que apresentam este tipo de comportamento é Inteligência Computacional, ou IC. Dentre as abordagens mais conhecidas de IC estão as redes neurais, lógica fuzzy e a computação evolutiva, já introduzidas no Capítulo 2, mas é interessante ressaltar que outras abordagens como máquinas de vetor suporte, Raciocínio Baseado em Casos, fractais e teoria do caos, dentre outros, já são citadas (Fulcher e Jain 2008) como membros da IC.

Segundo alguns autores, IC pode ser tratada como:

Essas tecnologias como as redes neurais, fuzzy e sistemas evolucionários foram abrigadas sob a rubrica de IC, um termo relativamente novo utilizado para descrever de maneira geral métodos computacionais que possam adaptar-se a novos problemas e que não se atenham a um conhecimento humano explícito (Fogel 2006).

Sendo mais específico que Fogel, Bezdek caracteriza um sistema computacionalmente inteligente como:

Um sistema é computacionalmente inteligente quando: lida apenas com dados numéricos (baixo nível), possui componentes de reconhecimento de padrões, não utiliza o conhecimento tradicional da IA e ainda quando começa a exibir (1) adaptação computacional; (2) tolerância a falhas computacionais; (3) rapidez na exploração de possibilidade e (4) obtenção de taxas de erro que se aproximam do desempenho humano (Bezdek 1994).

Apesar das diferenças entre várias definições de IC (Craenen e Eiben 2003), seja na ênfase do atendimento à capacidade de adaptar-se a novas situações ou detectar padrões, o que as liga é a extração da solução através da análise específica do problema, e não pelo tratamento homogêneo do problema por um modelo do domínio do conhecimento.

Através da apresentação do SIA, são perceptíveis diversas características inerentes ao grupo de paradigmas que compõe a IC, possuindo semelhanças e também diferenças que o tornam uma ferramenta extremamente versátil na solução de problemas (Garrett 2005). A Tabela 3.1 apresenta as principais características do SIA, suas semelhanças e diferenças em relação aos paradigmas mais comuns de IC que permitem habilitá-lo (de Castro e Timmis 2002) (Engelbrecht 2007) como membro relevante deste grupo.

Especificidades do SIA	Semelhanças de SIA com outros paradigmas de IC*
<ul style="list-style-type: none"> • Inspiração no sistema imune. • Aumento da especificidade através de mutação por afinidade e pressão seletiva. • Conhecimento do próprio e do não próprio. • Armazenamento da cadeia de atributos e de suas relações. 	<ul style="list-style-type: none"> • Detecção de padrões e aprendizado. • Capacidade de generalização e de lidar com incertezas. • Estrutura formada por um conjunto discreto ou uma rede de elementos. • Representação através de um conjunto de atributos. • Paradigma evolucionário e/ou de aprendizado. • Memória. • Robustez. • Metadinâmica.

Tabela 3.1 - * A semelhança se dá com qualquer um dos seguintes paradigmas: redes neurais, lógica fuzzy e computação evolucionária (de Castro e Timmis 2002).

Este conjunto de semelhanças e diferenças fornece subsídios para uso de SIA em abordagens híbridas, assim como já proposto por Soft Computing, ou SC, na qual paradigmas como lógica fuzzy, computação evolutiva e redes neurais artificiais podem ser usados em sinergia para resolver problemas com um desempenho muitas vezes melhor do que as abordagens individuais.

O uso de SIA com as RNAs pode ser realizado de diversas maneiras, sendo a mais comum o uso de algoritmos de otimização baseados no SIA para gerar vetores de pesos que serão utilizados em uma rede neural, de modo a melhorar a velocidade de aprendizado através da exploração do espaço, diminuindo assim a probabilidade de ater-se a mínimos locais. Esta proposta foi implementada no algoritmo SAND (de Castro 2001). SIA também foi aplicado na aprendizagem competitiva de RNAs através do algoritmo ABNET (de Castro 2001), onde os processos de seleção clonal e maturação por afinidade realizam a geração automática da topologia da rede. É interessante também citar o uso da rede imunológica nas RNAs para uso em conjuntos de classificadores (García-Pedrajas e Fyfe 2007), onde a representação dos antígenos se dá através das entradas da rede e dos anticorpos pela própria rede neural. Como um anticorpo, a rede neural está sujeita ao processo de mutação, podendo acrescentar e remover neurônios e alterar pesos, dependendo da afinidade entre os antígenos e os anticorpos. Nesta proposta os anticorpos interagem entre si e podem ser suprimidos para gerar diversidade. As afinidades entre antígenos e outros anticorpos são analisadas em conjunto para gerar uma rede imunológica. Nesta rede imunológica, os anticorpos somente interagem com aqueles anticorpos que foram estimulados pelo mesmo antígeno, criando assim áreas de abrangência de antígenos.

Apesar de aparente sobreposição, devido às suas características evolutivas, SIA e computação evolutiva podem interagir para propor abordagens conjuntas interessantes, como a manutenção da diversidade na população de anticorpos e atendimento a múltiplos objetivos em uma busca genética com base em restrições (de Castro e Timmis 2002). A lógica fuzzy também fornece contribuições interessantes quando utilizada com SIA, pois o próprio comportamento da ligação entre os antígenos e os anticorpos é intrinsecamente nebuloso, não sendo tratado pela exatidão, mas sim pela afinidade aproximada entre o anticorpo e o epítipo. Além disso, diferentes problemas de agrupamento em SIA podem ser tratados com a lógica fuzzy, onde os anticorpos são distribuídos pelo espaço de formas, seguindo os padrões de distribuição dos antígenos e a criação dos grupos, ou *clusters*, é feita com base na pertinência de cada elemento aos grupos detectados (de Castro 2001).

Além destes tradicionais paradigmas de IC, outros também podem ser utilizados em conjunto com SIA, como sistemas classificadores, computação DNA e Raciocínio Baseado em Casos (RBC). É de interesse especial neste trabalho a conjugação entre SIA e o RBC, pois sendo RBC uma metodologia, sem tecnologia ou ferramenta intrínseca, o SIA pode oferecer importantes contribuições nos processos formadores do ciclo de vida do RBC. Um modelo híbrido do RBC e SIA será apresentado no Capítulo 4.

Capítulo 4 - Raciocínio Baseado em Casos e o Sistema Imune Artificial

4.1. Introdução

O Raciocínio Baseado em Casos é uma metodologia que permite a solução de situações ou problemas, baseando-se em situações semelhantes previamente conhecidas. Como metodologia, o RBC não especifica uma técnica ou paradigma que possam ser utilizados na execução de seus processos, estando aberta àqueles que melhor se adequam ao problema ou situação a serem resolvidos.

Pelas suas características, o Sistema Imune Artificial é um paradigma que pode ser colocado como pertencente a sistemas de *Soft Computing* e inteligência computacional, podendo também trabalhar em conjunto com outros paradigmas para oferecer soluções híbridas que apresentem melhor desempenho em relação a técnicas individuais.

As metáforas do sistema imune humano oferecem opções interessantes, sob a forma do SIA, que podem ser utilizadas nos sistemas de RBC ao longo dos seus processos de recuperação, reutilização, revisão e retenção.

As características de detecção de padrões e compressão de dados do SIA podem ser utilizadas para facilitar o processo de busca e recuperação por casos e fornecer informações para detectar grupos de casos que apresentam características semelhantes e estabelecem relações de proximidade. Neste sentido, a rede imunológica é de grande interesse, pois forma relações de proximidade entre os anticorpos baseados nas suas reações com os antígenos e os próprios anticorpos. Outros processos do RBC, como a reutilização e revisão também podem ser contemplados através da edição somática, onde a solução dos casos mais próximos, ou parte dela, poderia ser modificada para fornecer uma solução viável para o problema ou situação proposta. Por último, a retenção ou armazenamento dos casos se dará pela proximidade entre eles, criando áreas de conhecimento especializado através de uma rede de casos, o que contribui para a descoberta de grupos específicos de casos e também para o fornecimento de informações importantes que facilitem a manutenção da base de casos.

Este capítulo apresenta os trabalhos precursores que relacionam o RBC e o SIA, propõe uma abordagem conjunta mais atualizada entre o RBC e o SIA

(onde é colocado um modelo de utilização das características mais relevantes do SIA para o RBC, levando-se em conta as potenciais contribuições do SIA nos seus processos) e as justificativas que fundamentam a proposta deste modelo híbrido.

4.2. Trabalhos Anteriores

A seguir estão os trabalhos precursores mais relevantes que inspiraram a abordagem híbrida aqui colocada.

4.2.1. Memória de Casos e o Sistema Imunológico Artificial

O potencial da aplicação de SIA em RBC foi inicialmente percebido e explorado em meados da década de 90 (Hunt, Cooke e Holstein 1995). A partir deste ponto, houve pouco desenvolvimento deste modelo híbrido, talvez devido ao momento incipiente vivido por SIA na época e à existência de outros paradigmas mais consolidados, como computação evolutiva e redes neurais.

Neste trabalho precursor, foram identificadas as restrições ou problemas enfrentados pelos sistemas RBC e como o SIA poderia endereçar estas restrições, partindo de suas características de memória, detecção de padrões e aprendizado.

O RBC tem como princípio, a existência de casos contextualizados organizados na sua base de casos, que devem ser recuperados e utilizados na solução de casos semelhantes. Desta maneira, a quantidade de casos, sua forma de indexação e organização são cruciais para que um caso relevante seja encontrado. Com as restrições da complexidade da indexação, crescimento da base de casos e aumento do tempo de desenvolvimento de sistemas de RBC, torna-se interessante uma abordagem que organize a sua própria memória e a utilize para recuperar seus elementos, constituindo-se como uma opção em potencial para minimizar essas restrições.

A rede imunológica foi o processo do sistema imune escolhido para estabelecer um novo tratamento ao RBC, pois ela possui a capacidade de reter o conhecimento aprendido e continuamente reforçado, esquecer ou remover o que não é utilizado e estabelecer vínculos entre seus elementos, criando regiões de diferentes densidades no espaço.

Os autores modelaram a rede imunológica para o RBC, considerando que os anticorpos são os casos armazenados na base de casos e os antígenos são os novos casos, ou casos alvo, gerando uma reação imune quando apresentados ao sistema. Esta reação objetiva detectar qual ou quais anticorpos são os mais próximos em relação ao antígeno ou caso a ser resolvido, selecionando assim os mais semelhantes para resolver o problema ou situação representado pelo caso alvo. A medida de afinidade utilizada para calcular a similaridade entre os casos é a medida de Hunt, mostrada na

Equação (4.1), que considera o número e a sequência de atributos iguais existentes entre os casos.

$$\text{Medida de Hunt} = D_H + \sum_i 2^{l_i} \quad (4.1)$$

Onde D_H é o complemento da distância de Hamming total e l_i é o comprimento l de cada parte i da região que possui mais do que dois atributos iguais. A identificação de um caso potencial se dará se o valor da medida de Hunt for maior que um limiar pré-estabelecido pelo solucionador do RBC. Um exemplo é ilustrado na Tabela 4.1.

Casos novos	Aventura	2498	2	Egito	Carro	14	Mar.	*	
Caso analisado	Aventura	3066	3	Egito	Carro	14	Abr.	*	
Avaliação	1	0	0	1	1	1	0	1	= 5
Comprimento				3					= 3
Afinidade	5 + 2 ³ = 13								

Tabela 4.1 - Lista dos atributos e cálculo da semelhança entre os casos.

A inserção dos casos na base de casos é realizada pelo seguinte algoritmo:

Algoritmo que insere casos

1. **Para** fator de inserção
 - 1.1. Selecionar aleatoriamente n posições na rede imunológica;
 - 1.2. **Até** não haver afinidade maior do que o *atual conjunto de melhores casos*
 - 1.2.1. Selecionar m casos de maior afinidade do conjunto de n casos;
 - 1.2.2. **Se** um ou mais casos ligados a estes m casos possuírem uma afinidade maior que pelo menos um caso do *atual conjunto de melhores casos então*
 - 1.2.2.1. Acrescentar ao *atual conjunto de melhores casos* e remover os que possuírem menor afinidade;
 - 1.2.2.2. Considerar os casos no entorno do caso acrescentado.
 - 1.2.3. **Fim Se**
 - 1.3. **Fim Até**
2. **Fim Para**
3. **Para** cada caso x no *atual conjunto de melhores casos*
 - 3.1. Conectar o novo caso a x como um filho de x ;
 - 3.2. Conectar o novo caso a todos os casos filhos (irmãos) de x ;
4. **Fim Para**

Este processo de inserção é uma aproximação do funcionamento da rede imunológica na qual os casos semelhantes são interconectados entre si segundo uma medida de afinidade que avalia sequências de atributos e recupera casos efetuando uma busca aleatória na base de casos.

O primeiro laço do 4.1, na linha 1 determina a quantidade de vezes que serão selecionados conjuntos de casos no espaço, permitindo que mais casos sejam recuperados na base de casos. O número n corresponde à seleção de 10% das posições da rede imunológica e o *atual conjunto de melhores casos* contém os 10 melhores casos considerados até o momento. O laço interno da linha 1.2 verifica se os n casos e seu entorno contém casos melhores que o *atual conjunto de melhores casos* e, se positivo, estes substituirão os casos de menor afinidade nesta lista.

Após a análise dos casos mais relevantes da base de casos em relação ao caso apresentado, é feita a inserção do caso apresentado na linha 3.1 e 3.2, ligando-o como filho dos casos da lista *atual conjunto de melhores casos* e também aos filhos destes casos, ou seja, seus casos irmãos, estabelecendo-se, desta maneira, uma relação de similaridade que cria a rede de casos, como colocado na Figura 4.1.

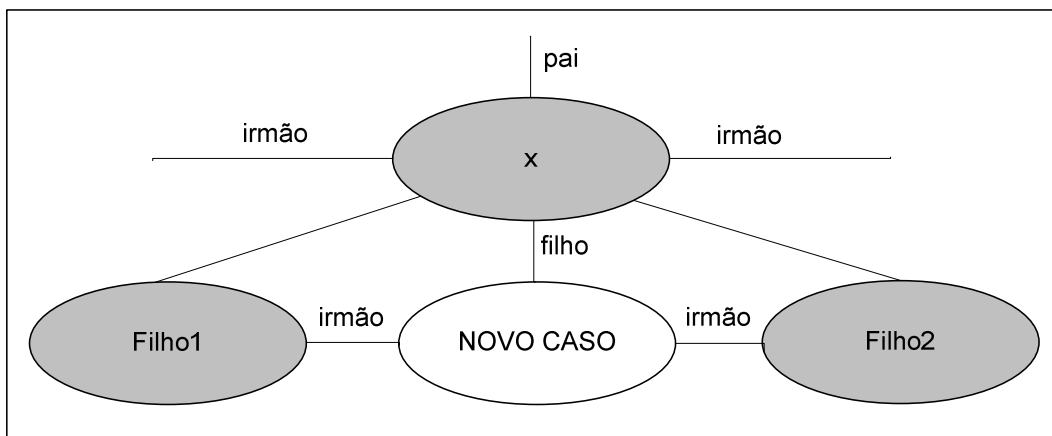


Figura 4.1 – Inserção de novo caso (Hunt, Cooke e Holstein 1995).

A recuperação de casos é realizada pelo seguinte algoritmo:

Algoritmo de recuperação de casos

1. Selecionar aleatoriamente n pontos na rede de casos;
2. **Até** não existirem mais casos a serem considerados
 - 2.1. Considerar cada caso representado por estes pontos e calcule a afinidade de cada um em relação ao caso apresentado;
 - 2.2. **Se** o valor da afinidade for maior que um determinado limiar **então**
 - 2.2.1. Selecionar este caso e os casos no seu entorno para análise posterior;
 - 2.3. **Fim Se**
3. **Fim Até**
4. Ordenar os casos pelo valor da afinidade;

Algoritmo 4.2 – Recuperação de casos inspirada na rede imune.

O objetivo da recuperação de casos é recuperar os casos mais próximos em relação ao caso em questão sem pesquisar toda a base de casos. Na linha 1 são selecionadas n posições aleatórias na base de casos a serem pesquisadas. Se o caso apresentar uma afinidade acima de um determinado limiar, o seu estímulo é calculado e considera-se que os casos irmãos e filhos têm potencial para serem explorados em momento posterior, pois a base de casos cresce constantemente, tornando proibitiva uma busca exploratória a todo o momento.

A busca pode terminar, quando uma das seguintes condições for atingida: a busca por toda a base de casos; expiração de um período de tempo pré-determinado ou permanência do mesmo caso como melhor caso após duas (valor válido para o experimento realizado) iterações seguidas. No protótipo feito pelos autores, dez casos são retornados ao usuário no final do processo.

Os autores também se inspiram no sistema imune para propor uma política de remoção de casos, que é baseada no cálculo do estímulo entre anticorpo-anticorpo e anticorpo-antígeno, correspondendo ao estímulo entre os casos da base de casos em relação aos seus casos vizinhos e ao novo caso apresentado. Se alguns casos da base de casos oferecem uma solução inadequada e também não apresentarem boa similaridade com os seus casos vizinhos, estes casos são candidatos a eliminação. A Equação (4.2) representa o cálculo do estímulo.

$$\text{Estímulo} = c \left[\sum_{j=1}^N sm(x_i x_j) + k_1 \sum_{j=1}^N sm(x_i y) \right] - k_2 \quad (4.2)$$

Onde N é o número de casos, x_i é o caso analisado da base de casos, x_j são os outros casos da base, y é o caso apresentado ao sistema, sm representa a similaridade dos casos. A constante c é dependente da taxa de comparações na unidade de tempo e k_2 é um valor que representa um estímulo negativo natural dos elementos da rede. Desta maneira, $\sum_{j=1}^N sm(x_i x_j)$

representa a afinidade dos casos em relação aos seus vizinhos, $\sum_{j=1}^N sm(x_i y)$ a afinidade entre os casos da rede e o caso apresentado, e k_1 é um fator de equilíbrio que reforça a afinidade de um caso da rede em relação ao caso apresentado mesmo com pouco estímulo advindo da sua vizinhança, colaborando assim para evitar a eliminação de casos com forte acoplamento.

Usando estes processos inspirados no sistema imune foi desenvolvido um protótipo, o qual teve o seu modelo de memória e recuperação de casos comparado qualitativamente com outros métodos, como vizinho mais próximo e árvores de decisão. Segundo os autores, o modelo inspirado no sistema imune apresenta maior número de vantagens em relação aos demais métodos segundo critérios de busca, organização e manutenção da base de casos.

A abordagem exposta possui algumas características que são derivadas da interpretação do sistema imune e outras que são ligadas às particularidades do domínio (pacotes de viagem turística) do problema (Hunt, Cooke e Holstein 1995):

- Representação simbólica: os atributos dos casos de teste são, na sua grande maioria, formados por conjuntos de caracteres que indicam os tipos de feriado, região, transporte, duração e conforto dos pacotes de viagem;
- Medida de afinidade: a medida de afinidade é baseada na similaridade de atributos contíguos e na utilização de um limiar para que os casos da base de casos sejam considerados aptos a propor uma solução ao novo caso;
- Processo imunológico: utiliza a afinidade entre anticorpo-anticorpo e anticorpo-antígeno para construir a rede imunológica inspirada no modelo de Perelson (Farmer, Packard e Perelson 1986) utilizando-se de características de reconhecimento de padrões e aprendizado para criação da memória;
- Aleatoriedade: o início dos processos de inserção e busca são baseados na seleção aleatória de pontos na rede de casos;
- Manutenção: através do estímulo entre anticorpo-anticorpo e anticorpo-antígeno estabelece-se uma política de remoção de casos desnecessários.

Segue, na Tabela 4.2, o resumo (de Castro e Timmis 2002) da relação entre RBC e SIA baseada no trabalho precursor aqui apresentado (Hunt, Cooke e Holstein 1995):

Sistema imunológico	Raciocínio Baseado em Casos
Antígeno	Novo caso
Anticorpo	Casos armazenados
Rede imunológica com seus anticorpos e ligações entre eles	Base de casos

Sistema imunológico	Raciocínio Baseado em Casos
Conexão anticorpo-antígeno	Compatibilidade de casos
Resposta imunológica secundária	Recuperação de casos
Recrutamento de novos anticorpos	Inserção de casos
Eliminação de anticorpos não estimulados	Remoção de casos

Tabela 4.2 – Analogia entre elementos do sistema imune e do Raciocínio Baseado em Casos.

Como continuidade, um dos autores investiu no relacionamento entre o RBC, SIA e mineração de dados (Hunt e Fellows 1996). Neste trabalho, o processo de criação da rede imunológica é realizado de maneira mais eficaz, removendo a necessidade de gerar constantemente os relacionamentos entre os casos, quando a base de casos é criada. Além disto, são introduzidas a mutação de casos e o uso do estímulo para criar a rede de casos e a supressão de casos. Este novo modelo de integração entre o RBC e SIA apresenta quatro etapas:

- Inicialização da memória: nesta etapa um montante inicial de casos é armazenado de maneira serial, ou seja, o caso que entra é armazenado no final da base de casos, não existe nenhuma estrutura de indexação e nenhuma operação é realizada com os casos. O objetivo é o de fazer uma coleta inicial de casos;
- Generalização da memória: neste ponto, mais casos são apresentados para compor a base de casos e a afinidade entre todos eles é contabilizada. Os melhores N casos encontrados terão o seu estímulo atualizado com base na afinidade entre o novo caso. Quando o estímulo ultrapassa um determinado limiar, o caso entra em processo de mutação, o que gerará número de clones proporcional à diferença entre o limiar e o valor do estímulo do caso. Em seguida o componente problema dos clones sofre mutação em baixas taxas, para que estes se posicionem em pontos próximos do espaço em relação ao caso original. É importante ressaltar que os atributos de solução do clone não sofrem mutação. Ao final, se um caso mutante estiver abaixo do limiar, ele não será incluído na base de casos e os que superaram o limiar serão inseridos da mesma maneira que na etapa anterior, ou seja, ao final da base de casos;
- Refinamento da memória: o refinamento consiste na remoção de todos os casos da base de casos que estão abaixo do limiar de estímulo (dependente da aplicação). Além desta atividade, é nesta etapa que acontece a criação da rede de casos propriamente dita, utilizando-se uma variação do 4.1. Ao final desta etapa, a rede de casos está criada e aglomerados de casos que lidam com problemas semelhantes podem surgir, o que é algo extremamente interessante para a mineração de dados;

- **Classificação:** esta é a etapa em que casos nunca vistos pelo sistema são submetidos à classificação do sistema para que a sua solução seja obtida pela reutilização dos casos armazenados na base de casos, estruturados sob a forma de uma rede inspirada na rede imunológica biológica. O algoritmo de recuperação de casos é baseado no Algoritmo 4.2, no qual regiões aleatórias da base de casos são selecionadas para busca do melhor ou melhores casos.

Este método foi comparado com a memória linear (sem criação da rede) e a memória em rede (sem mutação). Segundo os autores, a proposta da memória com rede imunológica teve desempenho melhor que todos os demais métodos.

4.2.1.1. Considerações

Apesar do trabalho de Hunt, Cooke & Holstein (1995) representar uma abordagem precursora, ele apresenta restrições para uso em alguns domínios de problema devido à sua forte característica não determinística, além de não apresentar resultados numéricos. As principais considerações são:

- **Não determinismo:** o processo de inserção e recuperação de casos é iniciado através da seleção de pontos aleatórios da rede de casos. A partir destes pontos, é dada continuação a uma pesquisa no entorno dos melhores casos encontrados segundo a medida de afinidade de valores contíguos. A limitação deste tipo de busca é que nem sempre o melhor caso ou nem mesmo um bom caso sejam recuperados da base de casos, pois a busca aleatória de posições, na rede de casos, pode resultar na seleção de áreas do espaço que possuem baixa afinidade com o caso apresentado, causando assim latência no processo da solução. Aplicações de detecção de falhas e diagnóstico são especialmente suscetíveis a esta característica de aleatoriedade do processo de busca, pois os resultados podem apresentar grande diferença;
- **Criação de hierarquia complexa:** o processo de inserção de casos gera a criação de conexões hierárquicas entre o novo caso e os casos mais semelhantes encontrados no processo de busca aleatório. Estas conexões ligam o novo caso a todos os casos considerados mais semelhantes, chamados de casos pai e, também, a todos os casos filhos dos casos pai. Estas conexões criam uma rede de casos semelhantes que podem se tornar bastante complexas com o aumento de número de casos armazenados e que devido ao processo de inserção não determinístico não necessariamente representem os casos mais semelhantes da base de casos;
- **Remoção de casos menos utilizados:** a remoção de casos se dá pela conjugação da afinidade que eles recebem, principalmente quando reagem com os casos já presentes na rede e com os novos casos apresentados. Ao final, os casos menos estimulados, ou seja, menos utilizados, são eliminados da base de casos. Este método remove

casos menos utilizados, mas que em contrapartida podem ser extremamente representativos para a competência (Smyth e McKenna 1998) de um sistema de RBC, o que na prática pode levar à incapacidade de resolução de problemas futuros;

- Ausência de resultados numéricos: o modelo proposto de rede de casos foi comparado com alguns outros métodos, gerando informações qualitativas como resultado destas comparações. Além das informações de cunho qualitativo, é importante que sejam gerados números que expressem as grandezas, contribuindo assim para uma percepção mais precisa das suas vantagens e desvantagens.

Em Hunt & Fellows (1996), o processo preliminar de armazenamento linear simplifica a criação da base de casos, já que a triagem dos casos para criação da rede é feita em etapa posterior, o que minimiza operações de busca na base de casos e criação de vínculos entre casos antes do momento necessário. Outro ponto interessante é a clonagem, seguida da mutação de casos que apresentam um estímulo acima do limiar pré-determinado, onde os casos gerados são ligeiramente diferentes do original, mas sem modificação da componente solução. Como é feita uma verificação do estímulo destes casos mutantes em relação ao novo caso, um resultado acima do limiar valida o caso também como possível solução e ele poderá ser inserido na base de casos. Este comportamento contribui para reforçar a cobertura do problema em regiões próximas ao caso inicialmente recuperado que originou o processo de expansão clonal. Esta abordagem é interessante, mas deve ser utilizada com critérios, pois se existirem muitas soluções diferentes no espaço de soluções, um caso mutante ligeiramente acima do limiar pode representar a indicação de uma solução mais específica, diferente da sinalizada pelo caso mutante que, na proposta apresentada, mantém a mesma solução do caso original. Após a etapa de refinamento da memória, a inserção de novos casos ao sistema é iniciada através de uma busca aleatória, o que acarreta a mesma limitação do trabalho anterior (Hunt, Cooke e Holstein 1995), ou seja, o não determinismo da busca e, também, a maior possibilidade de baixa eficiência em grandes bases de casos.

4.2.2. Busca Direcionada de Casos

Smyth & McKenna (1998) (1999) propuseram uma busca direcionada na base de casos, utilizando casos de referência (*Footprint*) como índices para regiões do espaço. Neste trabalho, introduzido no Capítulo 2, os casos de referência devem ser criados usando o princípio da competência, que é a capacidade de resolver um caso. Os mais competentes são usados como referência, ou índices, criando áreas de influência na sua região do espaço. Ao término da execução deste processo na base de casos, seus elementos são agrupados pelas suas áreas de competência para formar grupos de casos. A partir deste ponto a busca pode ser efetuada através dos casos de referência, que são geralmente em número bem menor que a base de casos. Ao se encontrar o caso de referência mais similar em relação ao novo caso, é realizada uma busca local na vizinhança deste caso de referência com o objetivo de encontrar

o caso mais relevante possível na base de casos. Ao final avalia-se esta proposta segundo os critérios de eficiência (o inverso do número de casos examinados durante a recuperação), competência (percentagens de problemas que podem ser resolvidos), qualidade (distância média entre o novo caso e o caso recuperado) e excelência (frequência de recuperação do melhor caso) em relação a dois tipos de buscas: uma busca completa na base de casos e uma busca na base de casos representada somente pelos seus casos referência.

Outro trabalho (Patterson, et al. 2002) que faz uso da busca direcionada, emprega o centro dos grupos de casos formados através do algoritmo k-means (Xu e Wunsch 2005) para otimizar o processo de busca. Neste processo, grupos são gerados e, ao final desta etapa, um novo caso é comparado com o centro dos grupos (em quantidade muito menor que os casos da base de casos) existentes. O grupo que apresentar o centro mais próximo do novo caso terá os seus elementos analisados com o objetivo de encontrar o mais próximo a este novo caso. Com objetivo de melhorar o desempenho do k-means em relação ao crescimento do número de grupos, uma versão otimizada foi desenvolvida para o problema. Os resultados deste método foram comparados com uma busca completa na base e demonstraram um aumento da eficiência e uma competência muito próxima da recuperação utilizando uma busca completa.

Em referência a métodos de busca baseados em grupos, Kolodner (1993) coloca que:

se você pode agrupar casos similares entre si e perceber em qual dos grupos se encontra o melhor caso, apenas os casos deste grupo precisam ser averiguados.

4.2.2.1. Considerações

Os dois últimos trabalhos citados direcionam a busca de maneira diferente, mas utilizam de referências, ou índices, baseados em grupos de casos para otimizar o processo de busca. Com base nestas características, seguem algumas considerações:

- O processo de criação dos casos de referência no método *Footprint*, analisa para cada caso todos os demais da base. Isto significa que para serem criados os casos de referência será necessário um tempo quadrático $O(n^2)$, onde n é o número de casos. Após a criação dos casos de referência, o processo de busca analisa todos eles para que o mais próximo seja encontrado. Em grandes bases seria interessante a possibilidade de criarem-se grupos de referência e utilizar seus centros para otimizar mais ainda o processo de busca, já que os centros e os casos de referência são em menor quantidade, se comparados com o restante da base;
- O método de busca direcionada que usa o algoritmo k-means trabalha com a criação de diferentes números de grupos para otimizar o processo de busca. À medida que a quantidade de grupos cresce, o tempo de recuperação de um caso aumenta, pois é necessário um

esforço para criar os novos grupos. Uma opção interessante seria a criação de casos de referência dentro dos grupos para otimizar a busca local e, desta maneira melhorar o desempenho global da busca, evitando passar por todos os casos, o que é significativo em grupos muito grandes.

4.3. Proposta de utilização do Sistema Imunológico Artificial no Raciocínio Baseado em Casos

Baseando-se nas considerações e restrições dos trabalhos citados e atendo-se à evolução de SIA em mais de dez anos de pesquisas e publicações, é proposta aqui uma abordagem que conjuga as características de SIA que possam oferecer contribuições nos processos de recuperação, reutilização, revisão e retenção (armazenamento) de casos que, juntos, constituem o ciclo de vida da maioria dos sistemas de RBC.

A proposta integrada utiliza um algoritmo inspirado no sistema imune biológico, especificamente nos processos de seleção clonal e da rede imunológica. Este algoritmo permite a criação de um tipo de rede imunológica artificial, a aiNet, apresentada no Capítulo 3, que será utilizada para detectar a estrutura interna dos antígenos e realizar a compressão de dados com objetivo principal de permitir uma busca direcionada no espaço dos casos. A partir deste ponto, novos casos podem encontrar casos semelhantes na base de casos com maior precisão, conjugando-se técnicas de agrupamento, além de também possibilitar a adaptação através da utilização de informações de vizinhança entre regiões fronteiriças da rede de casos. Ao final do processo, novos casos são armazenados na rede com base na sua relação de semelhança com os casos já existentes, estabelecendo conexões com os casos mais próximos.

São colocadas, a seguir, a importância das técnicas de agrupamento no contexto do RBC e as propostas do SIA para cada um de seus processos.

4.3.1. Agrupamento

Esta seção tem como objetivo fazer uma breve introdução sobre agrupamento e dispor as considerações gerais sobre este tópico no RBC, não pretendendo prover uma visão extensa e muito menos detalhada do assunto, visto não ser este o escopo deste trabalho. Para maiores informações sobre agrupamento consultar Xu & Wunsch (2005).

Apesar do conceito não encontrar unanimidade na comunidade científica, o agrupamento é considerado, de maneira geral, como “a classificação não supervisionada de padrões (vetores de características, itens de dados ou observações) em grupos (*clusters*)” (Jain, Murty e Flynn 1999) (Xu e Wunsch 2005). A taxonomia dos grupos também não é um consenso, mas o formato mais aceito os separa em:

- Agrupamento hierárquico: agrupa objetos em uma sequência de partições, tanto de grupos com um único elemento como de um grupo contendo todos os elementos. Os algoritmos de agrupamento hierárquico são classificados em métodos divisivos e aglomerativos. Os aglomerativos iniciam o processo com C grupos, onde C inicialmente representa o número de amostras a serem agrupadas. Em seguida, um conjunto de operações une estes grupos culminando em um grupo único com todos os elementos. O método divisivo se comporta de maneira oposta, começando com um único grupo com todos os elementos e realizando divisões sucessivas, até que todos os grupos sejam de um único elemento. Devido ao seu alto custo computacional, os métodos divisivos são pouco utilizados. Os grupos hierárquicos são comumente visualizados através de árvores binárias ou dendrogramas, que possibilitam a percepção das relações de vizinhança e hierarquia entre os dados. Esta característica é particularmente interessante, já que é comum a dimensionalidade dos dados ultrapassar três dimensões. Os algoritmos tradicionais de agrupamento hierárquico possuem complexidade temporal elevada, de $O(n^2)$, onde n representa o número de amostras, mas existem variações deste algoritmo que diminuem esta complexidade (Xu e Wunsch 2005) assim como os efeitos de algumas restrições como sensibilidade a ruídos e amostras distantes (*outliers*);
- Agrupamento por partição: divide as amostras em grupos pré-definidos sem estrutura hierárquica. Como o número de partições criadas pode ser muito grande, encontrar a melhor delas torna-se um processo custoso utilizando o método de força bruta. Para endereçar esta restrição, o conhecimento prévio do problema ou técnicas de heurística são utilizadas para direcionar a quantidade de grupos, de modo a evitar esforços desnecessários. Uma característica importante dos métodos de particionamento é a utilização de uma função critério, que tem a finalidade de definir o critério de parada da execução do algoritmo de agrupamento. A mais comumente utilizada é a função quadrática de erro, a que mede basicamente a diferença entre as amostras e o centro de cada grupo, fornecendo um valor de erro que pode ser considerado ou não aceitável de acordo com a condição de parada do algoritmo. O algoritmo de particionamento mais conhecido que utiliza o erro quadrático é o k-means (Forgy 1965). Este algoritmo é de simples implementação, resolve diversos problemas práticos e tem complexidade de $O(Nk)$, onde N é o número de amostras e k o número de grupos, o que o torna aceitável para grandes volumes de dados. Em contrapartida, ele apresenta algumas restrições, como a dificuldade de identificação de partições iniciais e número de grupos; a dependência do ponto inicial na convergência; a não existência de garantia de convergência para um ótimo global e a sua sensibilidade a ruídos. Estas restrições estimularam o desenvolvimento de algoritmos que atenuam os efeitos da grande quantidade de amostras, sua sensibilidade a ruídos, amostras distantes e as dificuldades em encontrar um ótimo global (Xu e Wunsch 2005).

O trabalho aqui desenvolvido coloca sua contribuição na medida em que propõe a utilizar técnicas de agrupamento para apoio no direcionamento da busca de casos, evitando a procura desnecessária por toda a base de casos. Assim como é importante atender ao quesito de otimização da busca, torna-se igualmente relevante entender e considerar o domínio da aplicação e o problema de RBC a ser resolvido. Esta composição permitirá a escolha de um método que tenha maior probabilidade de encontrar o melhor caso da base de casos em tempo aceitável.

Para conciliar estes interesses, o RBC será considerado dentro do procedimento de formação dos grupos (Xu e Wunsch 2005) composto pelas seguintes etapas:

- Seleção e extração de características: as seleções das características ou atributos para o agrupamento devem ser compatíveis com as escolhidas da representação dos casos, senão poderá haver distorções que recuperarão casos inadequados. A extração de características com base nos atributos explícitos, ou superficiais, é utilizada tanto em RBC como em procedimentos de agrupamento com o objetivo de representar e recuperar o melhor caso e, também, de melhorar a distinção entre os padrões e diminuir a exposição a ruídos, respectivamente. A remoção de casos antes da execução dos procedimentos de RBC e de agrupamento propriamente ditos não é recomendada pois, dependendo do processo utilizado, pode haver eliminação de casos relevantes da base de casos que inviabilizariam a recuperação de casos úteis na resolução de diversos problemas;
- Criação ou seleção de algoritmo de agrupamento: a criação ou escolha do algoritmo de agrupamento está ligada à medida de proximidade (ou afinidade no caso de SIA), pois esta medida afeta a formação dos grupos e geralmente faz parte da natureza do algoritmo selecionado. É importante lembrar que cada problema junto ao seu domínio de aplicação deve ter, de preferência, um método e algoritmo especificamente selecionados, já que não existe uma solução universal para todos os problemas de agrupamentos. Isto é particularmente verdade no RBC, já que existem situações nas quais os padrões de casos são conhecidos, a relação entre a quantidade atual de casos da base e dos casos futuros pode ser estimada, o dinamismo da aplicação, previsto e assim por diante, o que culmina em um conhecimento mais profundo do problema. Este conjunto de conhecimento possibilita uma escolha mais adequada do algoritmo e seus parâmetros, resultando na diminuição do tempo de *design* da solução e ciclos de execuções do algoritmo para ajustes que permitam encontrar um conjunto satisfatório de resultados. Por exemplo, se a base de casos é muito grande, torna-se custoso escolher um algoritmo hierárquico devido ao seu grau de complexidade, ao mesmo tempo em que, se o número de grupos é conhecido ou razoavelmente estimado, é melhor utilizar um algoritmo de agrupamentos por particionamento;

- **Validação:** a validação consta na proposição de critérios para que os resultados obtidos pelos algoritmos de agrupamento possam ser julgados para utilização ou não na solução dos problemas de RBC. Estes critérios devem ser objetivos e independentes de qualquer algoritmo. Para efeito de uso no RBC, critérios como eficiência e competência são interessantes (Smyth e McKenna 1999), pois cobrem o desempenho da busca e a relevância da solução do caso recuperado em relação ao novo caso apresentado;
- **Interpretação dos resultados:** o grande objetivo das técnicas de agrupamento é fornecer informações importantes dos agrupamentos dos dados para apoiar a resolução de problemas. Para o RBC, os grupos devem representar as diferenças reais entre os tipos de problemas existentes para que os processos de busca e adaptação sejam baseados em agrupamentos confiáveis de casos.

4.3.2. SIA e o Ciclo do Raciocínio Baseado em Casos

A abordagem de Hunt, Cooke & Holstein (1995) usa, de maneira geral, o paradigma da rede imunológica para estabelecer a memória e criar uma relação automática de proximidade e similaridade entre os casos. Tal abordagem é baseada em uma representação específica da rede imunológica, pois esta possui características interessantes, com aplicações no RBC que podem ser exploradas.

Devido a seus atributos, a aiNet, baseada na rede imunológica biológica de Jerne, será o modelo de rede imune artificial utilizado neste trabalho. Ela foi escolhida pelos seguintes motivos:

- a criação da rede imunológica é realizada através da leitura dos antígenos presentes no espaço de formas, obtendo como resultado um padrão de distribuição de anticorpos muito próximo aos antígenos. Esta distribuição cobre a diversidade dos antígenos;
- A quantidade de anticorpos criados no processo é geralmente bem menor que a quantidade de antígenos, o que representa uma compressão da distribuição dos antígenos, o que é útil em diversas aplicações;
- Como resultado do mapeamento dos antígenos, é possível detectar áreas com características específicas na rede imunológica e delimitá-las através de técnicas de agrupamento (*clustering*).

Segundo de Castro (2001) “As principais aplicações da aiNet incluem aprendizagem de máquina, reconhecimento de padrões, compressão e agrupamento de dados.”

Nesta nova proposta, os casos armazenados na base de casos são inicialmente considerados como os antígenos. Em um primeiro momento, o

algoritmo da aiNet é executado com o objetivo de gerar anticorpos que mapeiem a distribuição dos antígenos no espaço, de tal modo que estes anticorpos identifiquem de maneira compacta, ou resumida, as áreas ocupadas pelos antígenos, funcionando como seus índices. Estes índices recém criados possuem somente as informações necessárias para referenciar uma determinada região do espaço e não formam um caso propriamente dito. Em um segundo momento, um novo caso apresentado ao sistema pode ser inserido na base de casos para que seja armazenado próximo aos casos mais semelhantes ou úteis ou então simplesmente ser resgatado como resultado de uma consulta à base de casos para encontrar o caso mais adequado de acordo com critério de busca. É importante frisar que a consulta e o armazenamento são executados utilizando os índices gerados no primeiro momento. A Figura 4.2a ilustra os antígenos no espaço e a Figura 4.2b o resultado da execução da aiNet com os anticorpos e seu centro no mesmo espaço dos antígenos. Esta abordagem da aiNet traz algumas vantagens em relação à precursora (Hunt, Cooke e Holstein 1995):

- Otimização do processo de busca e recuperação de casos. A busca do caso (ou casos) mais semelhante ou útil é realizada usando os anticorpos como índices, pois eles fazem referência a uma região do espaço próxima a casos reais, representados pelos antígenos. Ao invés de buscar por toda a base de casos para encontrar o melhor caso, a busca é inicialmente realizada através dos índices, que são geralmente em quantidade bem menor que todos os casos da base de casos. Ao se encontrar o melhor índice, uma segunda busca, agora local, é realizada na vizinhança do índice para encontrar o melhor caso em relação ao novo caso;
- A segmentação do espaço direcionada através dos índices permite um aumento significativo do determinismo da busca, antes guiada exclusivamente pela escolha de pontos aleatórios em (Hunt, Cooke e Holstein 1995) na rede de casos. Como a busca aleatória também se desdobrava na exploração da vizinhança dos casos mais similares encontrados, esta segunda busca não necessariamente retornava os casos mais similares da base de casos.

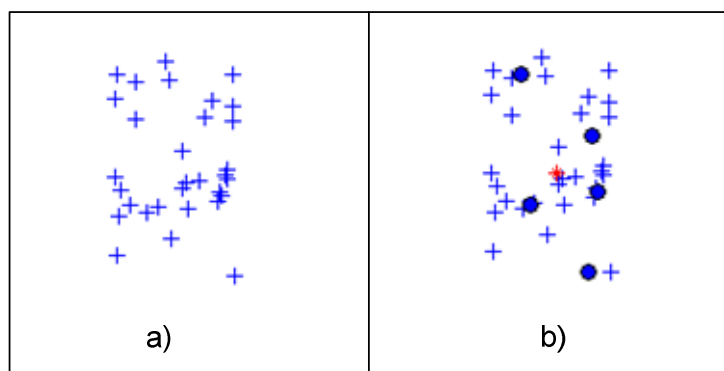


Figura 4.2 – (a) Antígenos distribuídos no espaço. (b) Os anticorpos, círculos, se posicionam em distribuição muito semelhante aos antígenos. O asterisco de cor vermelha representa o centro do agrupamento.

Na nova proposta, a implementação da afinidade da aiNet é calculada através de uma medida de similaridade baseada na distância Euclidiana, Equação (2.1), pois esta facilita a aproximação com formas de vizinhanças hipersféricas e de valores reais, sendo muito usadas em problemas do mundo real. Nada impede, entretanto, que uma medida diferente de afinidade seja adotada para se adaptar ao problema proposto.

A execução da aiNet, neste trabalho, é parte de um processo maior, que envolve técnicas de agrupamento (de Castro 2001) que possibilitam o delineamento de regiões do espaço ocupadas pelos casos, onde:

a aiNet pode ser vista como um sistema de pré-processamento para as técnicas de análise de grupos, tornando-se uma ferramenta poderosa para a filtragem e pré-processamento de dados de um conjunto amostral.

É importante ressaltar que a aiNet não concorre com nenhuma técnica de grupos, mas tem a função de mapear o espaço com índices para que técnicas e algoritmos de agrupamento se utilizem destes índices como dados de entrada para a detecção e criação dos grupos.

Se nenhuma técnica de agrupamento for utilizada, todos os índices, ou anticorpos, serão analisados, o que pode não ser eficiente em determinadas aplicações, pois a consulta em uma base de casos com grande quantidade de índices pode gerar latência na busca pelo melhor caso. Desta maneira, além de fazer uso dos índices, ou anticorpos da aiNet (distribuídos pelo espaço para aumentar a eficácia da busca), estes índices são agrupados, permitindo, assim, que seja realizada uma busca em etapas e aumentando mais ainda a segmentação da busca (Smyth e McKenna 1999) (Patterson, et al. 2002).

Com os índices representados pelos centros e pelos anticorpos, após a execução da aiNet em um primeiro momento, uma pesquisa na base de casos pode ser realizada com maior eficiência. Desta maneira, os casos avaliados serão em quantidade bem menor, se comparados a buscas completas ou aleatórias, pois estas últimas podem percorrer regiões do espaço que não tem nenhuma relação com o novo caso. A partir do tratamento agrupado da base de casos, torna-se possível trabalhar com uma idéia mais ampla na sua

adaptação. Assim, um caso mais favorável pode ser procurado em regiões fronteiriças de grupos vizinhos, ao invés de trabalhar a adaptação no grupo inicialmente eleito pelo processo de busca.

Ao final dos processos de pesquisa, agrupamento e adaptação, o caso pode ser armazenado na base de casos. Estes processos culminam com o reposicionamento dos anticorpos no grupo e a recriação das relações de vizinhança entre os anticorpos e antígenos.

A Figura 4.3 representa o diagrama de blocos da utilização da aiNet aplicada ao RBC, de tal modo que enderece as melhorias citadas anteriormente em relação ao trabalho de Hunt, Cooke & Holstein (1995). Para permitir uma apresentação mais didática e completa, a proposta será contextualizada com os processos de RBC, aproximando-se o máximo possível deles e explicitando-se a relação entre RBC e SIA através de uma abordagem algorítmica.

Como no modelo proposto os casos são sinônimos de antígenos e índices de anticorpos, será adotado um padrão da terminologia neste capítulo. Os termos índices e casos serão usados quando o contexto for relacionado ao RBC e anticorpo e antígeno quando houver correlação à biologia ou algum algoritmo imune inspirado.

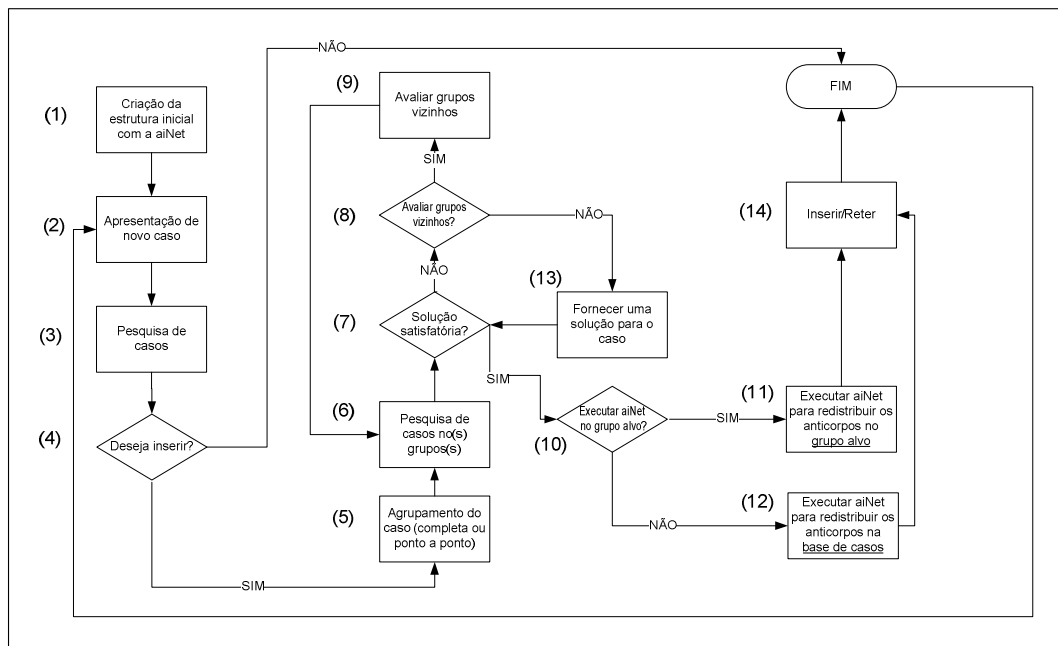


Figura 4.3 – Diagrama de blocos da aiNet para uso em conjunto com o RBC.

4.3.2.1. Recuperação

O objetivo deste processo de recuperação é possibilitar a busca para retornar o caso que tenha a melhor solução de uma maneira mais eficaz que o método de busca e inserção aleatório citado em (Hunt, Cooke e Holstein 1995).

Para criar uma ação mais integrada, a busca será segmentada em duas etapas, uma para a seleção do grupo onde o caso se encontra e outra para direcionar a busca pelos índices dentro do próprio grupo. Esta busca em duas etapas elimina a necessidade de análise de todos os grupos e de todos os casos de um grupo.

A ordem numérica dos itens a seguir é a mesma do diagrama de blocos da Figura 4.3, salvo quando explicitamente indicado o contrário. Por motivos didáticos, o passo 11 será incluído como parte do processo de busca, embora seja, na verdade, parte do processo de inserção, já que este contempla a modificação de índices, aqui representados pela redistribuição dos anticorpos no grupo alvo, antes da inserção propriamente dita.

A recuperação e todos os demais processos do RBC com SIA podem ser vistos como desdobramento de um estágio preparatório anterior, descrito pelo passo 1:

1. Criação da estrutura inicial: responsável pelo posicionamento dos anticorpos, como resultado da execução da aiNet. A aiNet deverá inicialmente, ler todos os casos, disponíveis na base de casos e gerar índices, levando em conta a afinidade entre anticorpos e antígenos, o que tem um custo computacional de $O(m^2)$, onde m representa o número de anticorpos de memória (de Castro 2001). Esta complexidade quadrática indica que quanto menor for a geração de anticorpos, menor será a complexidade computacional do algoritmo. Com base nesta colocação, torna-se relevante minimizar a execução desnecessária da aiNet. Ainda neste estágio preparatório, serão criados grupos de casos baseados nos índices, segundo algoritmo específico, dependendo das características de cada aplicação. Após a geração dos grupos, deve-se calcular os seus centros, a proximidade entre eles e, em seguida, realizar o levantamento da vizinhança de cada índice dentro de cada grupo para detectar os casos mais similares no seu entorno. É importante ressaltar que a execução do algoritmo de agrupamento baseia-se na análise dos anticorpos gerados na aiNet, já que são uma representação compacta dos antígenos. O resultado final é um conjunto de grupos, com seus índices referenciando um conjunto específico de casos dentro de cada grupo, ou seja, uma rede de casos inspirada no sistema imune que fornece informações relevantes sobre a distribuição dos casos e suas características;

A recuperação propriamente dita é iniciada com a apresentação de um novo caso e desenvolvida com a decisão da inserção do novo caso, como descrito a seguir:

2. Apresentação de novo caso: um novo caso é apresentado ao sistema para pesquisa, já contando com a existência de grupos, seus centros e índices criados no estágio preparatório, ou passo 1;
3. Pesquisa de casos: este passo inicia a busca do melhor caso em duas etapas, sendo que a primeira consiste em, sempre que possível, utilizar a informação dos centros gerados no estágio preparatório, evitando

assim uma análise exaustiva de todos os grupos da base de casos. Se, por alguma restrição da aplicação, não for possível utilizar a segmentação da busca através dos centros, pode-se utilizar somente os índices, criados no passo 1. Esta segunda etapa da busca baseada nos índices consiste em uma busca mais específica pelo caso mais relevante dentro do grupo selecionado através da utilização dos índices, segmentando mais ainda o processo de busca. O resultado desta etapa é a recuperação do caso mais relevante no entorno do índice mais relevante. Estes dois tipos de buscas podem ser usados de forma combinada ou independente e o resultado é a recuperação de um caso, ou casos, que tenha uma solução adequada para o caso apresentado;

4. Decisão de inserção: se houver interesse em inserir o caso após a consulta, um processo de agrupamento deverá ser executado para acomodar o caso e permitir que buscas futuras sejam beneficiadas pela estrutura segmentada dos casos da base de casos. Se não houver interesse de inserção do caso, o processo é reiniciado com a apresentação de um novo caso;
5. Agrupamento do caso: após a decisão afirmativa da inserção, o novo caso é analisado pelo algoritmo de agrupamento que, em linhas gerais, poderá efetuar a classificação do caso em um grupo já existente (que pode gerar a união de grupos próximos, caso seja percebida esta necessidade pelo algoritmo) ou criar um novo grupo, se as características do novo caso forem muito diferentes dos casos armazenados nos grupos existentes. Este processo de agrupamento poderá ser realizado, dependendo das restrições do algoritmo de agrupamento e da aplicação. Por exemplo, se a aiNet tiver de ser executada em toda a base de casos, o agrupamento será realizado levando em conta toda a base de casos, pois os anticorpos foram completamente reposicionados. Em contrapartida, se não for necessário executar a aiNet em toda a base de casos e ela puder ser separada em grupos, o processo de agrupamento será executado como um aprendizado ponto a ponto e a aiNet será executada somente no grupo indicado pelo processo de agrupamento para que os anticorpos cubram a distribuição, já considerando o novo caso no grupo em questão;
6. Encontrar o melhor caso no grupo: após a decisão de como agir em relação ao agrupamento (passo 5), será realizada uma busca local no grupo ou grupos selecionados e o melhor caso, ou casos, serão recuperados para avaliação. Neste sentido, a seleção do caso mais relevante pode se dar através de um limiar de relevância, sendo que casos acima de um determinado limiar (inverso da distância euclidiana, por exemplo), seriam candidatos a fornecer uma solução direta, ou sem adaptação, e o que possuísse o maior valor seria o mais apto. Caso nenhum obedeça ao critério, uma nova seleção deverá ser fornecida;
7. Decisão de aceitação: se a solução do caso ou conjunto de casos recuperados for satisfatória, os anticorpos serão reposicionados no grupo do passo 6 através da aiNet. A execução da aiNet no grupo é tratada no passo 11. Se a solução for insatisfatória, pode-se prosseguir

com a pesquisa em grupos vizinhos, optar por manter o caso no grupo pesquisado como um caso diferente em relação aos demais do mesmo grupo ou criar um novo grupo para este caso;

8. Decisão de avaliação de grupos vizinhos: se necessário, pode-se consultar grupos vizinhos ao inicialmente selecionado para encontrar uma melhor solução ao caso apresentado. Se positivo, os vizinhos são selecionados e a pesquisa é realizada. Se negativo, é avaliada a necessidade de criação de um grupo para acomodar o caso apresentado;
9. Avaliação de grupos vizinhos: este passo consiste no levantamento dos grupos vizinhos em relação aos da pesquisa recém realizada. Neste caso, o levantamento da vizinhança obedecerá a um critério de proximidade entre os grupos, como, por exemplo, escolha dos n grupos mais próximos, onde n é um número fornecido de grupos vizinhos; escolha dos grupos vizinhos que sejam mais próximo do grupo em questão do que em relação a qualquer outro grupo ou grupos que estejam a uma distância menor que a distância média entre todos os grupos. O critério utilizado pode ser mais bem definido, conhecendo-se a aplicação e o algoritmo de agrupamento. A avaliação de um grupo vizinho pode ser interpretada como uma adaptação de alto nível (que será tratada em um tópico específico), ou genérica, pois esta não avalia individualmente os casos, mas considera os grupos mais prováveis para que o caso, ou casos mais relevantes sejam encontrados. A partir da seleção dos grupos vizinhos, novas buscas são iniciadas, retornando ao passo 6 da Figura 4.3;
10. Decisão de executar aiNet no grupo alvo: se o resultado da busca no grupo selecionado não for satisfatório, uma pesquisa em grupos vizinhos seria necessária e, se um caso satisfatório for encontrado, a aiNet será executada nesse grupo, já que o caso apresentado fará parte deste grupo e os anticorpos deverão ser redistribuídos para cobrirem o novo caso. Em contrapartida, se for necessária a execução completa (depende da aplicação e algoritmo de agrupamento usado) da aiNet na base de casos (já considerando o novo caso) e não exclusivamente em um grupo específico, seguida do processo de agrupamento, o resultado será suficiente para classificar o novo caso em um grupo que apresente uma solução satisfatória, mas ao custo de executar a aiNet em toda a base de casos;
11. Execução da aiNet no grupo alvo: após uma decisão positiva nos itens 7 e 10, a aiNet será executada de maneira incremental, ou seja, somente no grupo no qual o caso será inserido. Os anticorpos são reposicionados, considerando-se o novo caso e, desta maneira, todos os casos do grupo estarão cobertos pelos índices. Ao final, o caso pode ser inserido na base de casos pelo processo de inserção (que será tratada em um tópico específico). A vantagem deste procedimento é que não será necessária uma execução completa da aiNet na base de casos;

12. Execução da aiNet na base de casos: após uma decisão negativa no item 10, a aiNet será executada em toda a base de casos, o que é justificável para domínios e algoritmos de agrupamento que apresentem restrições à execução incremental da aiNet;

Com estas ações e dependendo do algoritmo de agrupamento, evita-se uma busca descoordenada e em toda a base de casos, pois a maioria dos grupos não é utilizada nestes processos. O resultado desta etapa de recuperação de um caso, ou casos, é a percepção da necessidade de adaptação ou a preparação para inserção do caso em um grupo, recém criado ou existente. Sempre que possível e após o estágio preparatório, considerando a natureza do algoritmo de agrupamento e as restrições do problema, a aiNet deve ser executada somente no grupo alvo. Isto significa que os grupos de casos que não têm relação com o novo caso não terão o seu centro alterado e, conseqüentemente, manterão a mesma distribuição interna dos seus anticorpos.

Como pode-se perceber, esta proposta implica na necessidade de utilização de técnicas de agrupamento que irão inserir os casos em grupos de casos de acordo com suas características.

São propostos aqui algoritmos que possibilitem a criação da estrutura inicial da aiNet e da busca em uma base de casos.

Considera-se que:

- A aiNet já foi apresentada anteriormente no Capítulo 3 e seu algoritmo é conhecido;
- Os índices são formados pelos anticorpos resultantes da execução da aiNet na base de casos;
- Para a aiNet, a base de casos e os novos casos apresentados são considerados como antígenos.

Não serão definidos e utilizados aqui algoritmos específicos de agrupamento, pois estes são escolhidos dependendo do domínio e do problema a ser resolvido.

Algoritmo que cria a rede inicial de casos baseada na aiNet

1. Executar a aiNet nos antígenos (base de casos);
2. Agrupar anticorpos gerados na aiNet;
3. Calcular os centros de cada grupo;
4. Classificar os antígenos com base nos agrupamentos de anticorpos recém criados;
5. **Para** cada grupo $i = 1, \dots, C$
 - 5.1. **Para** cada anticorpo do grupo i
 - 5.1.1. Selecionar os antígenos, ou casos da sua vizinhança de acordo com o critério de vizinhança;
 - 5.2. **Fim Para**

6. Fim Para

Algoritmo 4.3 – Executa a aiNet e cria grupos com os casos presentes na base de casos.

A numeração aqui citada diz respeito aos passos do Algoritmo 4.3, salvo quando explicitamente indicado o contrário. O Algoritmo 4.3 utiliza os antígenos para gerar o repertório de anticorpos através da aiNet no passo 1. O resultado é o mapeamento da distribuição dos antígenos e uma representação comprimida da base de casos pelos anticorpos. Em seguida, no passo 2, os anticorpos são usados na criação dos grupos, lembrando que cada domínio/problema terá um método de agrupamento específico que atenderá às suas necessidades. A partir da criação dos grupos de anticorpos, os seus centros são calculados no passo 3, pois esta é uma informação importante que pode ser utilizada na segmentação da busca, cálculo da pertinência dos anticorpos e, também, para a futura classificação dos antígenos. A partir da geração dos centros, os antígenos podem ser classificados no passo 4 dentre os grupos gerados no passo 2. Esta classificação pode ser realizada a partir de diversas técnicas (como k-NN, por exemplo, apresentado no Capítulo 2), desde que estas não utilizem remoções de casos para melhorar a precisão da classificação, já que a existência de casos é primordial para o RBC e a sua eliminação deve ser realizada de maneira criteriosa. Na linha 5.1.1, é criada a relação de vizinhança entre os índices e os casos da base de casos. O princípio é que a relação se traduza na relevância ou utilidade real entre os casos e não seja o resultado de um critério que desconsidere a especificidade do problema. Ao final da execução deste algoritmo, os grupos e a rede de casos estarão criados.

Algoritmo de busca de casos baseado na aiNet

1. **Para** cada novo caso apresentado
 - 1.1. Efetuar a busca pelo grupo mais relevante, utilizando o centro de todos os grupos;
 - 1.2. Efetuar a busca pelo caso mais relevante dentro do grupo selecionado na busca 1.1, utilizando os índices deste grupo;
 - 1.3. **Se** inserção necessária **então** (será executado o agrupamento do novo caso)
 - 1.3.1. **Se** houver necessidade de execução da aiNet na base de casos completa (pela natureza do algoritmo de agrupamento ou aplicação) **então**
 - 1.3.1.1. Executar Algoritmo 4.3;
 - 1.3.2. **Senão**
 - 1.3.2.1. Realizar o agrupamento considerando somente o novo caso como dado de entrada (aprendizado ponto a ponto) e o centro dos anticorpos dos grupos; resultando na decisão de: inserir o caso em grupo existente ou criar um novo grupo;
 - 1.3.3. **Fim Se**
 - 1.3.4. NecessidadeDeBusca = 1;
 - 1.3.5. **Enquanto** NecessidadeDeBusca
 - 1.3.5.1. Efetuar busca pelo caso mais relevante dentro do grupo indicado;

- 1.3.5.2. **Se** a solução encontrada for satisfatória **então**
 - 1.3.5.2.1. NecessidadeDeBusca = 0;
 - 1.3.5.2.2. **Se** for necessário executar a aiNet no grupo alvo **então**
 - 1.3.5.2.2.1. Executar a aiNet nos antígenos do grupo do novo caso apresentado;
 - 1.3.5.2.3. **Senão**
 - 1.3.5.2.3.1. Executar a aiNet em toda a base de casos.
 - 1.3.5.2.4. **Fim Se**
 - 1.3.5.2.5. Inserir o caso na base de casos;
- 1.3.5.3. **Senão**
 - 1.3.5.3.1. **Se** necessário avaliar grupos vizinhos **então**
 - 1.3.5.3.1.1. Realizar levantamento de grupos vizinhos;
 - 1.3.5.3.1.2. Avaliar e selecionar grupo(s) vizinho(s) para busca;
 - 1.3.5.3.1.3. Efetuar busca pelo caso mais relevante dentro do(s) grupo(s) indicado(s);
 - 1.3.5.3.2. **Senão**
 - 1.3.5.3.2.1. Fornecer uma solução para o caso. Esta solução pode ser derivada de um processo de adaptação de caso ou de um especialista ou modelo simplificado do domínio;
 - 1.3.5.3.3. **Fim Se**
- 1.3.5.4. **Fim Se**
- 1.3.6. **Fim Enquanto**
- 1.4. **Fim Se**
- 2. **Fim Para**

Algoritmo 4.4 – Explora a base de casos à procura do grupo e caso mais semelhante que possa oferecer a solução para o caso apresentado.

Conhecendo-se a importância do agrupamento no processo de busca, torna-se pertinente reforçar que cada problema ou domínio de RBC demandará um tipo de algoritmo de agrupamento, devido à sua natureza, restrições e necessidades.

Por exemplo, em um problema de detecção de falhas em tempo real, onde se conhece somente alguns pontos da operação normal, os grupos de falhas e até mesmo de operação normal são dinamicamente criados. Assim, a atividade de encontrar um caso mais similar na mesma classe do problema apresentado demandará uma determinada precisão da atividade de classificação, logo o processo de agrupamento e a busca devem ser bem projetados para suprir esta demanda. Para esta situação, torna-se interessante selecionar algoritmos de agrupamento de baixa complexidade, que tratem agrupamentos de quantidade variável e que, ao mesmo tempo permitam agrupar as falhas, ou casos, com uma margem de erro aceitável com base no comportamento do fenômeno e objetivos do sistema projetado.

Antes da busca e inserção propriamente dita de um caso, seja no exemplo de falhas ou qualquer outro, é necessário executar a aiNet para gerar os anticorpos que serão utilizados pelo algoritmo de Algoritmo 4.3, sendo este já previamente selecionado de acordo com o problema.

A partir deste ponto, o Algoritmo 4.4 pode ser executado. No início deste algoritmo, um novo caso é apresentado, seguido da busca de um caso, ou casos, que possam resolver satisfatoriamente o caso apresentado. O processo de busca é realizado através dos centros dos agrupamentos existentes em combinação com a busca dentro do grupo selecionado, utilizando os índices deste grupo. A conformação dos anticorpos deve apoiar o processo de busca, possibilitando uma triagem através dos grupos, permitindo a rápida separação dos grupos relevantes dos irrelevantes. O resultado é a recuperação de um caso, ou casos, que levem à decisão de inserir, ou não, o novo caso em um grupo existente.

Com a decisão afirmativa de inserir o caso, torna-se necessário que o algoritmo de agrupamento seja capaz de tratar uma situação de leitura ponto a ponto, como indicado na linha 1.3.2 do Algoritmo 4.4, pois o esforço computacional é bem menor, se comparado à execução da aiNet em toda a base de casos. Desta maneira, a execução da aiNet fica restrita ao grupo no qual o caso for classificado, ao que chamaremos aqui de execução incremental da aiNet. Para que a aiNet seja executada em modo incremental, o ideal é que algum algoritmo de agrupamento suporte diretamente a apresentação ponto a ponto dos casos ou possa ser adaptado para este fim. Caso contrário, poderá ser feita uma execução da aiNet em toda a base de casos, como em 1.3.1 do Algoritmo 4.4, realizando, na sequência, um agrupamento completo na base de casos, incluindo o novo caso, o que terá um custo computacional significativamente maior em relação à execução ponto a ponto. Ao conciliar o algoritmo de agrupamento com a aiNet, o posicionamento dos anticorpos será atualizado somente no grupo alvo, evitando a execução desnecessária da aiNet em todos os grupos.

É importante ressaltar que a execução da aiNet não tem o objetivo de interferir no funcionamento dos algoritmos de agrupamento, mas sim de fornecer pontos relevantes no espaço, sob a forma de anticorpos, para que o processo natural do algoritmo de agrupamento seja executado sobre estes anticorpos, já que eles são uma representação compacta dos antígenos. Para que seja possível uma execução incremental da aiNet com algoritmos de agrupamento, ajustes podem ser necessários nestes algoritmos, ao passo que outros podem não permitir tal intervenção, necessitando assim que a aiNet seja executada em toda a base de casos.

Como exemplo de algoritmos que não suportam diretamente a execução incremental da aiNet com uma apresentação ponto a ponto estão os algoritmos de agrupamento hierárquicos *single linkage* e *complete linkage* (Xu e Wunsch 2005). Estes algoritmos lêem todos os dados de uma vez, mas, com ajustes, torna-se possível tratar os novos casos ponto a ponto para a realização de um processo de agrupamento, no qual a aiNet é executada somente no grupo em que o novo caso foi classificado.

Na eventual execução de um agrupamento completo (1.3.1 do Algoritmo 4.4) ou na geração da estrutura inicial, Algoritmo 4.3, todos os casos da base de casos são lidos, os anticorpos gerados pela aiNet, e os antígenos são posteriormente classificados nos grupos já definidos e formados pelo algoritmo de agrupamento. Esta é, basicamente, uma tarefa de classificação onde as classes são conhecidas, visto que os grupos foram definidos utilizando os anticorpos. Assim, a classificação dos antígenos pode ser feita de diversas maneiras, o que inclui técnicas como o uso do algoritmo K-NN, o centro de cada grupo ou suas variações, ficando a definição a critério da característica de cada problema. É interessante ressaltar a classificação dos casos utilizando os anticorpos, pois como eles representam uma versão comprimida da distribuição dos antígenos, a classificação de um caso é realizada com menos iterações, o que diminui a complexidade computacional. Além disto, os anticorpos representam, de forma explícita, um mapeamento mais claro dos antígenos, atenuando interferências causadas por ruídos se comparadas a uma classificação realizada exclusivamente baseada nos antígenos. Um trabalho (Garain 2008) demonstra como foi possível diminuir a quantidade de amostras armazenadas no banco de uma aplicação de reconhecimento de escrita, onde os anticorpos respondem pela representação compacta das amostras através do processo de seleção clonal (empregado implicitamente na aiNet). Além da vantagem da diminuição do número de amostras no armazenamento e, conseqüentemente, no tempo das consultas houve aumento na precisão, se comparada com técnicas como NN e CNN.

Em ambas as situações de agrupamento, completo e incremental, a execução da aiNet gera os anticorpos, diferenciando-se basicamente no desempenho, ou seja, ou eles são gerados novamente, utilizando toda a base de casos ou somente os casos pertencentes a um grupo (no qual o caso foi classificado). Com a execução da aiNet, os anticorpos terão uma nova conformação, que permitirá o cálculo dos centros de cada grupo, da vizinhança entre os seus índices e casos e, também, da vizinhança entre grupos.

4.3.2.2. Adaptação (Reutilização e Revisão)

Após o processo de recuperação, os dois próximos processos do ciclo de vida do RBC são a reutilização e a revisão. A reutilização corresponde à análise das diferenças entre o caso recuperado e o novo caso e à tomada de decisão de qual parte do caso recuperado pode ser transferida para o novo caso. A revisão é o processo de avaliação da solução fornecida no processo de reutilização, sendo que, se a solução não for adequada, ela deve ser reparada e, se for adequada, já pode ser diretamente transferida para o novo caso e armazenada.

Devido à dificuldade de distinção entre os processos de reutilização e revisão em aplicações práticas, os dois serão tratados como um único, denominado adaptação (Pal e Shiu 2004). É importante ressaltar que a adaptação é um processo muitas vezes complexo, pois lida com a manipulação da solução dos casos, o que geralmente demanda a existência de um modelo causal (ou a presença de um especialista) que possibilite o direcionamento da solução, baseando-se em relações de causa e efeito entre o problema, já que a solução não estaria disponível na base de casos.

O trabalho aqui desenvolvido dividirá a adaptação em dois tipos ou níveis. A de primeiro nível será chamada de adaptação geral e a de segundo nível, de específica. A de primeiro nível consiste na busca de casos que possam produzir uma melhor solução, possivelmente evitando a adaptação da solução do caso encontrado. Esta adaptação considera o grupo como um elemento importante, adaptando o processo de busca por casos mais relevantes em grupos vizinhos ao grupo inicialmente selecionado no processo de recuperação. Esta ação se justifica devido a que, em diversas situações, os grupos de casos estão próximos, podendo até mesmo apresentar sobreposições, justificando a exploração da vizinhança. A adaptação de segundo nível, ou específica, é a adaptação do caso propriamente dito. Se a solução encontrada não for suficiente para resolver o problema apresentado pelo novo caso, ela poderá passar pelo processo de adaptação, seja ele automático ou manual. A abordagem da adaptação específica pelo SIA se dará, nesta proposta, através de um processo aleatório (Pal e Shiu 2004) sob a forma de uma mutação em componentes específicos da solução do caso mais similar encontrado. Esta mutação pode ser controlada e criará novas soluções no espaço de soluções. O critério de parada deste ciclo pode ser amparado por uma função objetivo, pela validação de um modelo simplificado de domínio do problema, pelo aval de um especialista, ou por sua validação em um ambiente controlado ou real. Esta adaptação baseada em mutação é especialmente interessante em bases de casos de baixa densidade, pois a mutação permite uma exploração mais ampla do espaço. Cabe colocar que os métodos mais tradicionais expostos no Capítulo 2, como a cópia, substituição, a transformação ou derivação também podem ser utilizados no processo de adaptação, desde que sejam condizentes com o domínio e as restrições da aplicação.

O processo de adaptação descrito acima será contextualizado na figura 4.3, onde, por motivos didáticos, a adaptação do tipo geral se deu nos passos 8 e 9 do processo de recuperação. Por sua vez, a adaptação do tipo específica se dará como parte do passo 13 da mesma figura.

A adaptação geral é realizada quando a solução encontrada na busca do caso em um grupo da base de casos não é satisfatória. A partir deste ponto, toma-se a decisão pela busca em outros grupos, iniciando-se em seguida uma procura pelos índices dos grupos vizinhos até que seja encontrado, se existir, um caso mais relevante que preferencialmente não necessite de adaptação. Se a solução for mais satisfatória, o processo caminha para o passo 10, onde será tomada a decisão de execução da aiNet, seguida pela inserção do caso. Em contrapartida, se a solução não for satisfatória, será necessária a criação ou adaptação de uma solução.

Para descrever o quadro da adaptação específica, será seguida a ordem numérica do diagrama de blocos da Figura 4.3, salvo quando explicitamente indicado o contrário:

1. Fornecer ou confirmar uma solução para um caso: este passo representa duas situações, a adaptação do caso propriamente dita e o fornecimento de uma solução através da existência de algum modelo específico do domínio, se um caso não for resolvido diretamente pela

recuperação de casos da base de casos. Este modelo de domínio pode ser acessado de forma automatizada e/ou manual para fornecer a solução ao novo caso. Se for necessário realizar uma adaptação específica, ou secundária, o caso mais próximo encontrado no processo de recuperação será submetido à adaptação, que consistirá em um processo de mutação de seus atributos da componente solução que serão avaliados por um modelo de domínio disponível no sistema RBC e/ou por um especialista. Se o resultado da adaptação não for satisfatório, o processo é reiniciado, até que se obtenha uma solução adequada.

O Algoritmo 4.5 representa a adaptação por mutação contida no passo 13 da Figura 4.3, considerando que a pesquisa nos grupos vizinhos não retornou nenhum caso satisfatório e a mutação será realizada.

Algoritmo de adaptação que utiliza a aiNet

1. **Enquanto** a solução não for aceitável
 - 1.1.1. Realizar adaptação através do processo de mutação da componente solução.
 - 1.1.2. Avaliar o resultado através do modelo específico de domínio ou de um especialista
 - 1.1.3. **Se** a solução for aceitável **então**
 - 1.1.3.1. Prosseguir para o processo de inserção;
 - 1.1.4. **Fim Se**
2. **Fim Enquanto**

Algoritmo 4.5 – Uso da mutação para explorar o espaço de soluções.

A adaptação será realizada utilizando o processo de mutação da componente solução do caso recuperado e será avaliada por um modelo específico de domínio ou por um especialista. Outros operadores genéticos como *cross-over*, por exemplo, também podem ser utilizados, variando o uso de acordo com a aplicação.

4.3.2.3. Retenção

A retenção é o processo que contempla a inserção do caso na base de casos e inclui o aprendizado sobre o caso, a modificação da estrutura de índices, se necessário, e manutenção da base de casos. Com o aumento do tamanho das bases de casos, torna-se importante considerar a manutenção da base de casos, pois, de maneira geral, otimiza-se a relevância dos casos armazenados e o desempenho no ciclo de resolução. A retenção e manutenção estão representadas pelo passo 14 da Figura 4.3.

O aprendizado do processo de inserção consiste em fornecer informações do sucesso ou insucesso na tentativa de resolução do novo caso. Este trabalho não considera o SIA como um método que atuará na geração de explicações do sucesso ou insucesso das tentativas de solução, esta tarefa é, muitas vezes, delegada a métodos já apresentados no Capítulo 2, como a utilização

de regras e modelos que mapeiam o funcionamento geral de um domínio do conhecimento em uma aplicação de RBC.

Em relação aos índices, a execução da aiNet é realizada no grupo alvo para que os anticorpos sejam reposicionados com o objetivo de mapear o novo caso no grupo. Após a recriação dos índices, a inserção pode ser realizada. A criação dos índices está exposta nos passos 11 e 12 da Figura 4.3 e, por motivos didáticos, descrita no processo de recuperação.

Neste trabalho, a inserção propriamente dita é uma tarefa de reavaliação da vizinhança no grupo alvo entre os índices e seus casos, partindo do princípio de que os índices de maior afinidade (relevância ou utilidade) devem conectar-se entre si e com os casos propriamente ditos, criando assim uma rede de casos baseada no princípio da rede imunológica artificial.

Como a proposta aqui apresentada será pautada pelo uso da distância Euclidiana, a afinidade tomará forma através da similaridade entre casos, e se dará pelo inverso desta distância, onde casos mais próximos, de menor distância entre si, apresentam características mais similares do que outros, o que geralmente conduz a soluções semelhantes. Nada impede que outras medidas de similaridade ou afinidade sejam usadas, desde que contribuam para a seleção do melhor caso e sejam passíveis de utilização com SIA.

O método de decisão que guiará o estabelecimento da vizinhança através da distância euclidiana pode tomar várias formas, como:

- Limiar pré-definido: este método estabelece uma linha de corte a partir da qual qualquer distância acima, ou abaixo (dependendo da orientação), deste limiar pré-definido gera uma conexão da rede de casos;
- Conexão completa: não estabelece uma condição, todos os índices e casos são conectados;
- Análise interna: utiliza informações como média e desvio dos casos nos grupos para que sejam geradas as conexões da rede. Outros critérios podem ser usados, desde que sejam derivados dos casos e seus índices.

Os dois primeiros métodos não são interessantes na criação de conexões, pois, na maioria das vezes, eles não contribuem para a inferência das relações entre os índices e casos do grupo analisado, considerando as características do problema. O último método é mais adequado, porque considera as particularidades dos índices e casos, já que a média e o desvio padrão das distâncias entre eles são extraídos diretamente dos casos.

O trabalho desenvolvido em (Timmis, Neal e Hunt 2000) propõe que o cálculo do limiar para criação da conexão entre os elementos da futura rede imunológica seja baseado na média das distâncias dos anticorpos, o que apresentou bons resultados. O nome deste limiar é Limiar de Afinidade da

Rede (*Network Affinity Threshold* – NAT ou LAR), e está representado na Equação (4.3).

$$NAT = A \left[\frac{\sum_{i=0}^N aff(l_i)}{N} \right] \quad (4.3)$$

onde N é o número de afinidades entre os elementos do grupo, seja entre índice-índice ou entre o par índice-caso. l_i é a afinidade propriamente dita entre dois elementos e A é uma constante escalar utilizada para controlar a conectividade no sentido de tornar o limiar mais restritivo, onde $0 \leq A \leq 1$.

De modo a tornar esta equação mais flexível, será incorporado a ela um fator relativo ao desvio padrão com o objetivo de controlar o limiar para que ele se torne menos restritivo, favorecendo distribuições menos densas de grupos de casos. A Equação (4.4) expressa esta alteração.

$$NAT - S = A \left[\frac{\sum_{i=0}^N aff(l_i)}{N} + (FS) \right] \quad (4.4)$$

Onde $NAT - S$ é o NAT com o desvio padrão, F é a quantidade de desvios padrão adequada para posicionar o limiar em um nível mais aceitável para a aplicação e S é o desvio padrão propriamente dito.

A rede de casos será formada por dois tipos de conexão, uma entre índices, e outra entre estes e os casos, criando assim duas malhas, uma que integra os índices no grupo e outra que localiza os casos mais próximos de cada índice. A primeira malha permite a navegação em alto nível por regiões de casos e, a segunda, a busca pelos casos mais próximos ao índice mais relevante encontrado. A criação da primeira malha usará a Equação (4.4), variando os parâmetros até que o resultado seja uma rede de índices representativa da distribuição dos casos. Após a criação da primeira malha, os índices se ligarão aos casos pelo critério de proximidade, ou seja, o índice que estiver mais próximo a determinado caso, terá uma conexão com ele, o que será repetido para todos os casos, criando assim a segunda malha de conexão. O resultado deste processo está ilustrado na Figura 4.4.

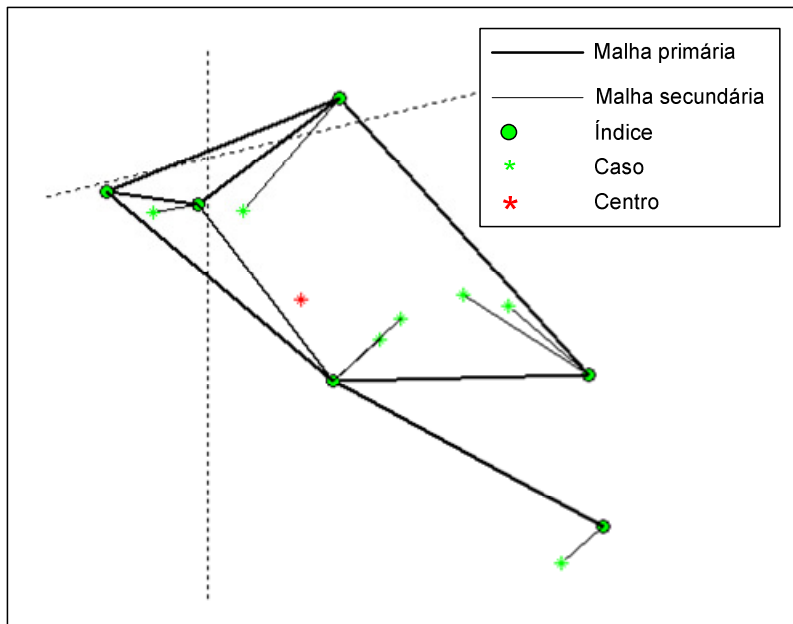


Figura 4.4 - Espaço em três dimensões onde os anticorpos são representados pelos círculos e os antígenos pelos pontos em forma de estrela. O ponto sem nenhuma conexão é o centro da distribuição dos anticorpos. As linhas tracejadas são para dar a impressão de profundidade no espaço de três dimensões.

Será utilizada a média das distâncias entre índices e entre índices e casos para que seja estabelecida a conexão, fazendo com que a busca em um grupo possa ser mais rapidamente direcionada para o caso mais relevante.

Algoritmo que cria a relação da vizinhança entre índices e casos

1. Calcular a distância Euclidiana entre todos os índices;
2. Calcular a média e o desvio padrão das distâncias medidas no passo 1;
3. Avaliar a quantidade de desvios padrão (≥ 0) que serão somados à média do passo 2 para tornar o limiar mais relaxado (dependente da aplicação);
4. Avaliar o valor A ($0 \leq A \leq 1$) que irá tornar o limiar mais restritivo (dependente da aplicação);
5. Calcular o limiar para criação de conexões entre os índices (neste está ligado diretamente à média e desvio padrão das distâncias entre os índices);
6. Criar as conexões entre os índices que possuem afinidade maior ou menor (dependendo da situação) que o limiar calculado no passo 5;
7. Criar a conexão entre o caso e o índice mais próximo.

Algoritmo 4.6 – Cria as conexões índice-índice e índice-caso.

Ao final do processo, os índices e casos estarão conectados, formando uma rede de casos baseada na rede imunológica artificial, onde os índices estarão conectados entre si como uma rede principal de referência que permeia cada grupo de casos. Por sua vez estes índices também estão ligados aos casos propriamente ditos, possibilitando encontrá-los diretamente através de uma busca direcionada pelos índices ou navegando pela estrutura de índices para explorar a sua vizinhança.

Ligado à inserção de casos e à utilização de sistemas de RBC em situações reais, surge um entrave de cunho prático: como controlar a relação entre o crescimento da base de casos e o desempenho na recuperação de casos para que se resolvam adequadamente os problemas/situações propostos(as). López de Mántaras *et al.* (2005) coloca que a manutenção da base de casos se inicia na concepção de um sistema de RBC e passa pelo uso de técnicas e ferramentas ou algoritmos de monitoração que possam determinar se, quando e como atualizar sistemas de RBC que atendam a medidas específicas de desempenho. Wilson e Leake (2001) propuseram um framework para a manutenção de casos regida pela utilização de políticas de manutenção e as dimensões onde elas poderiam ser aplicadas. Segundo estes autores, estas políticas se classificam em:

- Como coletar dados relevantes para as decisões de manutenção;
- Quando ativar as atividades de manutenção;
- Quais os tipos de operações de manutenção disponíveis;
- Como as atividades de manutenção são executadas.

Alinhado com o framework proposto, foi colocado no Capítulo 2, a separação das atividades de manutenção em duas grandes categorias, qualitativa e quantitativa (Pal e Shiu 2004). Sob esta ótica conjunta, vários aspectos do RBC são considerados na manutenção de casos (*Case-Base Maintenance* - CBM), como o controle da adição e remoção de casos da base de casos, a avaliação de índices, métodos de adaptação e medidas de similaridade.

A abordagem da utilização da rede imunológica proposta neste trabalho tratará especificamente de um aspecto qualitativo da manutenção, a precisão, e dois quantitativos, representados pela detecção de casos irrelevantes e controle do tamanho da base de casos.

A precisão responde pela capacidade de resolver a situação ou caso apresentado. Será utilizada como um indicador na identificação de casos irrelevantes na base de casos, gerando a sua posterior remoção, contribuindo para o aumento da eficiência (inverso do número de casos examinados durante a recuperação) geral do sistema.

O critério concebido para medir a precisão será o uso conjunto da medida de afinidade do sistema, com a percepção de que uma determinada quantidade de casos apresentados ao sistema possa ser resolvida por um número médio de casos recuperados para cada caso apresentado. Posto de outro modo: existe um número significativo de casos armazenados que também podem resolver o caso candidato? Se sim, é um indicador da possibilidade de existência de casos redundantes em uma determinada região da base de casos.

Se a média do número de casos recuperados que resolvem os casos candidatos for alta, significa que mais casos serão analisados e estarão aptos a

resolver o mesmo problema. Aparentemente este é um bom sinal, pois os casos candidatos podem ser resolvidos por diversos casos armazenados, mas, em compensação, também é um indicador de um possível excesso de casos que resolvam os casos candidatos, gerando latência no processo de análise e recuperação de casos. Este é o momento da ativação do processo de manutenção quantitativa, que identifica casos irrelevantes, removendo-os da base de casos, aumentando assim a eficiência dos processos do RBC.

Como as bases de casos tendem a crescer cada vez mais, torna-se importante identificar os irrelevantes e efetuar a sua remoção de maneira eficiente. Para encontrar os casos irrelevantes, este trabalho propõe identificar as áreas de maior densidade em cada grupo da base de casos, sem percorrê-la completamente. Como os índices da rede de casos de cada grupo referenciam regiões do espaço, as de alta densidade em cada grupo serão tratadas como prioritárias, já que tem maior probabilidade de possuírem elementos irrelevantes. O foco é dado na análise das áreas de maior densidade através de um levantamento da quantidade de casos que cada índice faz referência. Como a rede de casos conecta os índices com os casos, ela tem como fornecer diretamente a relação quantitativa entre índices e casos de cada grupo. Após obter a informação de número de casos indexados por cada índice em cada grupo, serão selecionados os índices que têm uma quantidade de casos indexados maior que o desvio padrão desta medida em todos os índices de cada grupo, obtendo assim, as áreas de maior densidade de casos. Após obter esta informação, bastaria, segundo métodos triviais de remoção de casos, que alguns casos fossem removidos (Patterson, et al. 2002). Apesar da aparente simplicidade, algumas perguntas devem ser feitas: como identificar os casos que resolvem problema(s) no seu entorno? Como saber, dentre estes casos, quais são irrelevantes e devem ser removidos? A resposta à primeira pergunta é que a solução de casos no modelo aqui proposto se dá quando um caso está abaixo de um determinado limiar, indicando semelhança dos problemas e, conseqüentemente, semelhança nas soluções. Como nas áreas de alta densidade um índice referencia diversos casos, cada um dos casos indexados por este índice deve verificar, na sua vizinhança, que casos ele consegue resolver.

A análise de vizinhança dos casos utilizará as conexões entre os índices presentes em cada grupo da rede de casos. Com o objetivo de explorar a vizinhança de maneira mais eficaz, cada caso dentro da influência dos índices previamente selecionados buscará um número fixo de índices mais próximos que serão explorados para encontrar possíveis casos resolvíveis, evitando assim uma busca completa na base de casos. O resultado será um conjunto de casos resolvíveis para cada caso que explorou a vizinhança.

Com base nos resultados colhidos das análises da vizinhança dos casos, devem-se avaliar quais são os casos mais irrelevantes, se houver, e executar a sua remoção. Respondendo à segunda pergunta, a identificação dos casos irrelevantes deve seguir um critério tal, que os casos mais relevantes permaneçam na base de casos, mantendo assim a competência do sistema, um indicativo da capacidade de resolver problemas (Smyth e McKenna 1999).

O princípio da competência foi introduzido no Capítulo 2 e será utilizado nesta proposta porque ele identifica de maneira segura os casos irrelevantes, ou que oferecem nenhuma ou pouca contribuição para a competência do sistema, podendo então ser removidos. Na terminologia da competência, os casos irrelevantes são chamados de auxiliares. Antes da definição formal dos casos auxiliares, serão introduzidas duas definições importantes, denominadas cobertura e alcance:

O conjunto de cobertura de um caso é o conjunto de todos os problemas alvo que este caso é capaz de resolver, enquanto o conjunto de alcance de um problema alvo é o conjunto de todos os casos que podem ser usados para resolvê-lo (Smyth e McKenna 1999).

Considerando C como um conjunto de casos e $c \in C$, o conjunto de cobertura é definido como $ConjuntoCobertura(c) = \{c' \in C: Resolve(c, c')\}$, enquanto o conjunto alcance é $ConjuntoAlcance(c) = \{c' \in C: Resolve(c', c)\}$. Isto significa que o $ConjuntoCobertura(c)$ contém os casos que c consegue resolver e o $ConjuntoAlcance(c)$ contém os casos que resolvem c .

Com a definição da cobertura e alcance de casos, os casos auxiliares podem ser definidos como: “casos que têm a sua cobertura sobreposta pela cobertura de um de seus casos alcançáveis” (Smyth e McKenna 1999).

A formalização do que é um caso auxiliar é tal que $Para c \in C, Aux(c) se c' \in Alcance(c) - \{c\}: Cobertura(c) \subsetneq Cobertura(c')$. A expressão indica que se algum caso que c resolve ($ConjuntoCobertura(c)$), conseguir resolver todos os casos resolvidos por c , então c é um caso auxiliar, ou redundante. A Figura 4.5 ilustra os casos auxiliares sob o ponto de vista da cobertura e alcance.

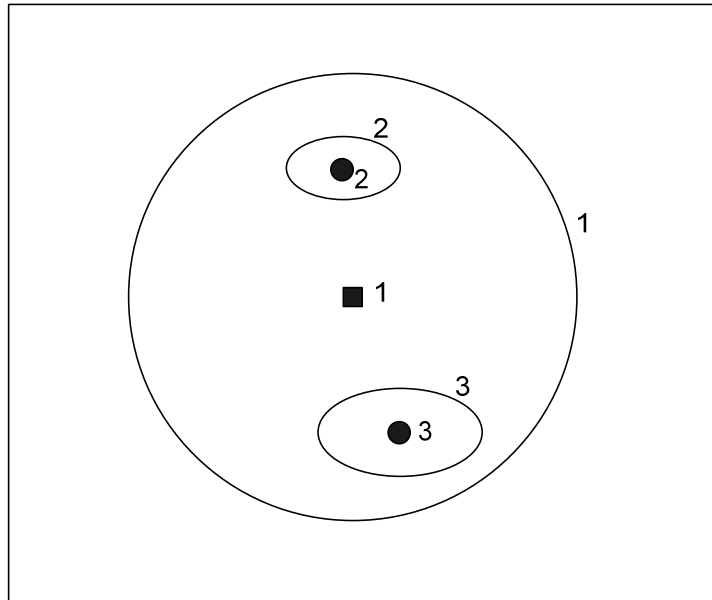


Figura 4.5 – Três casos e suas áreas de cobertura. Os casos 2 e 3 são casos auxiliares.

Os casos 2 e 3 da Figura 4.5 têm, como membros do seu conjunto de cobertura e alcance, somente eles mesmos, ao passo que o caso 1 tem em seu conjunto de cobertura ele, o caso 2 e o 3. Isto significa que os casos 2 e 3 podem ser removidos do sistema sem que haja perda da competência, pois o caso 1 resolve qualquer problema em sua região do espaço.

Os casos auxiliares tendem a se apresentar em regiões mais adensadas e geralmente menos periféricas dos grupos. Se um caso auxiliar é removido, um caso próximo à área do caso removido pode ser utilizado para fornecer a solução de um caso no entorno de tal maneira que a competência do sistema não seja afetada.

A partir do momento em que os casos auxiliares são identificados, eles podem ser removidos da base de casos. O Algoritmo 4.7 compila todo o processo de remoção de casos aqui colocado.

Algoritmo de remoção de casos baseados na aiNet

1. **Para** cada grupo $cl = 1, \dots, N_{CL}$
 - 1.1. Obter através da rede de casos o número de casos que cada índice referencia;
 - 1.2. Calcular o desvio padrão da quantidade de casos referenciados por cada índice;
 - 1.3. Selecionar os índices que tem o número de casos \geq ao desvio padrão;
 - 1.4. **Para** cada índice selecionado $i = 1, \dots, N_I$ (selecionado no passo 1.3)
 - 1.4.1. **Para** cada caso $cs = 1, \dots, N_{CS}$ (casos ligados ao índice i)
 - 1.4.1.1. Explorar os casos dos V índices vizinhos ao caso cs ;
 - 1.4.1.2. Criar o *ConjuntoCobertura* (cs) (problemas alvo - ou pa - que podem ser resolvidos pelo caso cs);
 - 1.4.1.3. **Para** cada $pa = 1, \dots, N_{PA}$
 - 1.4.1.3.1. Explorar os casos dos V índices vizinhos ao pa ;

- 1.4.1.3.2. Criar o *ConjuntoCobertura* (*pa*) para o caso *pa*;
 - 1.4.1.3.3. **Se** *ConjuntoCobertura* (*cs*) \subsetneq *ConjuntoCobertura* (*pa*) **então**
 - 1.4.1.3.3.1. *cs* entra para grupo de *CasosAuxiliares*;
 - 1.4.1.3.4. **Fim Se**
 - 1.4.1.4. **Fim Para**
 - 1.4.2. **Fim Para**
 - 1.5. **Fim Para**
 - 1.6. Remover os casos presentes no grupo de *CasosAuxiliares* e as suas conexões da rede de casos;
 - 1.7. Reexecutar a aiNet nos grupos que tiveram casos removidos;
2. **Fim Para**

Algoritmo 4.7 – Cria as conexões índice-índice e índice-caso.

O Algoritmo 4.5 trabalha para identificar as áreas mais adensadas de casos em cada grupo utilizando a rede de casos objetivando acesso direto às informações de distribuição de casos. Desta maneira, o foco é dado na remoção de casos auxiliares nestas regiões, evitando assim que a base de casos seja completamente percorrida e, ao mesmo tempo, contribuir para diminuição do tamanho da base e manutenção da competência.

É importante frisar que existe uma taxonomia de casos no tocante à sua remoção segundo o princípio da competência (Smyth e McKenna 1998). Esta categorização inclui os casos auxiliares, pois este é o único tipo de caso que, se removido, não gerará nenhuma alteração na competência do sistema. Os demais tipos de casos (pivôs, transitivos e suporte) não foram aqui colocados e estudados, porque o interesse deste trabalho na manutenção de casos é o de identificar e remover casos seguramente irrelevantes, o que não é necessariamente verdade para os demais tipos de casos, mas pode ser fruto de trabalhos futuros relativos à manutenção de sistemas RBC, utilizando o Sistema Imune Artificial.

Capítulo 5 - Experimentação e Análise de Resultados

5.1. Introdução

Após a apresentação do Raciocínio Baseado em Casos, Sistema Imune Artificial e da proposta de trabalho sinérgico entre estes dois paradigmas, no Capítulo 4, torna-se necessária a avaliação desta abordagem, com o objetivo de constatar mais claramente o grau de representatividade das suas contribuições.

Neste sentido, a avaliação se dará através da realização de experimentos conduzidos sob no domínio de detecção de falhas em uma máquina de corrente contínua. É importante frisar que a meta não é propor um método de classificação ou agrupamento ótimo para tratamento de falhas em máquinas elétricas e nem realizar um estudo comparativo, mas, sim usar este domínio para ilustrar as contribuições do SIA ao ciclo de vida de RBC e colocá-lo como uma opção em potencial para resolver problemas ou situações em domínios que demandem soluções baseadas em experiências anteriores.

O experimento é modelado sob a forma de um sistema de RBC contemplando os processos de recuperação, reutilização, revisão e retenção, utilizando propostas do SIA em cada um destes processos.

Tendo em vista o domínio de máquinas de corrente contínua, os objetivos do experimento são detectar e diagnosticar falhas que ocorrem no seu funcionamento, considerando duas modalidades e, também, avaliar a proposta conjunta de RBC e SIA com base em indicadores dos processos do ciclo de vida do RBC. Na primeira modalidade, o sistema passa por um aprendizado supervisionado onde alguns pontos são fornecidos junto com a respectiva solução. Após esta etapa de treinamento, o sistema passa a detectar e diagnosticar por conta própria. Na segunda modalidade, somente um conjunto inicial de pontos de comportamento normal é fornecido, deixando por conta do sistema a detecção do que ele considera como diferente ou novo em relação ao que é previamente conhecido. Os experimentos da segunda modalidade são realizados com uma fração da massa de dados da primeira modalidade.

Sob a ótica do RBC, são implementados novos métodos de busca, adaptação, inserção e relacionamento de casos, considerando técnicas de agrupamento e os princípios da seleção clonal e rede imunológica artificial do

SIA. Ao final, são realizadas análises e considerações com base nos resultados encontrados.

Este capítulo se divide na apresentação da motivação pelo domínio de falhas, o domínio de falhas em máquina de corrente contínua, a obtenção dos dados, modelagem do experimento e coleta e análise de resultados.

5.2. Motivação Pelo Domínio de Falhas em Máquinas Elétricas

Com o surgimento da disciplina de controle de processo, diversos processos complexos passaram por melhorias expressivas na sua regulação, devido ao aumento significativo da automação destes através do uso de computadores em tarefas antes realizadas por agentes humanos.

Apesar da diminuição da intervenção humana em muitas atividades, diversas delas ainda dependem de ações humanas, realizadas, na sua maioria, pelos operadores dos processos industriais, principalmente no tocante à resposta ou ação a eventos anormais, como falhas e desvios do processo. Estas ações envolvem a detecção dos eventos anormais, seu diagnóstico e as respostas necessárias para recolocar o processo no seu regime normal de operação. Este conjunto de atividades é chamado de Gerenciamento de Eventos Anormais (*Abnormal Event Management* – AEM ou ou GEA). Ao mesmo tempo em que houve diminuição da abrangência de atividades originalmente acompanhadas por operadores, também houve o aumento da complexidade, devido principalmente a fatores como maior diversidade dos distúrbios (falhas nas plantas, processos, desvios nos parâmetros, etc), grande número das variáveis de processo (centenas ou até milhares de variáveis observadas em intervalos de poucos segundos) e insuficiência ou incapacidade na medição destas variáveis. Com estas novas condições, que contribuem diretamente para a complexidade da tomada de decisão, as falhas humanas se tornaram um fator majoritário como causadores ou catalisadores de problemas, respondendo por aproximadamente 70% das causas de acidentes industriais (Venkatasubramanian, et al. 2003).

Estima-se que somente a indústria petroquímica nos Estados Unidos teve perdas anuais de 20 bilhões de dólares, e a indústria britânica, de até 27 bilhões de dólares, isto desconsiderando outras indústrias, como farmacêutica e química (Venkatasubramanian, et al. 2003).

5.3. Detecção e Diagnóstico de Falhas Em Máquina de Corrente Contínua

Um ambiente industrial necessita constantemente de potência nos seus processos fabris, seja no funcionamento de esteiras, pontes rolantes, máquinas de usinagem, bombas ou exaustores. Dentre as opções existentes para estes fins, os motores elétricos são muito utilizados, pois eles têm geralmente um baixo custo, facilidade de transporte, limpeza e comando, – construção

simples, melhores rendimentos e grande versatilidade de adaptação às cargas dos mais diversos tipos (de Almeida 2004). Os motores elétricos possuem vários componentes e servem aos mais variados propósitos, e como todo dispositivo, estão sujeitos a eventos anormais, inclusive falhas, sejam derivadas da sua concepção, implantação, operação ou manutenção.

Será utilizado neste trabalho o modelo de um sistema dinâmico de Corrente Contínua (CC) não linear de terceira ordem que considera os efeitos de falhas. Serão simulados o comportamento normal do motor, e as falhas, divididas em três grandes grupos: de atuadores (conversores da armadura e do campo); de planta ou processo (máquina e carga mecânica) e de sensores (medidores de corrente e velocidade).

O modelo será utilizado nos experimentos de falhas fornecendo os dados para a análise conjunta do RBC e SIA. Este modelo foi originalmente proposto com o objetivo de criar uma ferramenta de projeto e teste de sistemas de detecção e diagnóstico de falhas (Caminhas 1997). Para maiores detalhes, consultar o Apêndice A deste trabalho.

5.4. Geração dos Dados do Motor Elétrico de Corrente Contínua

A geração dos dados para uso do sistema de RBC será realizada através de um simulador que obedece aos princípios apresentados no modelo do sistema de acionamento do motor de corrente contínua.

As falhas geradas no simulador são divididas em falhas de atuadores, falhas em componentes de planta e falhas de instrumentação. As falhas geradas no simulador incluirão somente as de instrumentação, ou seja, no sensor de corrente da armadura (i_a), sensor de corrente da fonte (i_{fd}) e sensor de velocidade (ω_r), que são representadas pelas variáveis de saída do motor de corrente contínua. Os motivos da escolha pelas falhas de instrumentação são dois, primeiro pela facilidade de visualização gráfica e segundo pela simplificação da avaliação, explicitando, inicialmente, fortalezas e fraquezas do modelo de RBC e SIA. À medida que problemas e limitações forem encontrados, e eventualmente corrigidos, diferentes falhas podem ser considerados para avaliações futuras.

A geração de dados se deu com as seguintes configurações:

- Foram gerados os eventos de comportamento normal (sem falhas) e os três tipos de falhas de instrumentação;
- A cada simulação um tipo de evento era aleatoriamente escolhido;
- O simulador realizou 50 simulações com amostragens de 20 ms cada, coletando em cada simulação 200 pontos no tempo total de 4 s. As Figuras 5.1, 5.2 e 5.3 ilustram 4 simulações independentes, uma para cada evento. Cada figura representa as variáveis medidas por

simulação e como estas se alteram com as falhas geradas. A distribuição espacial de todas as 50 simulações está apresentada na Figura 5.4;

- Cada simulação possui um momento transitório inicial, que começa sempre com um comportamento normal, para em seguida sujeitar-se aos eventos selecionados aleatoriamente, que podem, inclusive, ser de comportamento normal;
- A máquina parte a vazio e caso um evento de falha aconteça, ele se dará em 50% do período da simulação, ou seja, 2 s após o início da simulação. Todas as simulações operam no comportamento normal (regime normal) até 50% do período, momento no qual uma falha pode ocorrer;
- Como toda simulação começa com um comportamento normal, ou seja, sem falha, este se torna muito repetitivo e pouco representativo ao longo das simulações, o que motivou a eliminação de 25% dos primeiros pontos gerados em cada simulação, reduzindo o conjunto de dados obtidos inicialmente de 10.000 para 7.500 pontos. A Tabela 5.1 mostra a quantidade de pontos por tipo de evento ao final de todas as simulações;
- Para cada simulação foi gerada variação aleatória na tensão de entrada de campo e armadura;
- Todas as variáveis de saída do motor são normalizadas entre [0,1] para uso pelo protótipo;
- Os parâmetros de funcionamento do motor estão na Tabela 5.2;
- Foi gerado ruído máximo de 0,5% na saída do modelo.

Evento	Número de pontos
Comportamento normal	4100
Falha no sensor de corrente da armadura	900
Falha no sensor de corrente da fonte	1300
Falha no sensor de velocidade	1200
TOTAL	7500

Tabela 5.1 – Distribuição dos pontos por evento.

Parâmetros	Valor
Ra (Ohm)	0,0046
Rfd (Ohm)	12,1875
La (H)	0,000238
Lfd (H)	8,750
Lafd (H)	0,23
Jm (Kg.m ²)	2580
Va (V)	$375 \leq V_a \leq 750$

Parâmetros	Valor
V _{fd} (V)	$375 \leq V_{fd} \leq 750$
I _a (A)	17098
I _{fd} (A)	61,5
B _m (Kg m ² /s)	127
W _r (rad/s)	48

Tabela 5.2 – Parâmetros do motor.

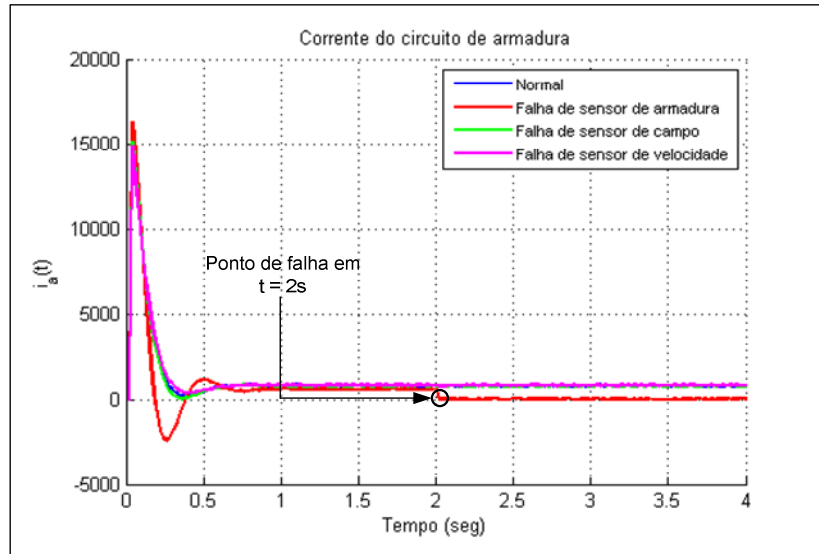


Figura 5.1 – Falha do sensor da corrente de armadura.

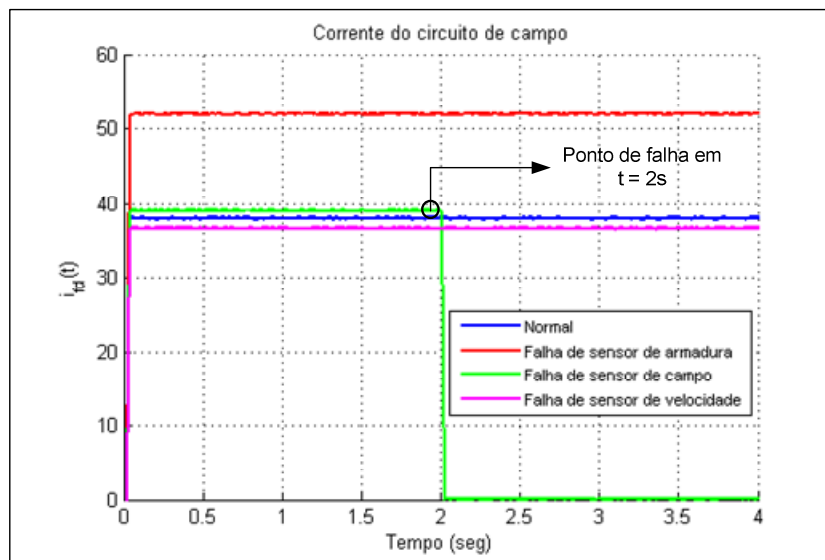


Figura 5.2 – Falha do sensor da corrente de campo.

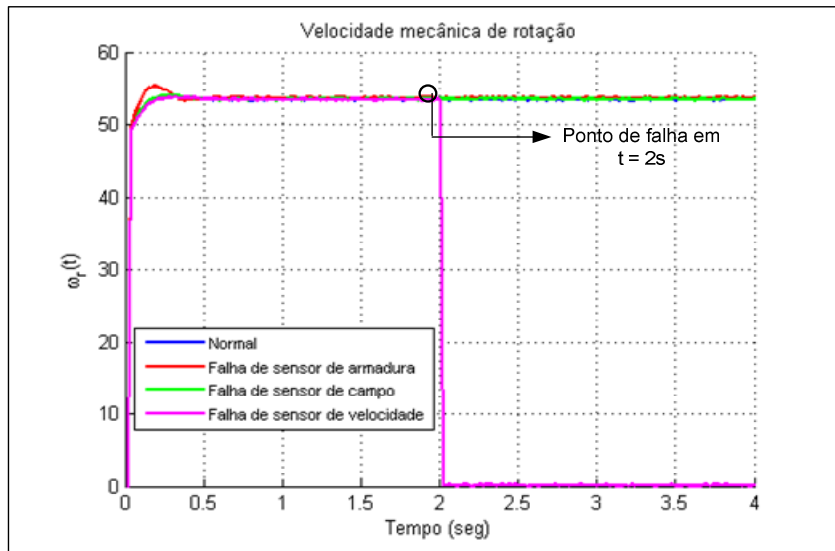


Figura 5.3 – Falha do sensor de velocidade.

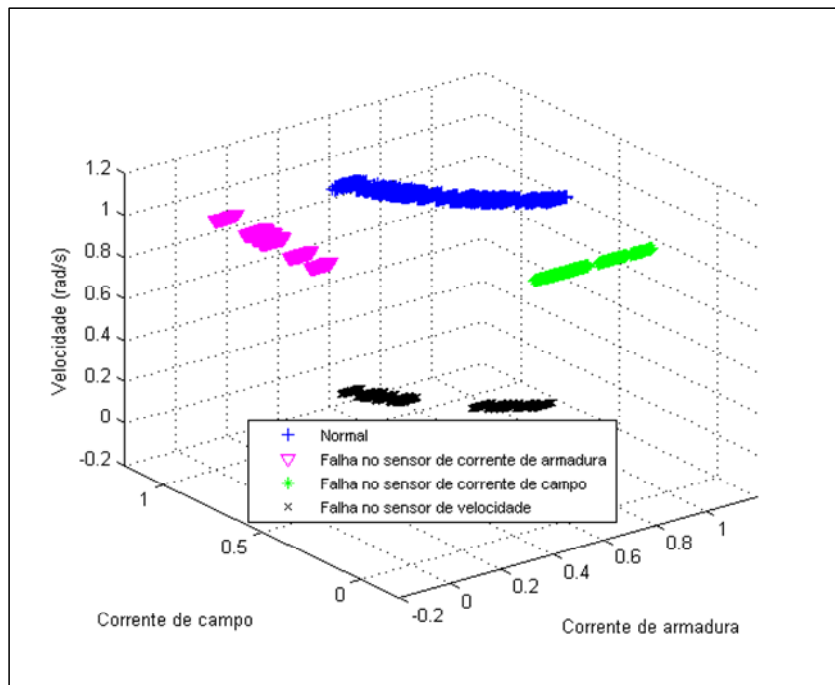


Figura 5.4 - Representação espacial dos casos de falhas da tabela 5.1.

5.5. Modelagem do Experimento

O experimento se dará através de uma abordagem conjunta do Raciocínio Baseado em Casos e do Sistema Imune Artificial, ambos já previamente apresentados. Este experimento será realizado através de protótipo que considera os quatro processos principais de um sistema RBC, a recuperação de casos, a reutilização e revisão (estes dois juntos sob o nome de adaptação) e por último a retenção (que inclui o armazenamento e a manutenção).

Toda base de casos começa com um número inicial de casos que tenham, de preferência, uma grande abrangência, facilitando a resolução dos futuros casos apresentados.

A nova proposta conjunta do RBC e SIA se baseia fortemente em dois pontos: no agrupamento da base de casos e no mapeamento da base de casos com os anticorpos do SIA, que funcionam como índices na execução de diversas atividades ligadas aos casos.

No domínio do experimento de falhas, é necessário que os eventos sejam detectados e diagnosticados com base em conhecimento prévio explícito ou, até mesmo, na ausência deste conhecimento, já que a abrangência dos eventos nem sempre é completamente conhecida antes do início do funcionamento real de um processo, obrigando o sistema a preparar-se para detectar novos eventos. A maneira como o RBC e SIA trabalham juntos será abordada aqui, utilizando um algoritmo específico de agrupamento.

No primeiro método de experimento, ou modalidade, será utilizado um conhecimento prévio dos possíveis eventos, sob a forma de treinamento do um sistema, o que facilita a resolução de problemas futuros semelhantes. A segunda modalidade, apesar de possuir um baixo conhecimento prévio, possibilita lidar com problemas ou situações, em geral desconhecidos ou não registrados. O direcionamento, na primeira modalidade, é feito, inicialmente, por uma base de casos com muitos casos cuja solução seja conhecida, viabilizando assim o treinamento de um algoritmo que agrupe e resolva novos casos, onde a resolução para o caso é dada pela detecção e diagnóstico do evento ou, em outras palavras, pelo tipo de operação que ele representa. Na segunda modalidade, como existe pouco conhecimento prévio dos casos do domínio, a proposta é a detecção da novidade em que, depois de detectada, pode ter a sua solução fornecida por um especialista ou por um modelo simplificado do domínio. Nas duas modalidades a solução é dada pela detecção e diagnóstico dos eventos, que são os casos já previamente apresentados ao protótipo ou inicialmente inseridos na base de casos.

5.5.1. Representação do Caso

No domínio de falhas de motor elétrico de corrente contínua, o caso será representado por duas partes, a parte problema e a parte solução. O problema possui três dimensões, sendo elas a corrente de armadura (i_a), a corrente de campo (i_{fd}) e a velocidade da máquina (ω_r), todas elas obtidas como variáveis de saída do motor elétrico. A solução se dará por uma única dimensão, o grupo ou classe ao qual aquele caso pertence, ou seja, se é um comportamento normal, falha no sensor da corrente de armadura, falha na corrente de campo ou falha na velocidade da máquina.

5.5.2. Algoritmo de Agrupamento

Em um ambiente fabril as leituras do ambiente são, muitas vezes, obtidas por sensores que encaminham as informações a um sistema que avalia a situação e toma decisões, podendo ou não haver interferência humana. Estas leituras são realizadas a partir da coleta de amostras em um determinado intervalo de tempo. Um sistema que trate a avaliação e classificação de eventos desta maneira, ou seja, ponto a ponto, torna-se interessante, pois trabalha em sintonia com a coleta de dados do processo.

O protótipo híbrido de RBC e SIA aqui proposto faz uso de um algoritmo de agrupamento participativo em conjunto com a aiNet para o diagnóstico e detecção de falhas de instrumentação em um motor de corrente contínua. Este algoritmo tem o objetivo de criar os grupos de casos baseados na compatibilidade dos casos para que os demais processos de RBC utilizem-nos de forma compartimentada, conforme detalhado no Apêndice B.

5.5.3. Detecção e Diagnóstico de Falhas Através de um Protótipo de Sistema Híbrido de RBC e SIA

Os experimentos são divididos em duas modalidades, na primeira são abordadas situações onde o número de grupos é conhecido, mas somente a classificação de alguns casos é conhecida, realizando-se assim um período de treinamento, seguido por um de validação, quando casos completamente desconhecidos para o protótipo são apresentados. Ao longo deste processo, o ciclo de vida de RBC com SIA é tratado e medido para avaliação, passando pela recuperação, reutilização, revisão e retenção dos casos. Na segunda modalidade, somente alguns pontos iniciais de operação normal são conhecidos e, a partir daí, os casos são apresentados, agrupando-se com base no conhecimento prévio existente e na ausência de treinamento relativo a outros possíveis agrupamentos presentes nos dados. A segunda modalidade também passa pelo ciclo de vida de RBC com SIA, mas focando-se em como o algoritmo de agrupamento participativo, junto com a aiNet, é capaz de criar agrupamentos com conhecimento prévio mínimo sobre os casos existentes.

Os casos utilizados nas duas modalidades foram obtidos realizando-se uma filtragem na base original da Tabela 5.1. Na primeira modalidade, Tabelas 5.3, 5.4 e 5.5, os dados de cada evento da Tabela 5.1 foram selecionados seguindo a mesma distribuição da Figura 5.4, objetivando facilitar o processo de análise de resultados. Nesta modalidade foram realizadas três execuções, uma com 50% (Tabela 5.3 e Figura 5.5), 25% (Tabela 5.4 e Figura 5.6) e 12,5% (Tabela 5.5 e Figura 5.7) dos dados da Tabela 5.1. Cada execução origina-se com a inicialização de uma quantidade fixa de casos de comportamento normal, passando, em seguida, por um treinamento que envolve casos de todos os tipos existentes nos dados (comportamento normal e falhas) para ajustes dos parâmetros do algoritmo. Na seqüência, acontece a validação, que é a apresentação de casos desconhecidos ao protótipo para

serem resolvidos (detecção e diagnóstico) sem supervisão. Após a validação, são realizadas operações de busca de casos, cuja função é medir a eficiência do processo de busca em uma base de casos segmentada em grupos e permeada pelos índices, ou anticorpos, gerados pela aiNet. Ao final das três execuções da primeira modalidade, será realizada manutenção da base de casos para eliminação de casos desnecessários. As distribuições dos dados das três execuções foram criadas a partir da distribuição da Tabela 5.1 e são muito semelhantes a esta, mas todos os casos das etapas de inicialização, treinamento, validação e busca de cada execução são apresentados de forma aleatória.

Na segunda modalidade será realizada uma única execução com 12,5% dos dados da Tabela 5.1. Esta execução é iniciada com uma pequena quantidade de casos de comportamento normal, quando são apresentados em seguida e de forma aleatória o restante dos casos, até então desconhecidos do protótipo. O objetivo é medir o quão hábil foi o algoritmo de agrupamento participativo junto com os anticorpos da aiNet em agrupar os casos, de forma a detectar e diagnosticar os eventos, partindo de uma base de casos com baixa diversidade.

A execução da segunda modalidade obedece à mesma distribuição da terceira execução da primeira modalidade, diferenciando-se desta nas etapas posteriores à inicialização, quando todos os casos apresentados são desconhecidos e o processo irá operar de maneira não supervisionada, sem o conhecimento prévio de todos os possíveis agrupamentos existentes nos casos. Apesar de algumas quantidades de casos nas tabelas 5.3, 5.4 e 5.5 serem fracionárias para atender à porcentagem, o experimento arredondou os valores, preservando os as quantidades presentes nas classes normal, falha 1, falha 2 e falha 3.

1ª. Modalidade – Execução1 - base de casos com 50% dos casos gerados					
Tipo	# Casos	Início – 10%	Trein.– 60%	Val. – 20%	Busca – 10%
Normal	2050	205	1230	410	205
Falha1	450	0	292,5	112,5	45
Falha2	650	0	422,5	162,5	65
Falha3	600	0	390	150	60
TOTAL	3750	205	2335	835	375

Tabela 5.3 – Distribuição dos casos na 1ª. execução da 1ª. modalidade.

1ª. Modalidade – Execução2 - base de casos com 25% dos casos gerados					
Tipo	# Casos	Início – 10%	Trein.– 60%	Val. – 20%	Busca – 10%
Normal	1025	102,5	615	205	102,5
Falha1	225	0	146,25	56,25	22,5
Falha2	325	0	211,25	81,25	32,5
Falha3	300	0	195	75	30
TOTAL	1875	102,5	1167,5	417,5	187,5

Tabela 5.4 – distribuição dos casos na 2ª. execução da 1ª. modalidade.

1ª. Modalidade - Execução3 - base de casos com 12,5% dos casos gerados					
Tipo	# Casos	Início – 10%	Trein.– 60%	Val. – 20%	Busca – 10%
Normal	512,5	51,25	307,5	102,5	51,25
Falha1	112,5	0	73,125	28,125	11,25
Falha2	162,5	0	105,625	40,625	16,25
Falha3	150	0	97,5	37,5	15
TOTAL	937,5	51,25	583,75	208,75	93,75

Tabela 5.5 – distribuição dos casos na 3ª. execução da 1ª. modalidade.

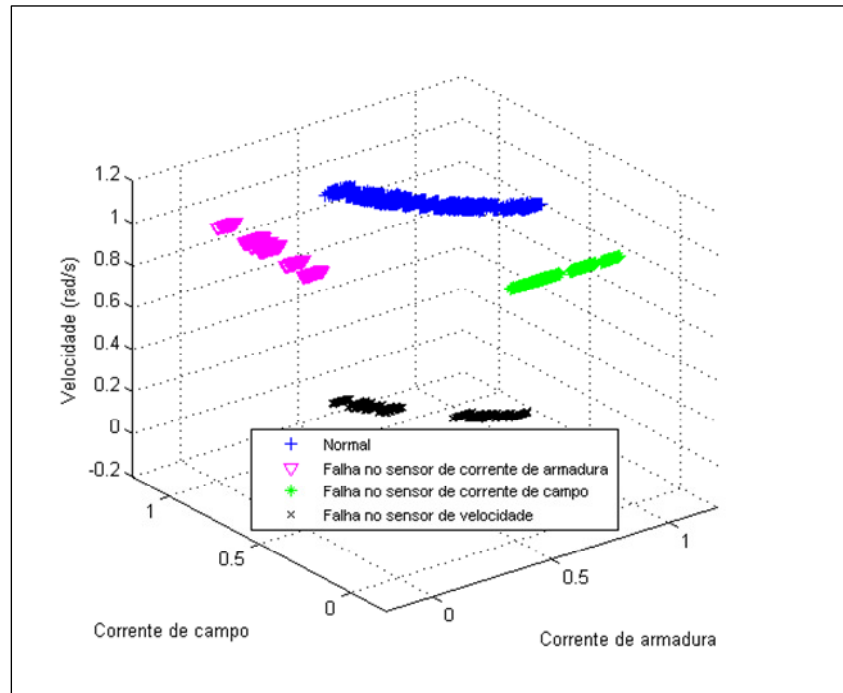


Figura 5.5 – Dados da execução 1 da 1ª. modalidade, com 3750 casos.

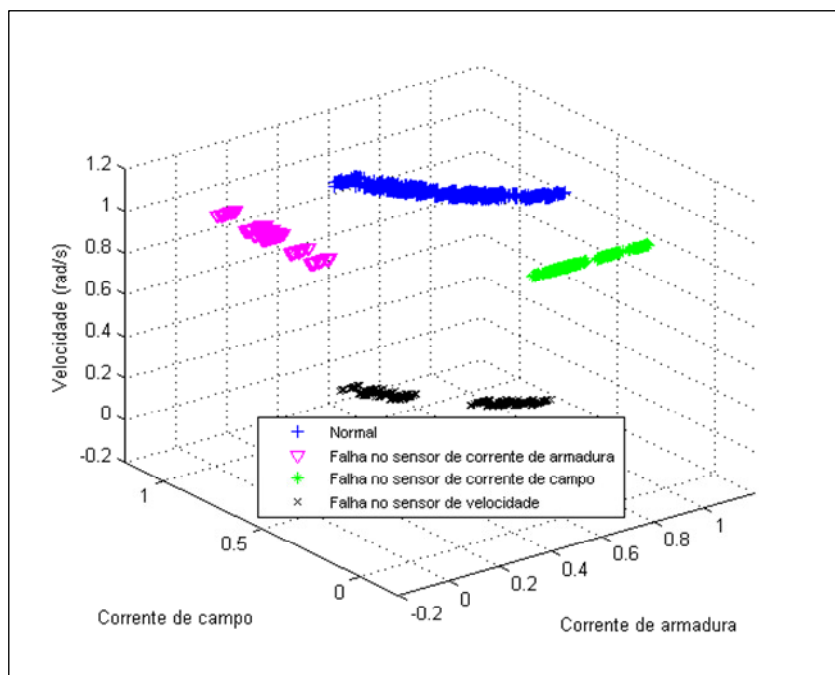


Figura 5.6 – Dados da execução 2 da 1ª. modalidade, com 1875 casos.

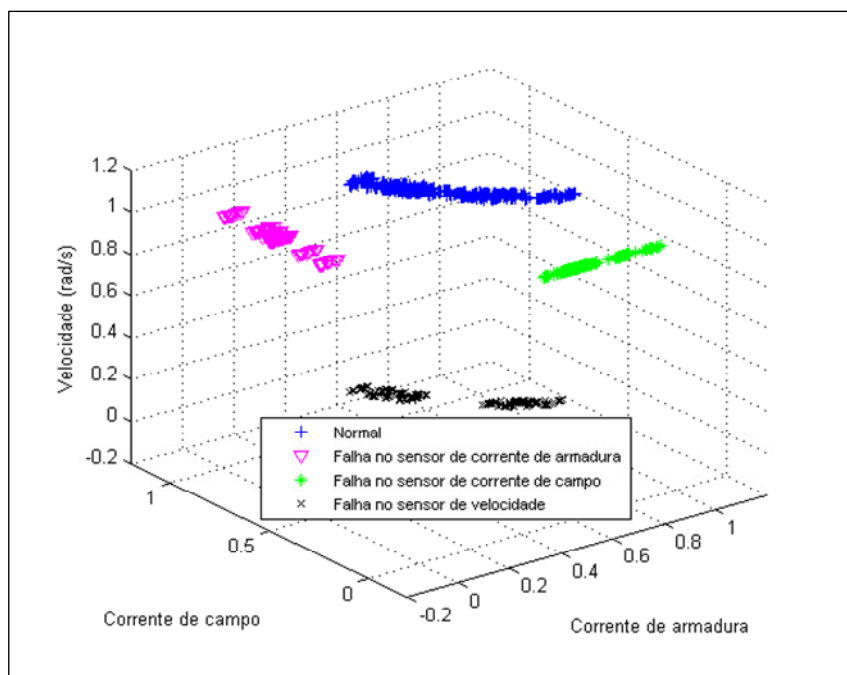


Figura 5.7 – Dados da execução 3 da 1ª. modalidade, com 932 casos.

Após a contextualização do domínio de motores elétricos de corrente contínua, representação dos casos e detalhamento do algoritmo utilizado, as execuções da primeira modalidade de testes operam como a seguir:

- I. Seleccionam-se aleatoriamente alguns casos de operação normal.

- II. Executa-se a aiNet nos casos do item I. Ao final, obtem-se a distribuição dos anticorpos relativa aos casos e o centro do grupo calculado baseando-se na distribuição dos anticorpos. Os anticorpos são chamados de índices.
- III. Cria-se a rede de casos entre os índices e os casos.
- IV. Inicia-se a apresentação, ponto a ponto, dos casos da etapa de treinamento, sendo estes agrupados, considerando-se os parâmetros do algoritmo de agrupamento participativo, adaptado a um aprendizado supervisionado.
- V. Como processo supervisionado, o algoritmo de agrupamento realiza as devidas correções nos agrupamentos (caso a detecção e diagnóstico estejam incorretos), ajustando o centro do grupo ao qual o caso pertence e executando a aiNet, neste grupo, para reposicionar os anticorpos. Ao final do treinamento, grupos de casos estão criados, todos com seus centros, índices de compatibilidade e alerta atualizados, incluindo os índices de casos, distribuídos de acordo a última atualização da aiNet em cada grupo.
- VI. Em seguida é realizada a etapa de validação, onde uma fração dos casos, dependendo de cada execução, é apresentada ao protótipo, mas com a diferença que a resolução de problemas é realizada contando com o conhecimento adquirido na etapa de treinamento. O ajuste de parâmetros e execução da aiNet é realizado da mesma maneira que no item V. Como não existem mais correções de casos erroneamente classificados, como no treinamento, o algoritmo de agrupamento participativo pode executar livremente todas as suas principais funções: inserir o caso em um grupo existente, criar um novo grupo ou unir grupos.

Ao final da etapa de validação, o processo de resolução de diagnóstico e detecção de falhas está concluído, mas ainda existem atividades importantes que são realizadas na base de casos para que o ciclo de vida de RBC com SIA esteja completo, são elas:

- VII. Buscar casos na base de casos. Além da detecção e diagnóstico necessários apresentados nos estágios anteriores, é perfeitamente possível que consultas sejam realizadas fora deste ciclo de resolução de problemas, seja para levantamento de informações ou manutenção da base de casos.
- VIII. Manter a base de casos. À medida que as bases de casos aumentam, aumenta a probabilidade de resolução direta de casos, mas, em contrapartida, cresce o tempo para encontrar um caso que possivelmente venha a resolver o caso apresentado. A manutenção endereça esta dicotomia, removendo casos desnecessários e melhorando a eficiência do sistema.

Se necessário, uma adaptação do tipo geral é realizada (busca por casos mais relevantes em grupos vizinhos ao grupo inicialmente selecionado no processo de recuperação) na primeira e segunda modalidade dos testes.

Já na segunda modalidade, a execução do algoritmo de agrupamento opera de forma semelhante à sua concepção, ou seja, em modo não supervisionado e diferenciando-se da primeira modalidade de execuções a partir do item IV. Apesar da utilização do termo “não supervisionado” na execução da segunda modalidade, o mais correto seria algo próximo ao não supervisionado, ou semi-supervisionado, pois se, em RBC, um caso não for solucionado de forma direta, é muito comum o uso do conhecimento de um especialista ou de um modelo genérico do domínio para que o caso seja resolvido, constituindo-se uma supervisão, mesmo que em menor escala. Desta maneira, desconsidera-se o processo como não supervisionado. Conseqüentemente, os itens IV, V e VI são alterados para que operem na segunda modalidade em modo semi-supervisionado e os itens VII e VIII permanecem inalterados.

5.6. Coleta e Análise de Resultados

Após a execução dos experimentos, na primeira e segunda modalidade de execuções, os resultados obtidos na detecção e diagnóstico de casos, na recuperação, adaptação (reutilização e revisão) e retenção (inserção e manutenção) de casos estão listados a seguir. Concomitantemente são realizadas análises dos resultados, baseando-se em cada processo abordado pelo modelo híbrido de RBC e SIA.

5.6.1. Primeira Modalidade

A primeira modalidade engloba três execuções, nas quais a detecção e diagnóstico de falhas são iniciados de modo supervisionado, a partir de um treinamento, quando, em seguida, as execuções seguem com o conhecimento adquirido no treinamento para realizar a detecção e diagnóstico de novos casos. Após o processo propriamente dito de detecção e diagnóstico de falhas, diversos métodos de buscas são comparados, baseando-se em medidas específicas de desempenho do RBC. Em seguida é realizada a avaliação dos processos de adaptação e retenção de casos.

5.6.1.1. Diagnóstico e detecção de falhas

Após a realização do treinamento nas execuções com 50%, 25% e 12,5% da base de casos originalmente gerada na Tabela 5.1, os dados de validação foram apresentados ao protótipo, que realizou a resolução destes casos com 100% de acerto, como mostrado na Figura 5.8.

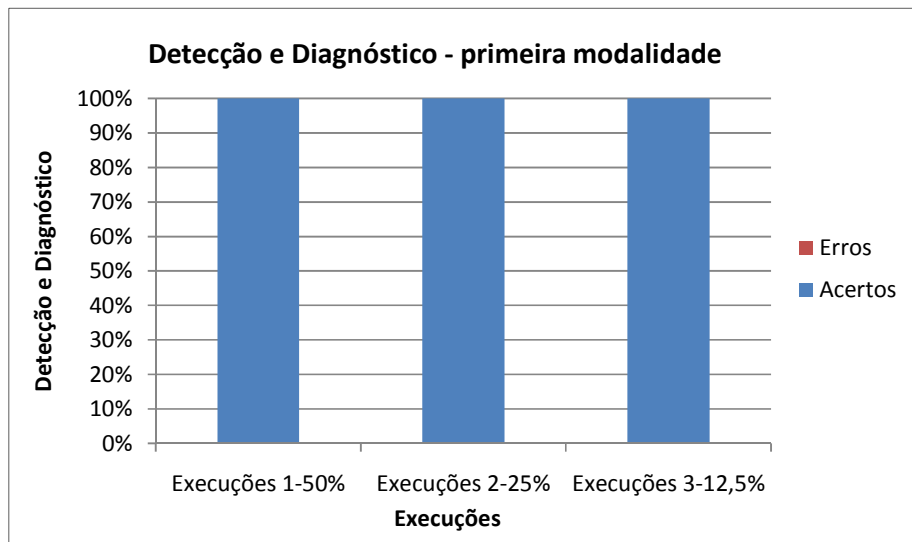


Figura 5.8 – Resultado das detecções e diagnósticos dos casos na primeira modalidade.

É interessante colocar que, apesar do resultado final ter sido completamente favorável, ao longo do processo de detecção e diagnóstico formaram-se grupos de casos que não representavam distribuições reais. À medida que a execução prosseguia, estes grupos foram unidos a outros grupos, pois o algoritmo de agrupamento entendeu que os casos eram semelhantes e deveriam fazer parte do mesmo grupo. O algoritmo de agrupamento utiliza o centro do grupo para dar início à resolução do caso apresentado, mas diferentemente de como foi concebido, o centro não é derivado da análise dos casos do grupo, mas, sim, da distribuição dos anticorpos do grupo, demonstrando que é possível realizar um trabalho de resolução de casos, especificamente de agrupamento, com uma representação compacta dos casos, que são em número bem menor aos casos.

Como a distribuição dos casos é muito semelhante nas três execuções, serão representados espacialmente alguns estágios da execução com menor número de casos, da base que representa 12,5% (Tabela 5.5 e Figura 5.7) dos casos gerados, pois essa quantidade de casos facilita a visualização dos anticorpos, antígenos e conexões da rede. O treinamento e validação desta execução estão presentes nas Figuras 5.9 a 5.13. As demais execuções, de 50% e 25%, tiveram o mesmo aproveitamento, mas com distribuições e número diferente de anticorpos, já que a apresentação dos casos é aleatória e o tamanho das bases é diferente.

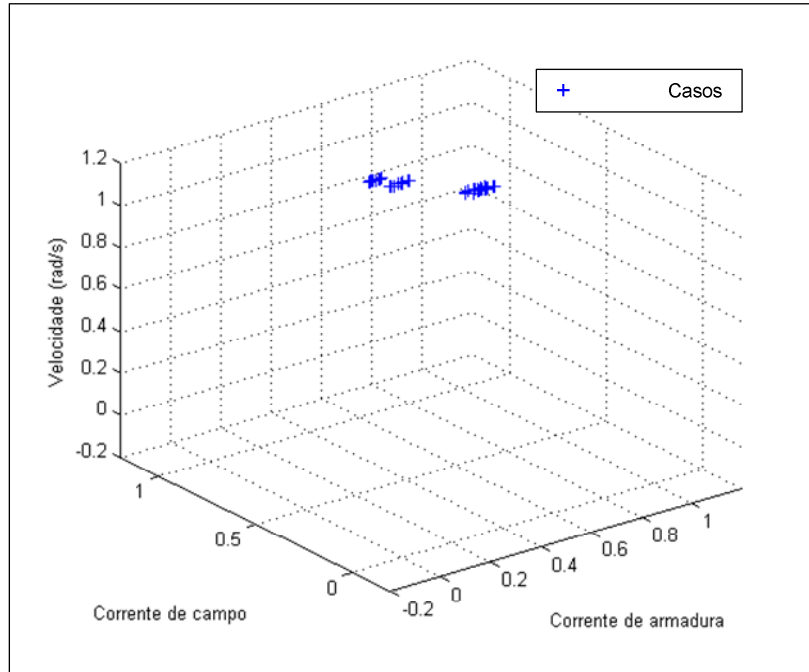


Figura 5.9 – Casos de uma operação normal inicialmente fornecidos ao protótipo.

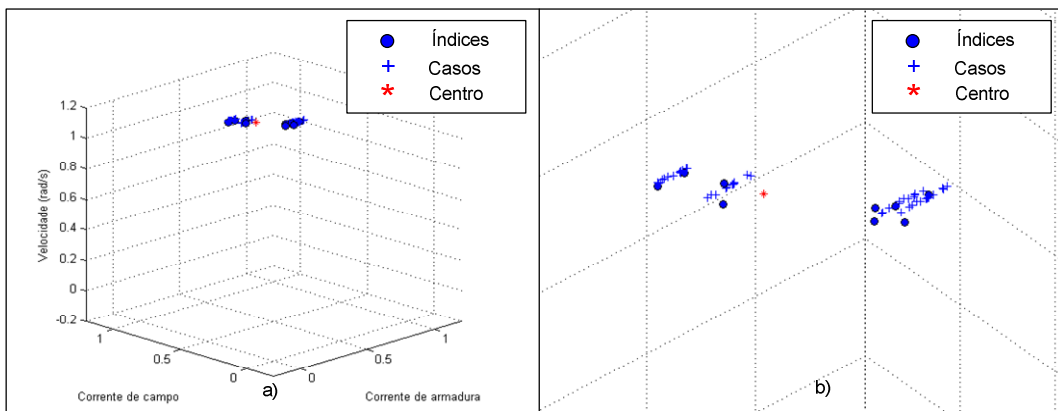


Figura 5.10 – a) Os índices distribuídos no espaço e seu centro após a execução da aiNet considerando os casos fornecidos na Figura 5.9. b) Ampliação da área povoada com casos e índices da Figura 5.10a.

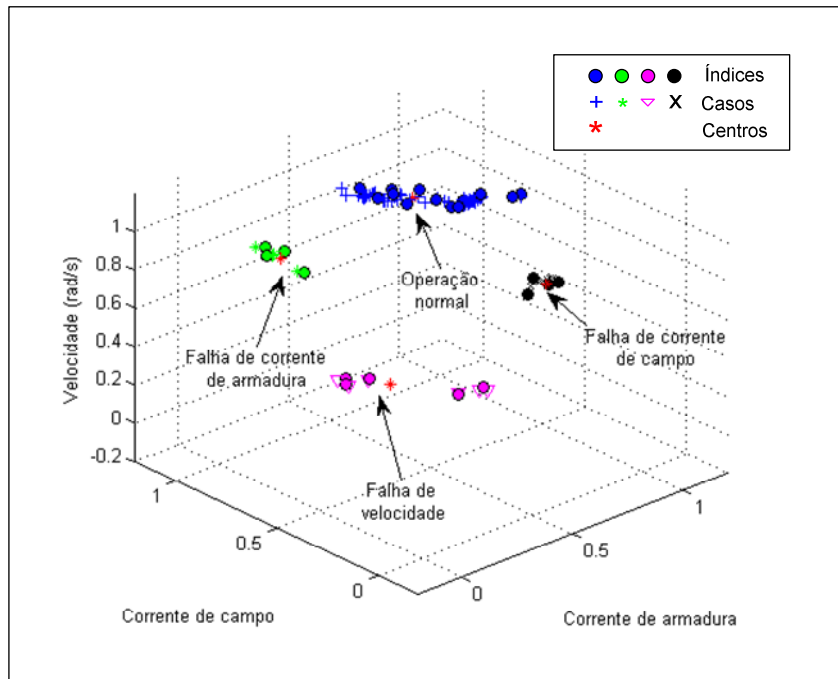


Figura 5.11 – Casos agrupados com seus índices e centros durante o treinamento.

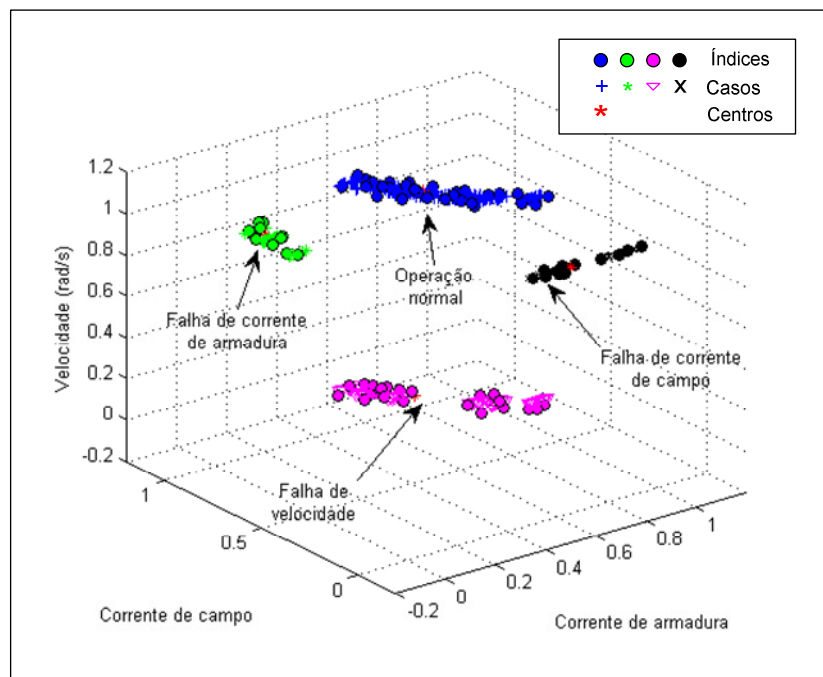


Figura 5.12 – Casos agrupados durante a validação.

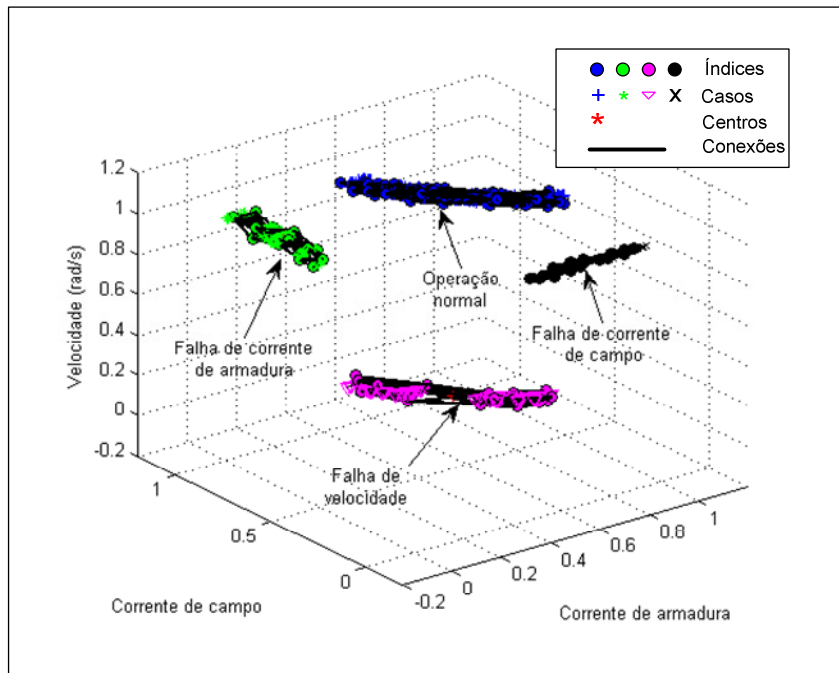


Figura 5.13 – Resultado final da detecção e diagnóstico de casos com formação da rede de casos.

A criação de grupos de casos melhora o processo de busca, pois evita uma ampla recuperação de casos e execução desnecessária da aiNet em toda a base de casos, o que seria computacionalmente custoso. Mesmo realizando-se execuções incrementais da aiNet através do reposicionamento dos anticorpos somente no grupo ao qual o caso apresentado pertence, atestou-se que, quanto maior o grupo, mais lenta era a execução da aiNet. Este comportamento se deve porque todos os anticorpos inicialmente criados são apresentados a cada antígeno do grupo pela aiNet, quando a afinidade de cada anticorpo é calculada. Ao final do processo de remoção dos anticorpos de menor afinidade em relação aos antígenos, é realizada uma avaliação geral, analisando também o quão próximo eles estão entre si. Quanto maior a proximidade, maior a chance de um deles ser eliminado, pois considera-se que são muito semelhantes, o que não contribui para a diversidade. Essa etapa de avaliação entre os anticorpos é de complexidade quadrática para o número de anticorpos e é consequência direta do número de antígenos do grupo e de parâmetros configuráveis da aiNet, como a condição de parada (número de épocas, por exemplo), número de anticorpos gerados pelo processo de clonagem, tolerância entre os anticorpos, etc. Encontrar um ponto ótimo entre estes fatores é extremamente importante para o desempenho da detecção e diagnóstico de falhas do protótipo.

Neste trabalho, optou-se por abordar um problema que demandasse uma solução baseada exclusivamente no agrupamento de casos, que se deu pela detecção e diagnóstico de falhas. O par problema/solução, aqui analisado é tratado por um algoritmo de agrupamento que funciona em conjunto com a aiNet ao longo dos processos do RBC. A recuperação de casos é o primeiro dos processos executados e objetiva conseguir identificar a qual grupo o caso pertence, resolvendo assim o problema de detecção e diagnóstico do caso

apresentado, pois a busca, aqui, se restringe a encontrar o grupo que apresenta a maior compatibilidade com o novo caso. Em muitos sistemas de RBC, a busca de casos se dá de maneira diferente, através do processo de descoberta do possível caso mais útil armazenado (e não exclusivamente pela identificação do grupo ao qual ele pertence), para que se extraia deste caso a informação necessária para a resolução do caso apresentado. A diferença básica entre os dois métodos de busca se dá pelo tipo de problema a ser resolvido, um de agrupamento/classificação e outro de extração de uma informação específica de um caso para ser utilizado na resolução do caso proposto. Estas diferentes abordagens geram diferentes tipos de busca.

A busca considerada na próxima seção representa a busca pelo caso mais útil, pois muitos domínios do RBC demandam soluções derivadas especificamente dos casos, o que gera a necessidade de avaliação de um método de recuperação de casos (busca e análise de compatibilidade) voltado para esta necessidade.

5.6.1.2. Recuperação de Casos

A recuperação de casos contempla a busca e a análise de casos na base de casos. O objetivo do experimento é encontrar o caso mais útil na base de casos, considerando um conjunto de casos a serem verificados, representados pelo campo de busca nas Tabelas 5.3 a 5.5. O caso encontrado neste experimento não apresentará uma solução explícita, já que esta foi dada pela detecção e diagnóstico das falhas através do algoritmo de agrupamento e aiNet na seção anterior, mas indicará o caso mais similar que poderá ser utilizado para comparação como, por exemplo, na realização de consultas a casos similares em um sistema híbrido de RBC e SIA. Enfatiza-se que outros trabalhos podem utilizar o método de busca aqui proposto para que este ajude a fornecer a melhor solução para um caso, para o qual se deseja obter, por exemplo, a solução de um problema de falha de instrumentação de velocidade e não somente a detecção e o diagnóstico da falha.

O método de recuperação de casos, baseado nos índices de casos e apresentado no Capítulo 4, será comparado com mais dois métodos: a busca completa e busca baseada no centro de um grupo.

A busca baseada em índices é realizada detectando-se primeiramente a qual agrupamento o caso candidato pertence e, em seguida, identificando qual índice, deste grupo, é o mais compatível com as suas características. A partir deste ponto, é realizada uma busca pelos casos mais próximos no entorno deste índice, recuperando-se os que estiverem abaixo de um determinado limiar (menor distância entre os casos), considerando uma base de casos normalizada. Como o processo de posicionamento dos índices é estocástico, é provável que a exploração dos casos no entorno de um único índice seja insuficiente para encontrar os casos mais relevantes. Torna-se necessário, então, que o experimento realize uma busca, ampliando sucessivamente o número de índices analisados, como na Figura 5.14, partindo-se sempre do mais próximo ao caso candidato até um número de índices que forneça um caso ou conjunto de casos semelhantes aos recuperados no método de busca completo.

Os outros dois métodos de busca são a busca completa, que analisa todos os casos da base de casos (desconsiderando agrupamentos) e a busca baseada no centro dos grupos, que analisa primeiramente o centro de cada grupo para decidir onde será realizada uma busca mais detalhada (Patterson, et al. 2002). Estes dois métodos desconsideram a existência dos índices na base de casos.

As medidas utilizadas para avaliar o desempenho dos três métodos são derivadas do trabalho de RBC apresentado por Smyth & McKenna (1999) e são:

- **Eficiência:** é o inverso do número de casos analisados durante o processo de recuperação de casos, mas para facilitar a apresentação, será representada pelo número direto de casos analisados e será calculada como uma média para cada execução.
- **Competência:** a percentagem de casos que podem ser resolvidos com sucesso, dado um algoritmo de recuperação de casos. Como neste trabalho a solução é a detecção e o diagnóstico de falhas, esta medida será alterada para indicar o quão próxima, em percentagem, está a recuperação dos casos úteis baseados nos métodos do índice e centro em relação ao método de buscas completo. A utilidade aqui se dará pela proximidade de casos que estejam abaixo de um determinado limiar. A competência será calculada como uma média da proximidade para cada execução.
- **Qualidade:** em linhas gerais, a qualidade da solução é uma função da proximidade entre o caso apresentado e o caso recuperado. À medida que a distância entre casos aumenta, um maior esforço de adaptação é necessário, tendendo a degradar a qualidade da solução. Apesar de ser uma abordagem simplificada, acredita-se que seja válida, pois fornece meios para conhecer as implicações do processo de recuperação na qualidade dos casos recuperados. A qualidade será calculada como a média das distâncias dos casos recuperados em cada um dos métodos de recuperação de casos.
- **Excelência:** a excelência representa o quão capaz o método analisado é de recuperar o caso mais útil para o caso apresentado. Se esse caso existir, o método de busca completa sempre conseguirá obtê-lo, pois ele procura em toda a base de casos, ao passo que os outros métodos realizam buscas direcionadas na base de casos e podem não conseguir recuperar este caso considerado ótimo. Esta medida será calculada como a média das vezes que se consegue recuperar o melhor (mais próximo), caso na base de casos em cada execução.

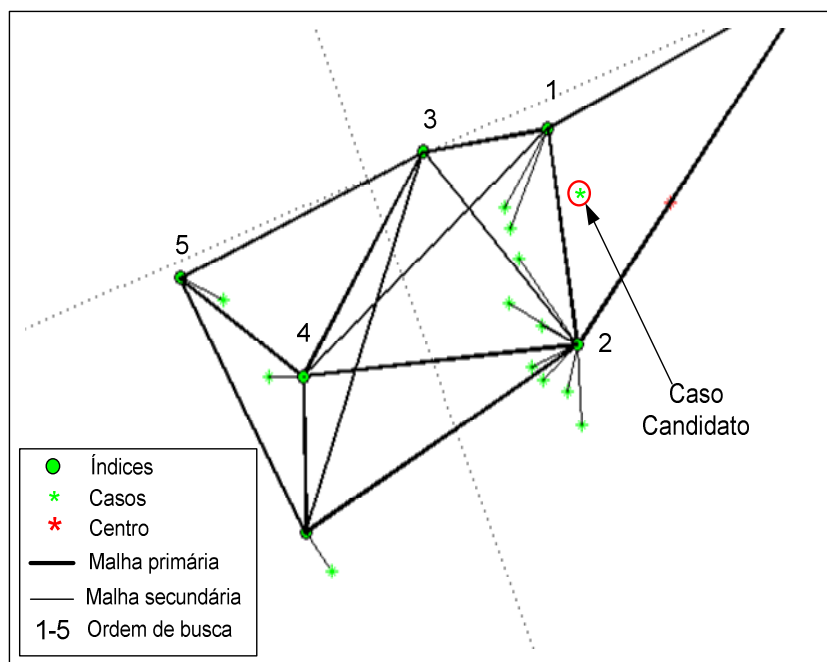


Figura 5.14 – Busca baseada em índices utilizando os cinco anticorpos mais próximos, ou vizinhos, do caso candidato.

As Figuras 5.15 a 5.34 representam os resultados dos três métodos de busca e análise de casos utilizando as medidas de avaliação já apresentadas.

A Figura 5.15 indica a eficiência da busca, utilizando-se a busca completa, o método baseado no centro e o baseado em um único índice de vizinhança. A Figura 5.16 mostra a competência para os três métodos de busca e uso de único vizinho para o método baseado nos índices. A Figura 5.17 coloca a qualidade para os três métodos de busca e uso de único vizinho para o método baseado nos índices. A Figura 5.18 expõe a excelência para os três métodos de busca com o uso de um único índice vizinho para o método baseado nos índices. As demais Figuras, de 5.19 a 5.34 apresentam os resultados, utilizando as mesmas medidas, ampliando gradativamente o número de índices de busca do método baseado em índices, de um para cinco. O número cinco foi escolhido, porque se observou que, com esta quantidade de índices, foram obtidos bons resultados na busca por casos úteis.

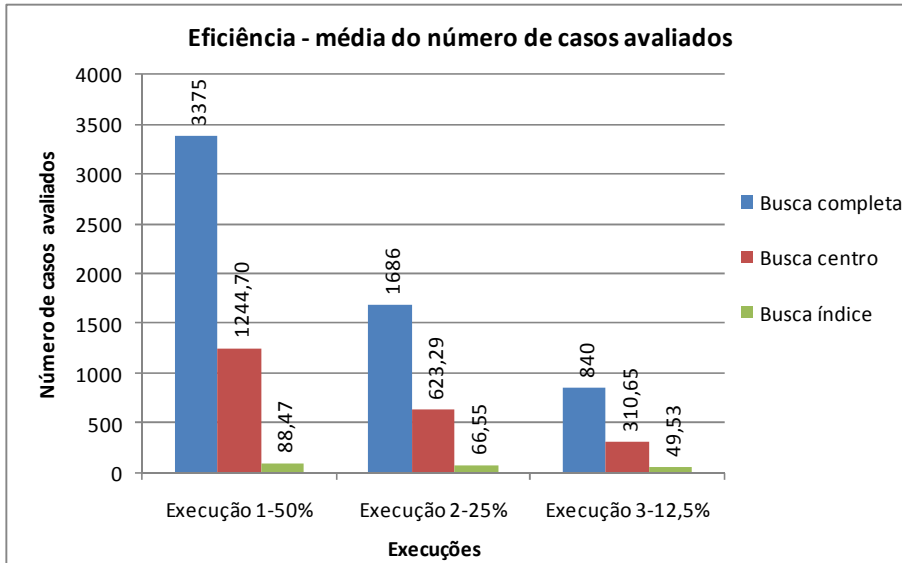


Figura 5.15 - Eficiência com a busca por índice com um vizinho.

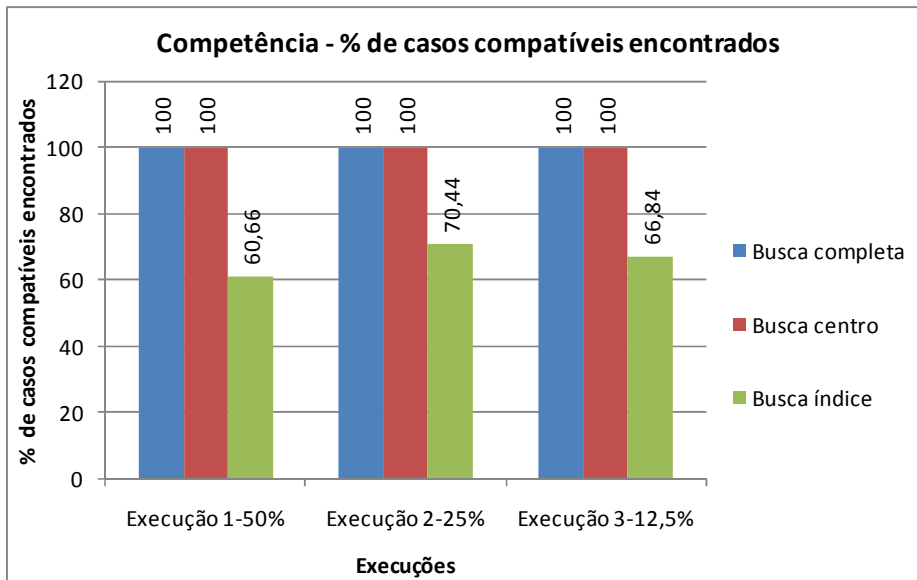


Figura 5.16 - Competência com a busca por índice com um vizinho.

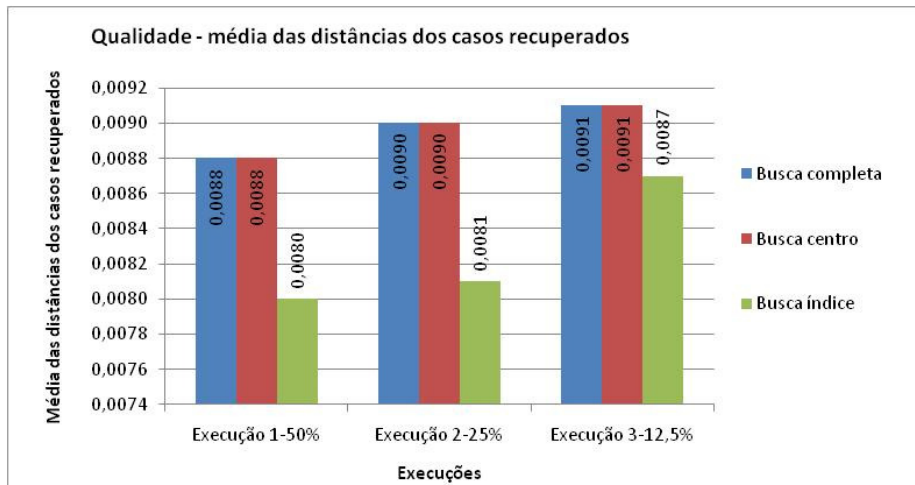


Figura 5.17 - Qualidade com a busca por índice com um vizinho.

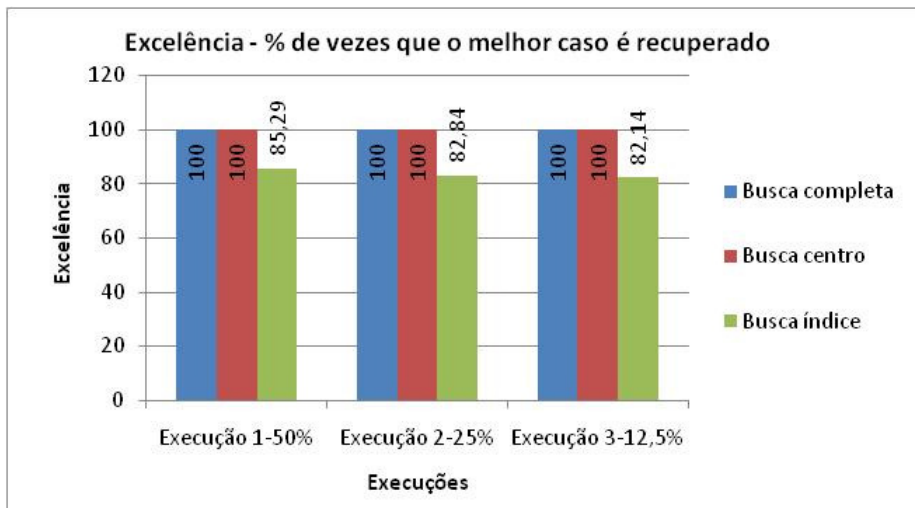


Figura 5.18 - Excelência com a busca por índice com um vizinho.

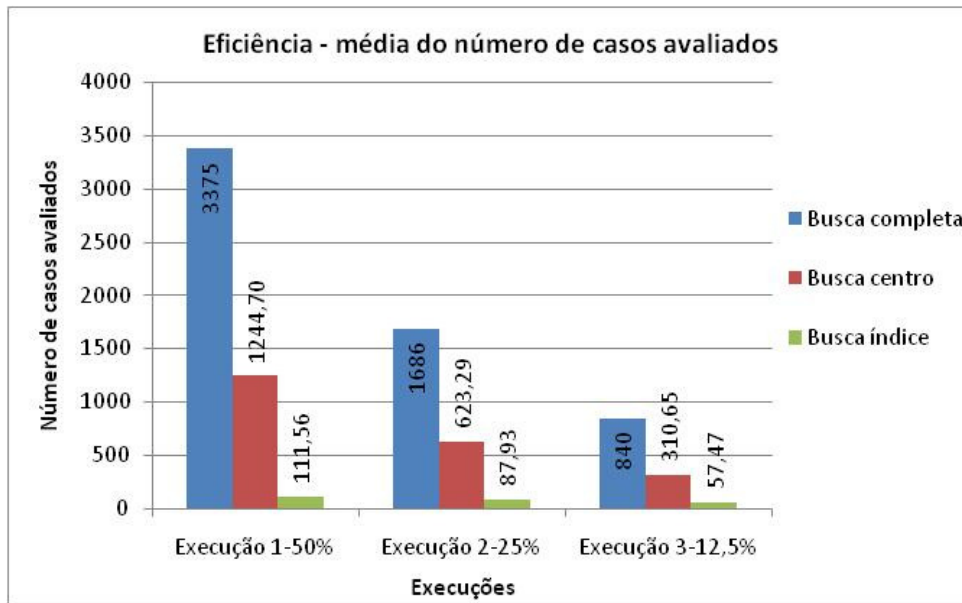


Figura 5.19 – Eficiência com a busca por índice com dois vizinhos.

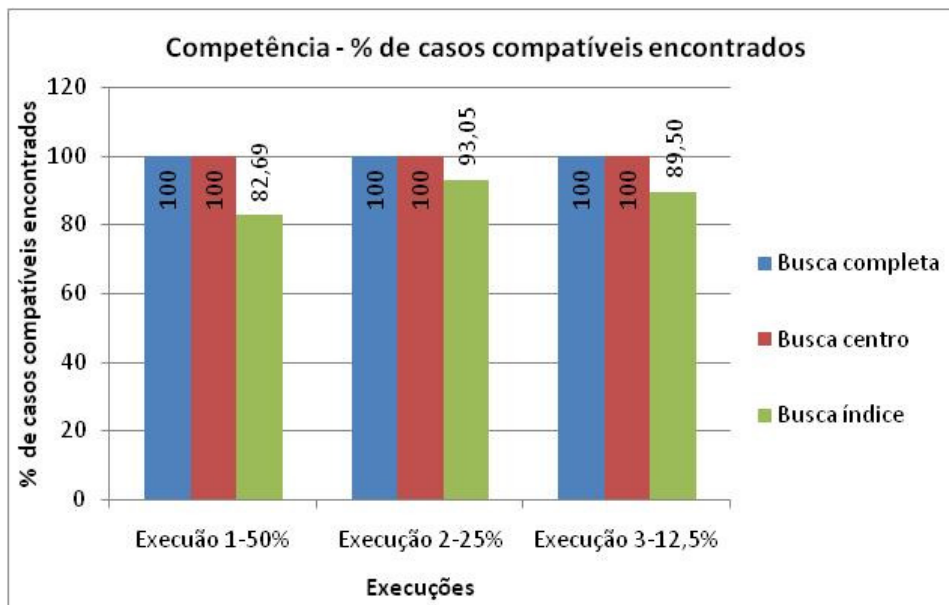


Figura 5.20 - Competência com a busca por índice com dois vizinhos.

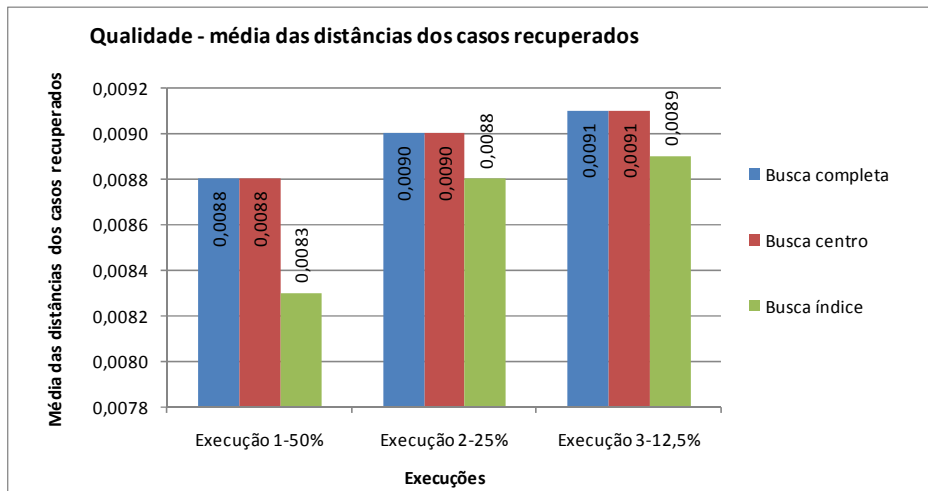


Figura 5.21 – Qualidade com a busca por índice com dois vizinhos.

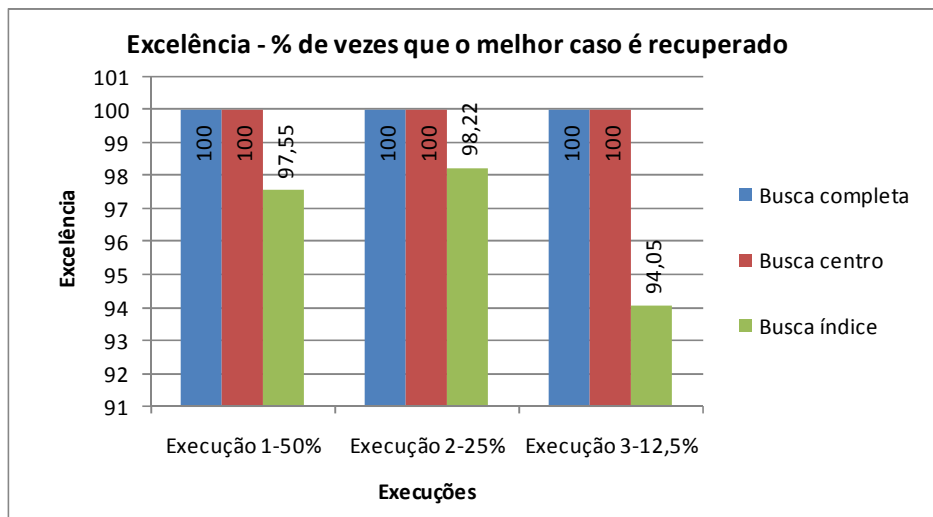


Figura 5.22 – Excelência com a busca por índice com dois vizinhos.

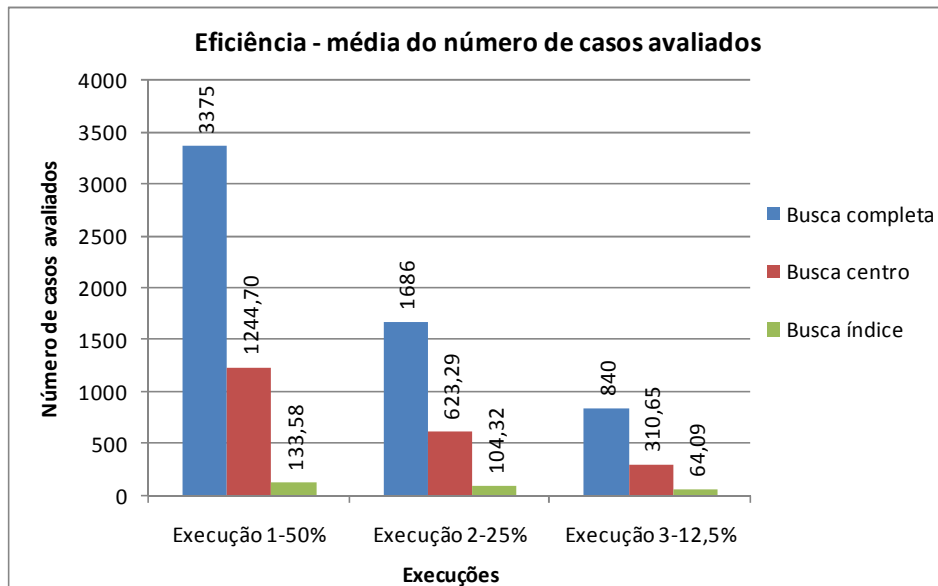


Figura 5.23 - Eficiência com a busca por índice com três vizinhos.

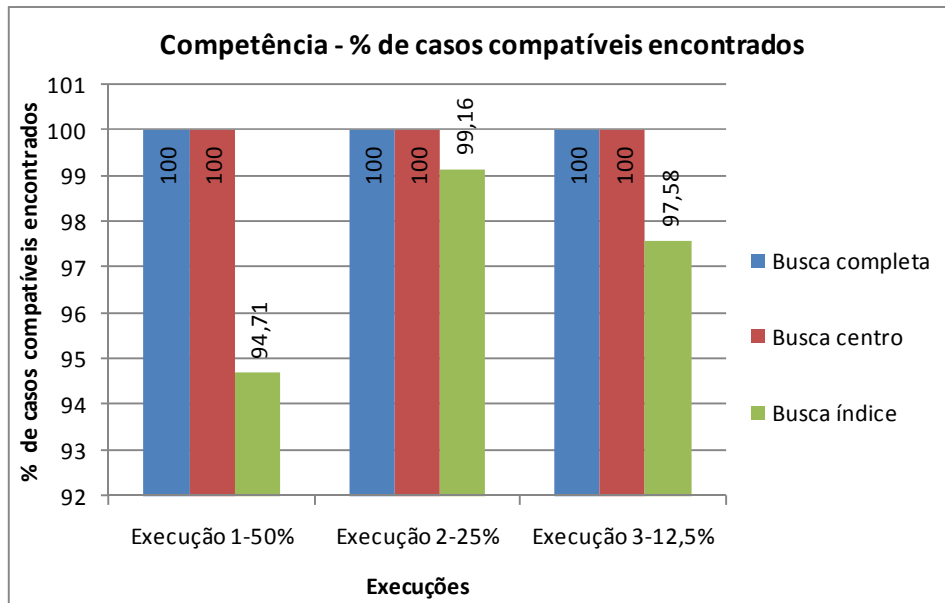


Figura 5.24 - Competência com a busca por índice com três vizinhos.

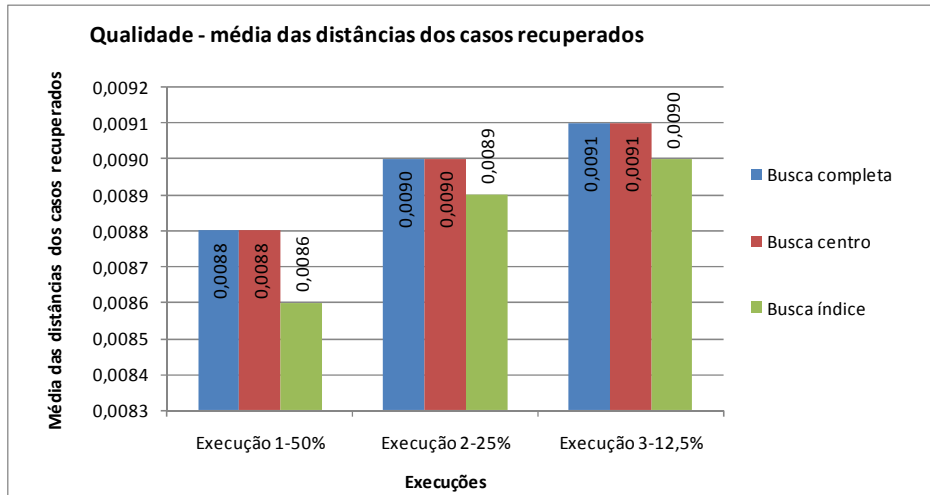


Figura 5.25 - Qualidade com a busca por índice com três vizinhos.

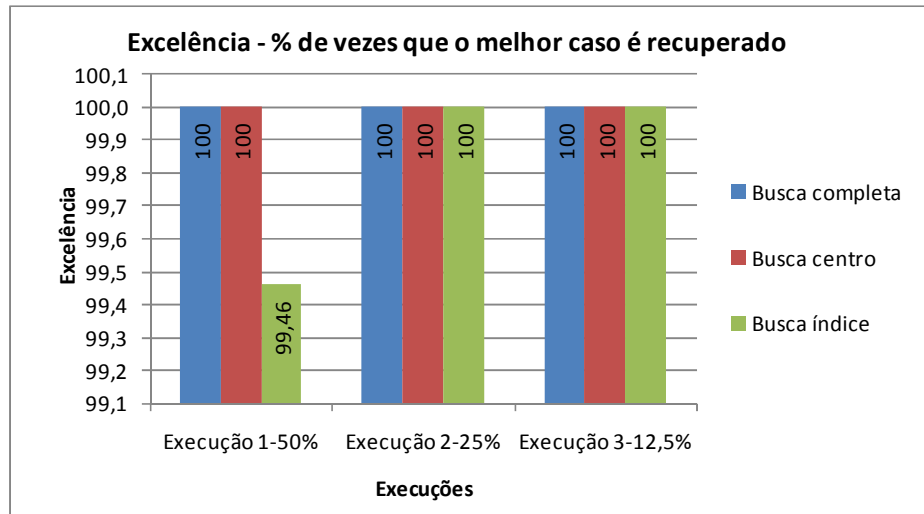


Figura 5.26 - Excelência com a busca por índice com três vizinhos.

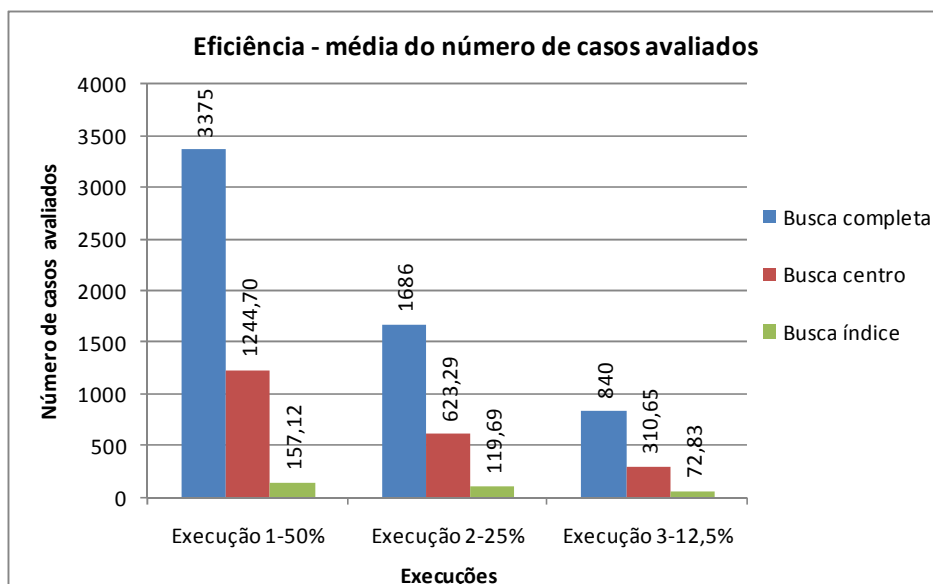


Figura 5.27 – Eficiência com a busca por índice com quatro vizinhos.

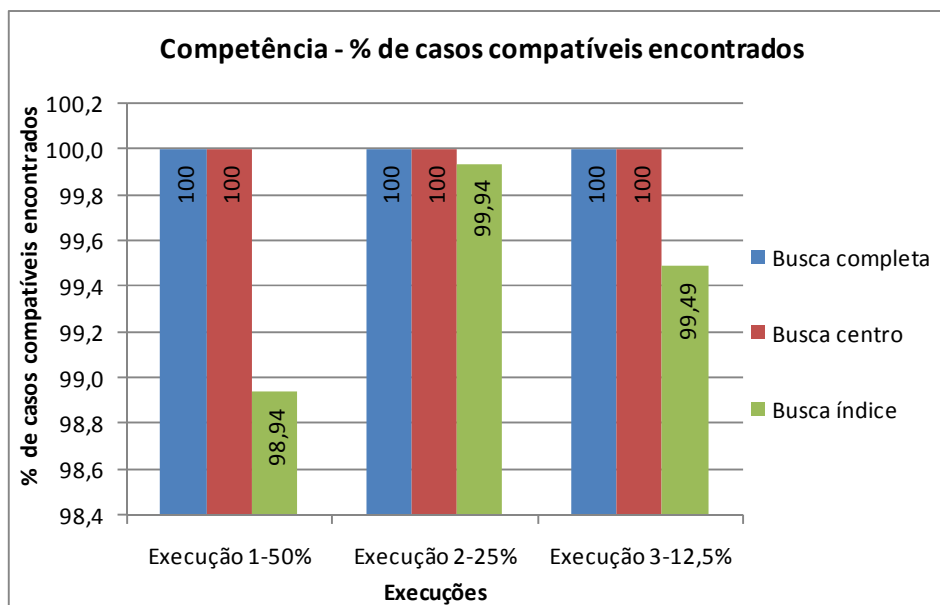


Figura 5.28 – Competência com a busca por índice com quatro vizinhos.

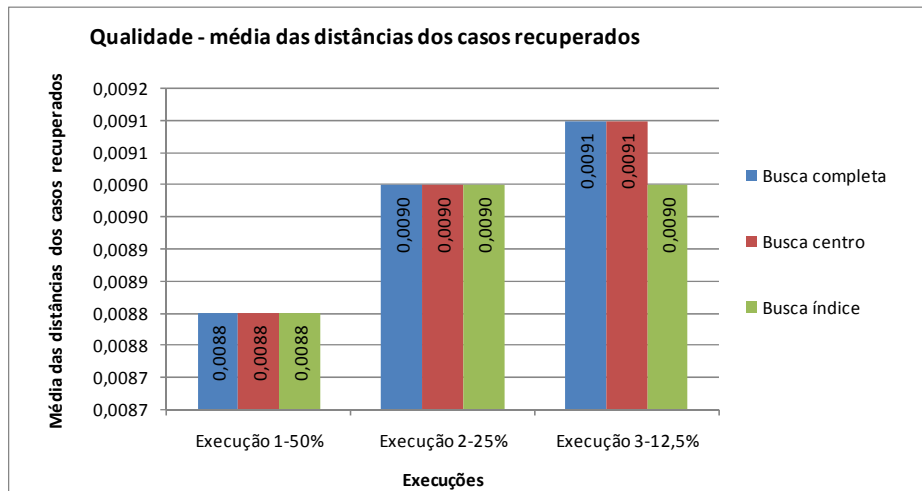


Figura 5.29 – Qualidade com a busca por índice com quatro vizinhos.

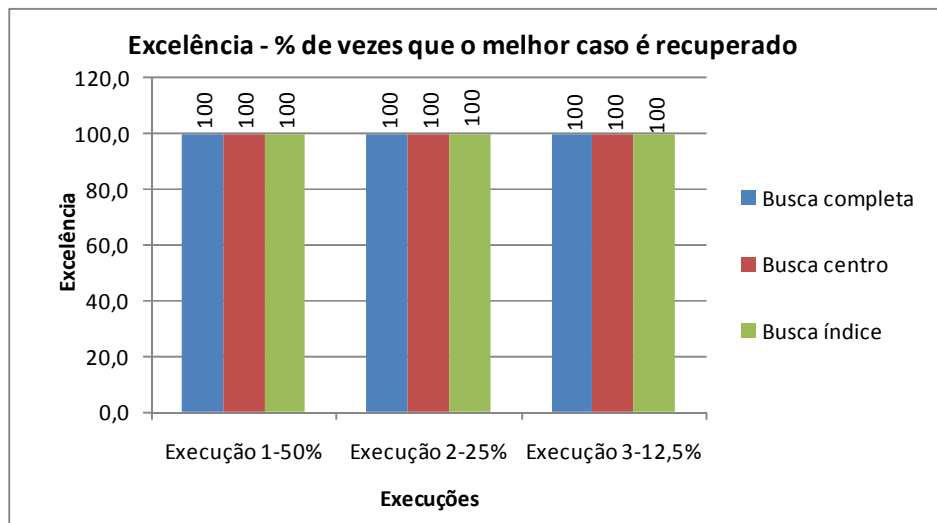


Figura 5.30 – Excelência com a busca por índice com quatro vizinhos.

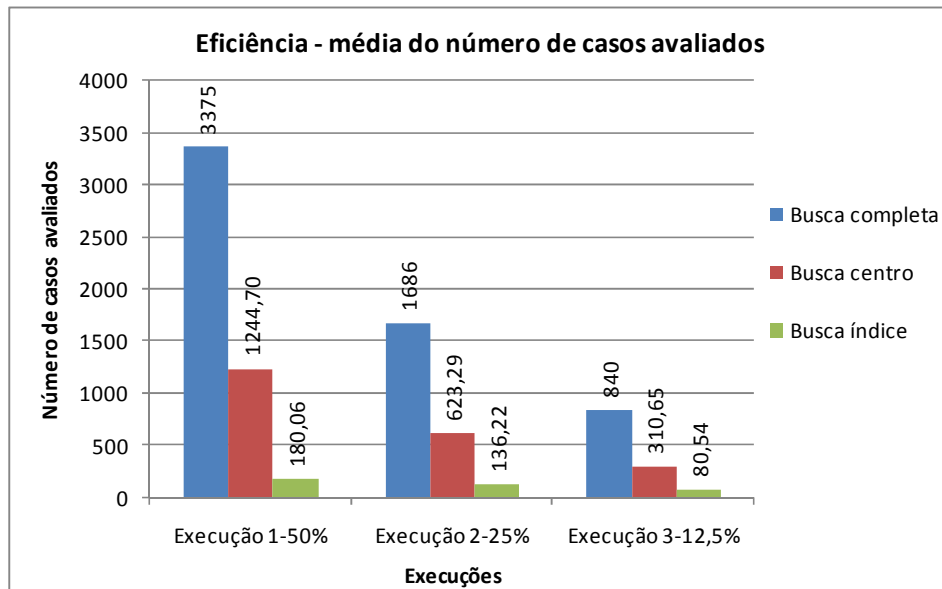


Figura 5.31 – Eficiência com a busca por índice com cinco vizinhos.

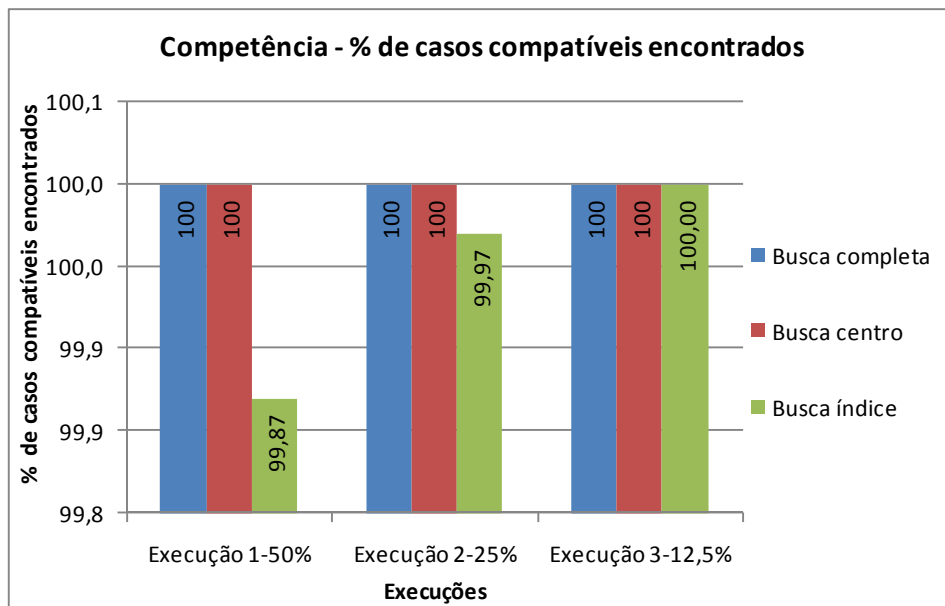


Figura 5.32 – Competência com a busca por índice com cinco vizinhos.

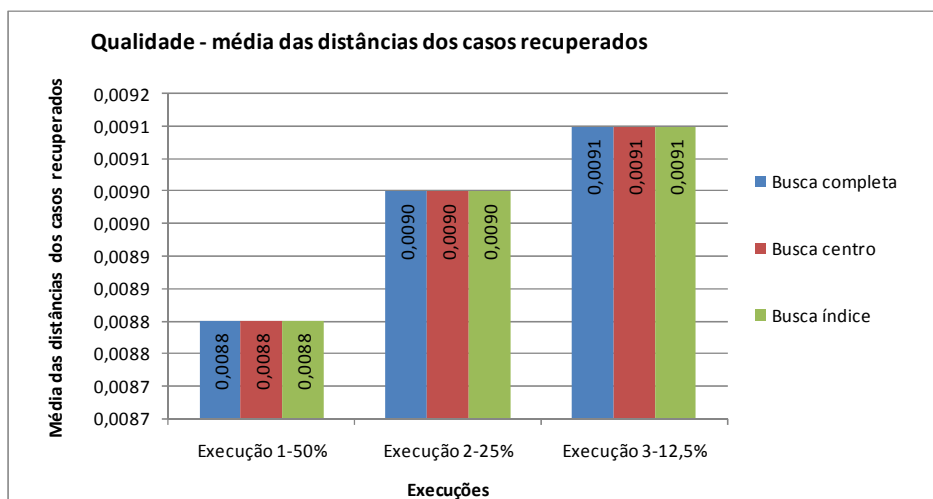


Figura 5.33 – Qualidade com a busca por índice com cinco vizinhos.

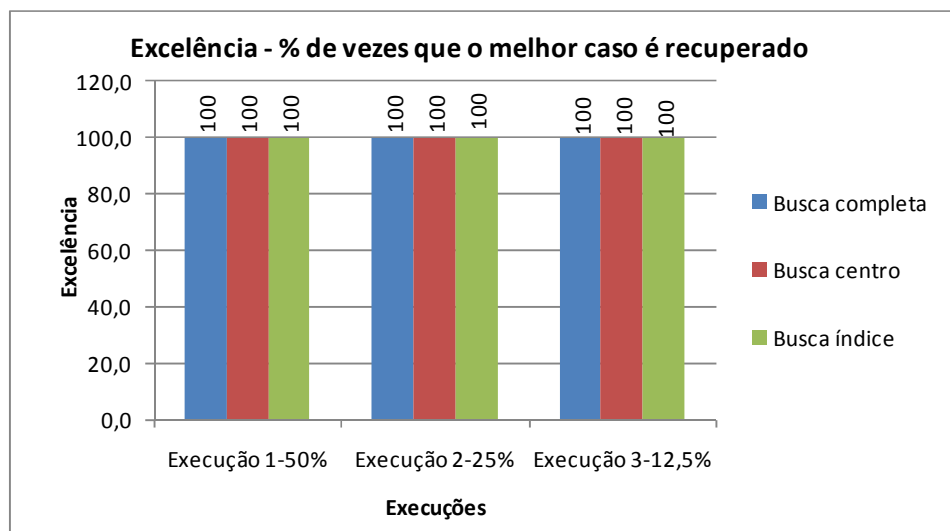


Figura 5.34 - Excelência com a busca por índice com cinco vizinhos.

É interessante notar que existe um número ótimo de vizinhos que fornece um equilíbrio entre a eficiência e a competência, já que o aumento gradativo e constante do número de vizinhos, no método baseado em índices, não gera melhoria nas medidas avaliadas, pelo contrário, diminui drasticamente a eficiência e competência, pois passa a explorar regiões do espaço onde não existe mais nenhum caso útil em relação ao caso apresentado. Os resultados demonstram que o uso dos índices melhora sensivelmente a eficiência com valores muito próximos nas medidas de competência, qualidade e excelência em relação aos métodos de busca completa e baseada no centro. Os resultados positivos foram obtidos após diversas execuções dos experimentos com diferentes quantidade de casos, índices, ordem de apresentação e distribuição, o que gerou o refinamento dos parâmetros do algoritmo de agrupamento participativo – taxa de aprendizagem, variação do alerta, limiares de criação e junção de grupos - e da aiNet – limiar de remoção de anticorpos, taxa de reprodução e critério de seleção de células de memória.

A Tabela 5.6 representa a taxa de compressão dos índices em relação aos casos, indicando uma alta compactação, que está também aliada a uma significativa representatividade dos casos da base de casos, já que a detecção e diagnóstico de falhas tiveram 100% de acerto, utilizando os índices como referência para cálculos dos parâmetros do algoritmo de agrupamento. Observando os dados da tabela, percebe-se que a compressão atinge seu máximo na maior base de casos. A explicação reside no fato de que dada uma distribuição semelhante de casos, os índices também se posicionarão de forma semelhante e irão abranger mais casos em uma base maior do que em uma menor base de casos.

Execução	Quantidade de casos	Quantidade de índices	Taxa de compressão
50%	3375	154	95,44%
25%	1686	119	92,94%
12,5%	840	110	86,94%

Tabela 5.6 – Quantidade de casos, índices e compressão dada a presença dos casos de início, treinamento e validação nas execuções das Tabelas 5.4 a 5.6.

5.6.1.3. Adaptação (Reutilização e Revisão)

Como o protótipo de detecção e diagnóstico de falhas trabalha com a identificação de um caso apresentado, ele não faz uso das informações no caso para extração de uma solução propriamente dita para a falha, mas utiliza as informações do adensamento de casos para fornecer a solução de detecção e diagnóstico de falhas de instrumentação. Nada impede que o protótipo ou sistema seja aperfeiçoado e também passe a fornecer uma solução mais ampla, como a correção da falha.

Tendo em vista que a atenção se dá na detecção e diagnóstico, a adaptação utilizada no protótipo é a de primeiro nível, ou geral, apresentada no Capítulo 4 e que consiste na identificação de casos similares no entorno de grupos diferentes do agrupamento inicialmente selecionado para o caso, diminuindo a possibilidade de uma adaptação custosa ou inserção incorreta nos agrupamentos da base de casos.

A percepção de que o caso pode fazer parte de outro(s) grupo(s) se dá, quando a diferença da compatibilidade entre o caso, o grupo mais compatível e outros grupos é menor que 5%. A partir deste ponto, inicia-se um processo de avaliação dos casos mais próximos do caso candidato em cada grupo com base no algoritmo de K-NN para que o caso candidato seja inserido no grupo que tenha mais semelhança com as suas características.

Todas as situações de adaptação em primeiro nível que aconteceram nas três execuções favoreceram à correta inserção do caso no seu grupo, devido em grande parte à maior distância existente entre centros dos diferentes grupos, o que acarreta poucas situações onde a compatibilidade entre o novo caso, o grupo mais provável e demais grupos seja menor que 5%.

A adaptação de segundo nível não foi utilizada aqui devido ao tipo do problema abordado, pois o protótipo objetiva a detecção e diagnóstico da falha,

onde não será necessária a solução da falha propriamente dita, que poderia ser encontrada através de operadores genéticos, como proposto no modelo do Capítulo 4.

5.6.1.4. Retenção (Inserção e Manutenção)

A retenção será dividida em duas partes, a inserção do caso e a manutenção da base de casos.

A inserção tradicional se dá no armazenamento do caso na base de casos. O modelo híbrido do RBC e SIA considera a inserção não somente como o armazenamento do caso mas, também, inclui a criação das conexões entre índice-índice e índice-caso do grupo em que o caso foi inserido. A criação desta conexão é de suma importância nas atividades de busca, manutenção e exploração do conhecimento armazenado na base de casos.

A criação das conexões entre os índices, neste experimento, se deu com o cálculo da média das distâncias entre todos os índices de um grupo, usando a expressão 4.4 $NAT - S$ apresentada no Capítulo 4. Esta expressão prevê o uso de um fator A , que trabalha para aumentar a restrição das conexões, ou seja, estas serão estabelecidas, se os pares índice-índice e índice-antígeno estiverem cada vez mais próximos. O termo F responde pela quantidade de desvios padrão adequada para posicionar o limiar em um nível mais aceitável para a aplicação e S é o desvio padrão propriamente dito. Os valores utilizados foram $A=1$, $F=0$ e $S=0$, mantendo neutros os efeitos de todos estes fatores. O motivo de utilização destes valores foi que os resultados, para o problema proposto e algoritmo de agrupamento utilizado, se mostraram mais satisfatórios se em comparação com outros valores iniciais para A , F e S . Ao final, a criação das conexões índice-índice foi resultado exclusivo da média das distâncias entre todos os índices de um mesmo grupo, onde as conexões foram criadas somente se a distância entre os índices fosse menor ou igual a esta média.

Já a criação da conexão índice-caso é realizada com base na proximidade entre o índice e o caso.

O resultado destes dois tipos de conexões são a malha primária (conexões índice-índice) e a malha secundária (conexões índice-caso), ambos utilizados para efetuar as buscas e explorar as áreas de conhecimento da base de casos. A Figura 5.35 mostra em detalhes, as conexões de algumas partes do grupo de falhas de corrente de armadura.

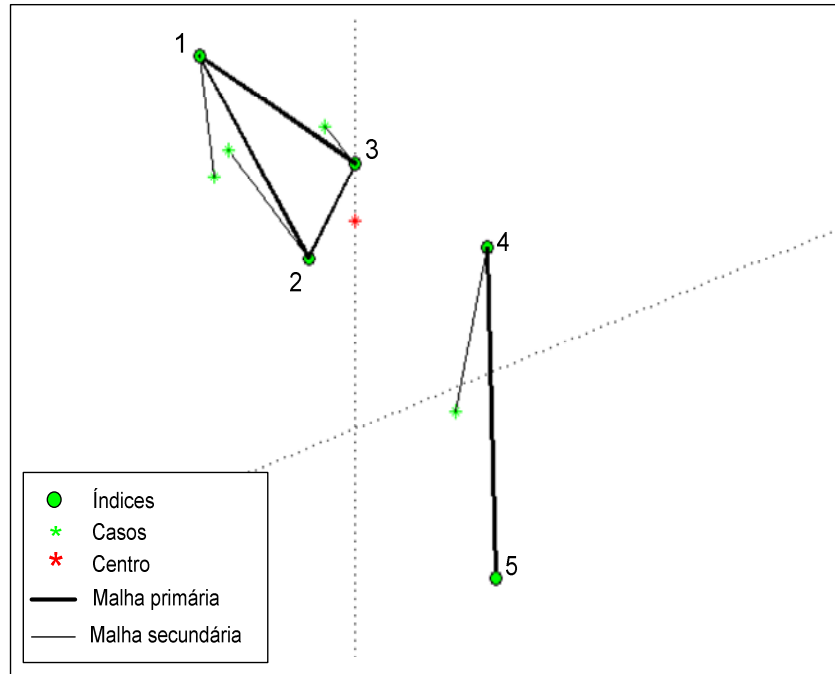


Figura 5.35 – Através da média das distâncias entre os índices, foram criadas as conexões 1-2, 1-3, 2-3 e 4-5, mas os índices 1, 2, 3 não possuem conexão com 4, 5, pois as distâncias destes grupos de índices são maiores que a média das distâncias dos índices do grupo. As linhas tracejadas tem por objetivo passar a noção de perspectiva em um espaço tridimensional.

Durante os experimentos percebeu-se que, no grupo com maior quantidade de casos, houve criação de conexões entre índices muito distantes, pois o grupo apresenta uma dispersão maior em relação aos demais grupos, o que aumenta a média das distâncias entre os índices. Estas conexões pouco representativas acabam por alocar esforço computacional na sua criação e também criar uma impressão distorcida da relação entre os índices.

Desta maneira, é importante que, a partir do tipo da aplicação e dos resultados do experimento, os valores dos fatores A , F e S possam ser alterados para que se crie uma rede de casos mais condizente com a representação dos dados, seja com mais ou menos conexões entre os índices, desde que o critério se baseie na análise dos dados e não em um valor arbitrário que não tem identificação com a distribuição dos casos.

É importante ressaltar que sempre que um caso é eleito para inserção na base de casos, a aiNet é executada no grupo ao qual o caso pertence. O objetivo é reposicionar os índices de tal forma que estes se adequem à distribuição considerando o novo caso. Se o grupo possuir uma distribuição abrangente, torna-se custoso executar a aiNet toda vez que um caso é inserido no grupo, já que existem índices neste grupo, que serão pouco afetados com a inserção de um novo caso.

Por sua vez, ao final das atividades de detecção e diagnóstico de falhas, buscas e inserção, a base de casos apresenta um número bem maior de casos em relação ao tamanho inicial, o que leva ao último estágio do experimento, a manutenção da base de casos.

A manutenção, no experimento, se dará na avaliação e remoção de possíveis casos desnecessários existentes nas áreas de maior densidade da base de casos. Isto significa que a técnica de remoção não vasculhará toda a base de casos, mas somente as áreas de maior probabilidade de ocorrência de casos considerados redundantes sob o ponto de vista da resolução (independente do seu tipo) de casos.

Como proposto no Capítulo 4, a avaliação da necessidade de se executar a manutenção em uma base de casos se dará a partir da percepção de que a recuperação de casos começa a retornar uma quantidade crescente de casos (cuja média está acima de determinado limiar) aptos a resolverem o problema ou situação candidata. Com base nas execuções deste experimento, o limiar escolhido é igual a 3. Isto significa que, a partir do momento em que a recuperação média de casos aptos for maior ou igual a 3 por 5 consultas consecutivas em pelo menos um agrupamento, o processo de manutenção é iniciado na base de casos.

O processo de manutenção continua com o levantamento, em cada grupo, do número de casos que cada índice referencia. A partir deste ponto, calcula-se para cada grupo a quantidade média de casos referenciados pelos índices e o desvio padrão por grupo desta quantidade. Os índices selecionados para a manutenção de cada grupo serão aqueles cuja quantidade de casos referenciada for maior ou igual ao desvio padrão calculado. A partir deste ponto, os casos ligados a cada índice são avaliados, segundo o princípio da competência, onde a cobertura e abrangência de cada caso são computadas com a finalidade de detectar possíveis casos auxiliares a serem eliminados. Todo o processo segue como no Algoritmo 4.7 e é executado para as três execuções. Todas as execuções utilizam 5 índices vizinhos para explorar a vizinhança à procura de possíveis casos auxiliares. Este número de vizinhos foi escolhido devido aos bons resultados obtidos nos experimentos de busca. As Figuras 5.36 a 5.41 apresentam os resultados.

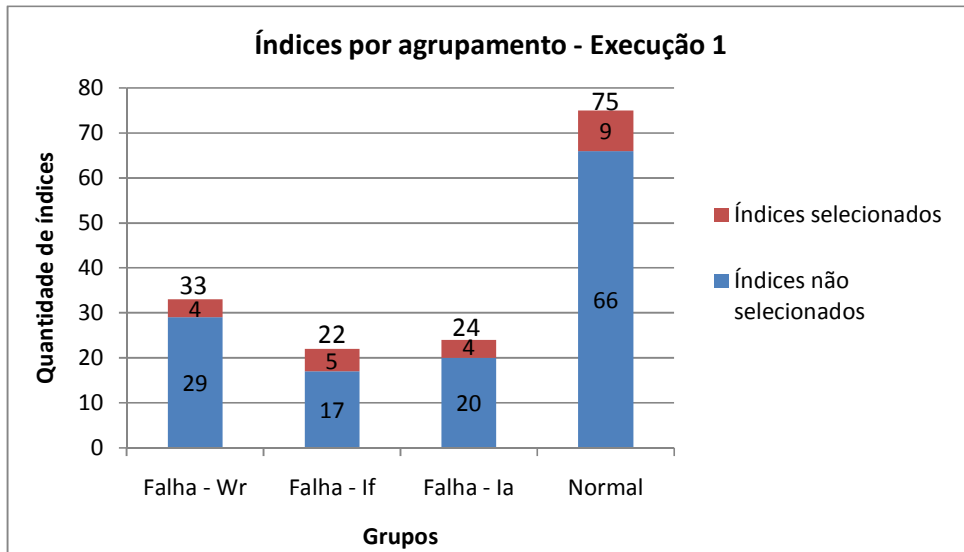


Figura 5.36 - Identificação de índices para realização da manutenção – execução 1.

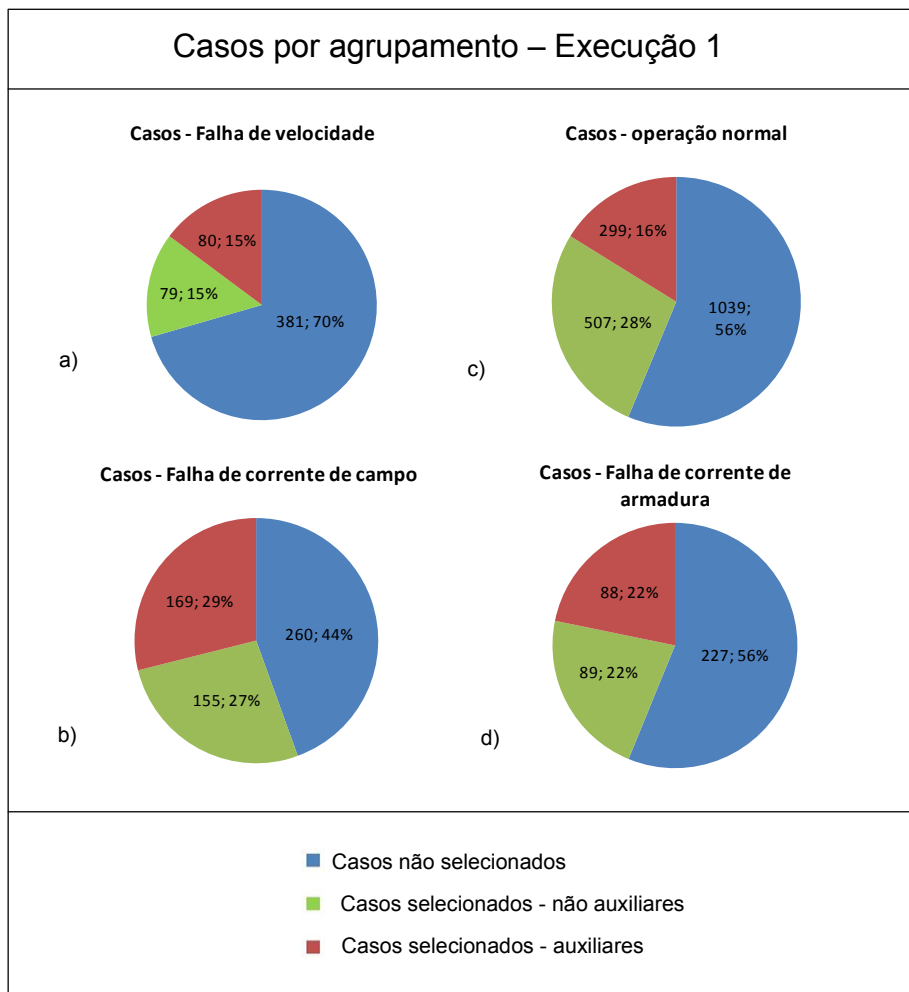


Figura 5.37 – Seleção de casos para eliminação da base de casos – execução 1.

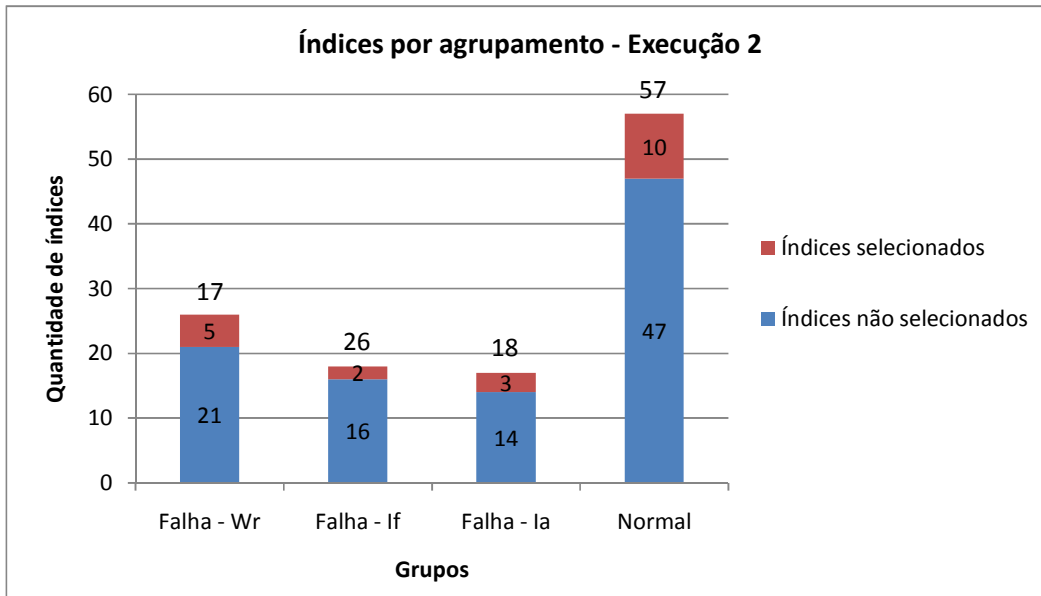


Figura 5.38 - Identificação de índices para realização da manutenção – execução 2.

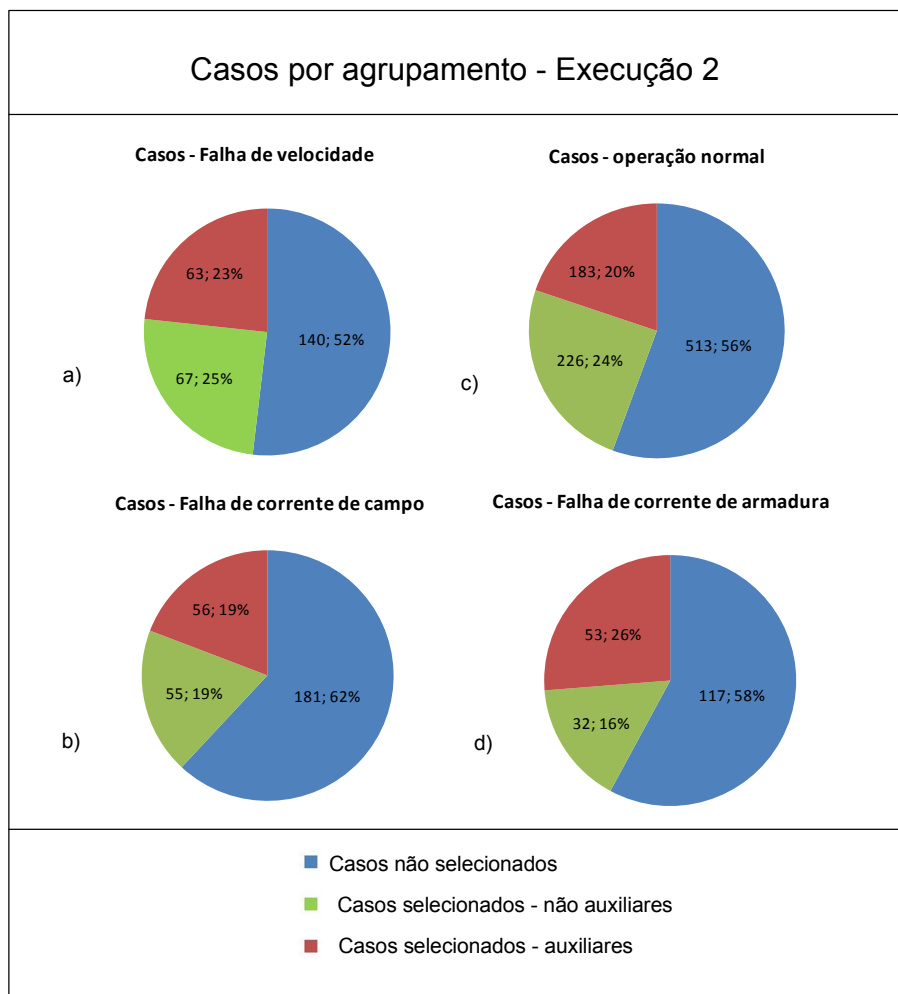


Figura 5.39 - Seleção de casos para eliminação da base de casos – execução 2.

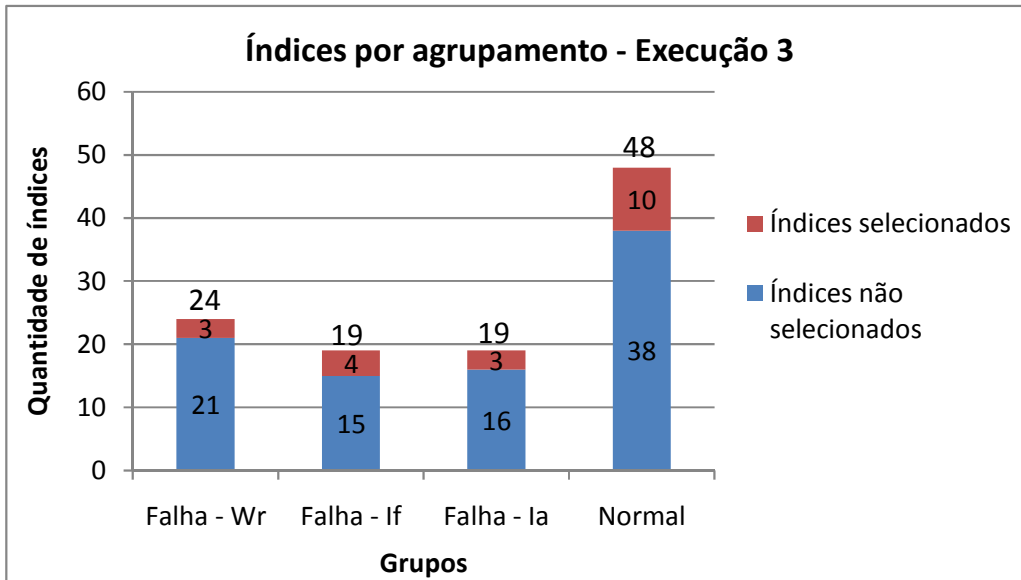


Figura 5.40 - Identificação de índices para realização da manutenção – execução 3.

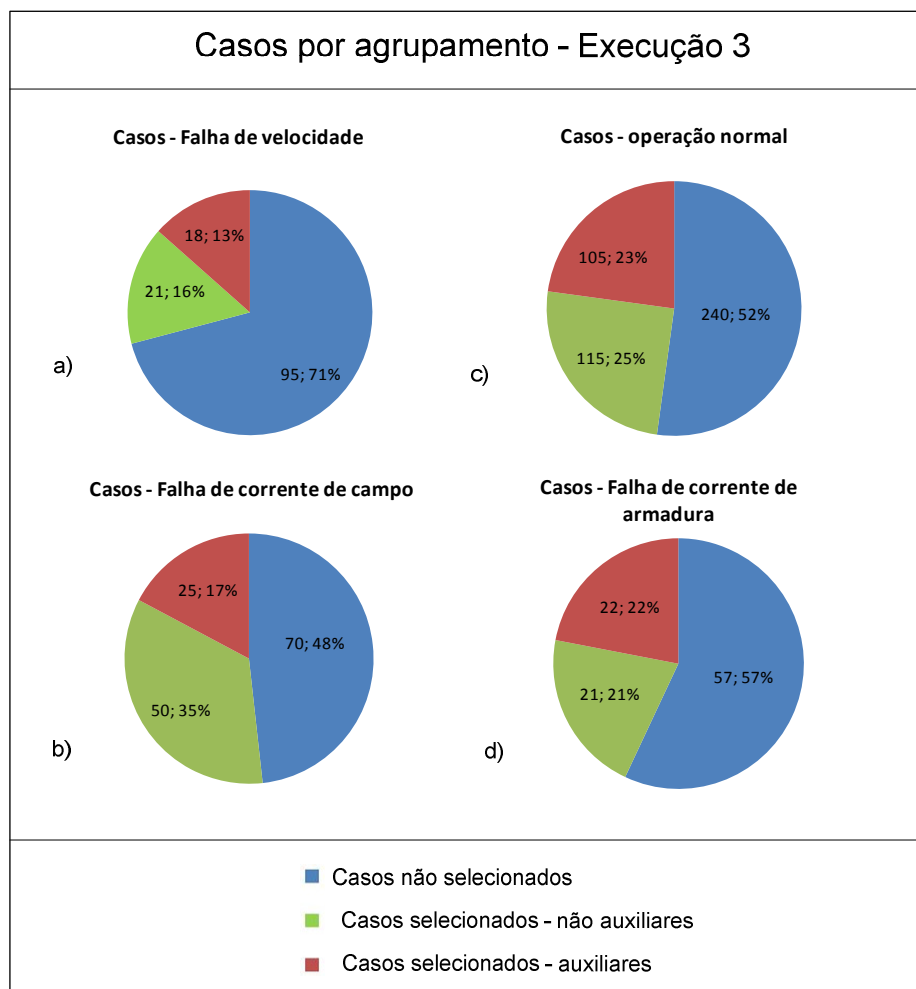


Figura 5.41 - Seleção de casos para eliminação da base de casos – execução 3.

Como pode ser constatado nas Figuras 5.36, 5.38 e 5.40, poucos índices são selecionados dentre os existentes nas bases de casos de todas as execuções. Ao mesmo tempo, estes poucos índices referenciam aproximadamente 50% dos casos na maioria das execuções e agrupamentos, como observado nas Figuras 5.37, 5.39 e 5.41. Isto significa que o trabalho de remoção de casos desnecessários, ou auxiliares, tem uma boa probabilidade de realmente eliminar estes possíveis casos, pois áreas de maior adensamento estão representadas pelos índices selecionados. De fato, observando a primeira execução na Figura 5.37, percebe-se que cada agrupamento contém entre 15% a 29% de casos auxiliares. Isto significa que existem casos na base de casos que não são representativos e que outros casos mais bem posicionados conseguem representar ou resolver as mesmas situações que os casos auxiliares, o que implica na remoção destes casos desnecessários dos agrupamentos. As Figuras 5.39 e 5.41 também mostram um comportamento semelhante em relação aos casos auxiliares para as outras duas execuções. A Figura 5.42 mostra os tamanhos antes e depois da remoção dos casos auxiliares nas bases de casos em cada uma das três execuções. Como já colocado no Capítulo 4, a política de remoção de casos baseada nos casos auxiliares garante que somente casos desnecessários são removidos da base de casos, ao contrário de outros critérios, como a escolha aleatória (Smyth e Keane 1995) e baseada no estímulo de casos (Hunt, Cooke e Holstein 1995), que terminam por comprometer a competência de um sistema de RBC.

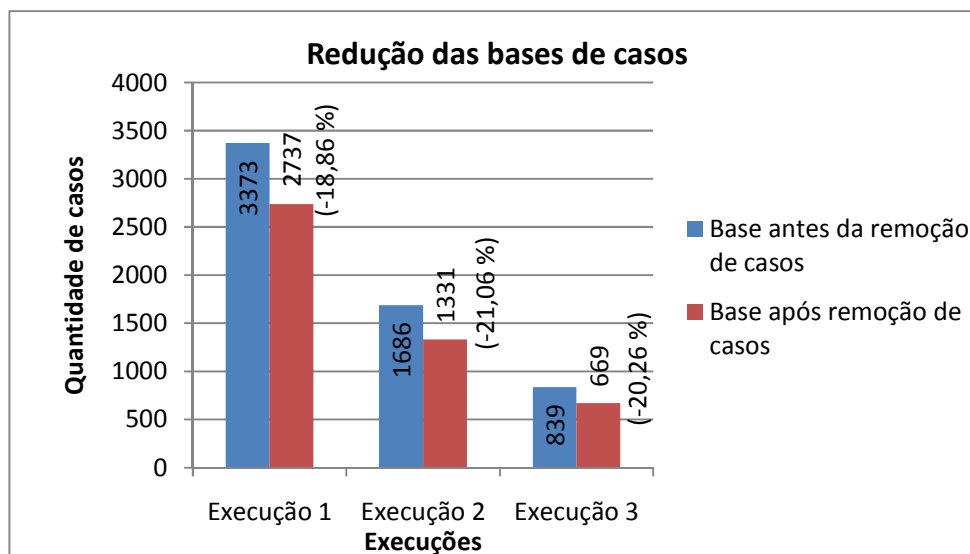


Figura 5.42 – valores e taxas de redução da base de casos após remoção dos casos auxiliares.

Entende-se que os resultados obtidos são relativos ao problema específico abordado e à conseqüente conformação dos casos na base de casos e que outros problemas podem não apresentar a mesma proporção de casos auxiliares. Entretanto, acredita-se que o princípio de procurar por casos desnecessários nas áreas mais adensadas é válido, pois quanto maior o adensamento, maior a probabilidade de se encontrarem casos semelhantes e possivelmente auxiliares que, quando eliminados, contribuirão para o aumento

da eficiência das buscas no processo de recuperação em bases de casos, o que é significativo para grandes bases de casos.

Apesar da rede de casos ser concebida neste trabalho, objetivando apoiar os processos do ciclo de vida de um sistema de RBC, esta pode apresentar características interessantes para o campo da mineração de dados, ou *data mining* (Hunt e Fellows 1996). O motivo é que a rede de casos é resultado do posicionamento de índices em uma base de casos de tal modo que sejam representativos da sua distribuição, estabelecendo uma relação de afinidade entre outros índices e os casos da base de casos, o que, sob o ponto de vista da mineração de dados, pode ser interpretado como a identificação de diferentes áreas de conhecimento e como estas áreas se relacionam, possibilitando a descoberta de aglomerados ou relações ainda não observadas e de utilidade para a aplicação de um sistema RBC.

5.6.2. Segunda Modalidade

A segunda modalidade dos experimentos consta exclusivamente da avaliação da capacidade de detecção e diagnóstico dos casos de acordo com o seu padrão, sejam eles normais, de falha na corrente de armadura, falha na corrente de campo ou falha de velocidade. A diferença em relação à primeira modalidade se dá no ponto em que somente uma pequena quantidade de casos de operação normal será inicialmente fornecida ao protótipo e diagnosticada como tal. A partir daí, diversos casos desconhecidos ao protótipo serão apresentados aleatoriamente, sejam de operação normal ou falha. Este tipo de experimento não tem o objetivo de avaliar o desempenho da busca, adaptação, inserção ou manutenção de uma base de casos, pois considera-se que estes já foram tratados nos experimentos da primeira modalidade.

A base de casos utilizada é a mesma base de casos da primeira modalidade, terceira execução, com 12,5% dos casos da base de casos original da Tabela 5.1. A razão do uso desta base de casos é que ela é menos densa em relação às outras, dificultando assim o processo de detecção e diagnóstico de casos, já que o protótipo inicia o seu funcionamento desconhecendo completamente qualquer caso de falhas.

O resultado obtido foi que 100% dos casos foram corretamente detectados e diagnosticados, indicando que o protótipo conseguiu através do algoritmo de agrupamento participativo e da aiNet, utilizando o centro dos anticorpos, e não dos casos, identificar um padrão diferente para cada um dos agrupamentos de falha e comportamento normal.

Dado que a segunda modalidade dos experimentos inicia-se somente com alguns casos da operação normal, à medida que novos agrupamentos eram detectados pelo algoritmo, um supervisor realizava o diagnóstico segundo as seguintes possibilidades: comportamento normal, falha de corrente de campo, armadura ou velocidade. Este comportamento é diferente dos experimentos da primeira modalidade, que possuem uma etapa de treinamento onde as soluções são fornecidas constantemente para ajuste dos indicadores do algoritmo. A partir do diagnóstico do supervisor para um novo grupo na

segunda modalidade, casos subsequentes são detectados e diagnosticados, utilizando o conhecimento recém incorporado ao sistema. É importante citar que somente 3 intervenções, Figuras 5.43 a 5.44, do supervisor foram necessárias em todo o experimento da segunda modalidade, uma para cada tipo de falha, um indicativo que o protótipo foi capaz de detectar padrões diferentes, ou novidades, existentes nos casos.

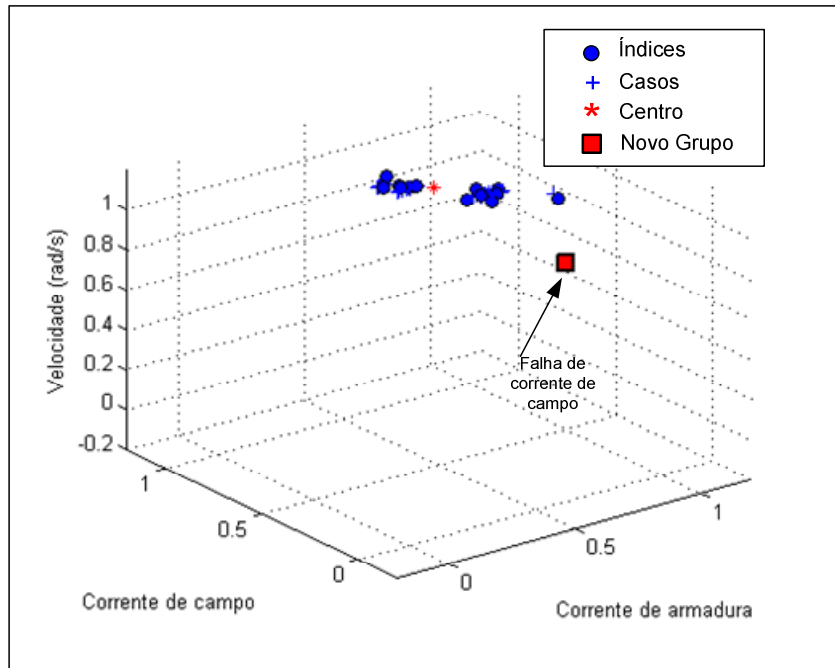


Figura 5.43 – Diagnóstico da falha de corrente de campo pelo supervisor.

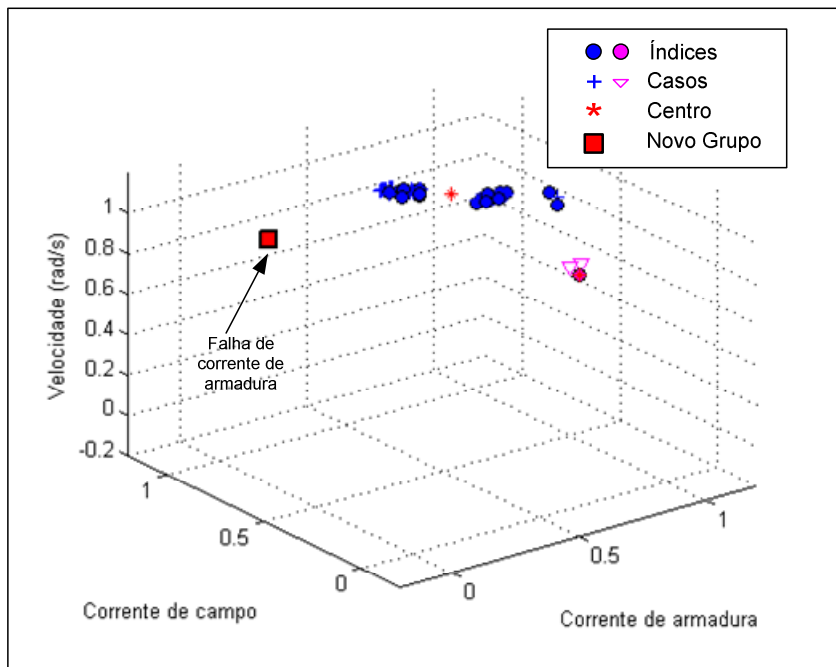


Figura 5.44 - Diagnóstico da falha de corrente de armadura pelo supervisor.

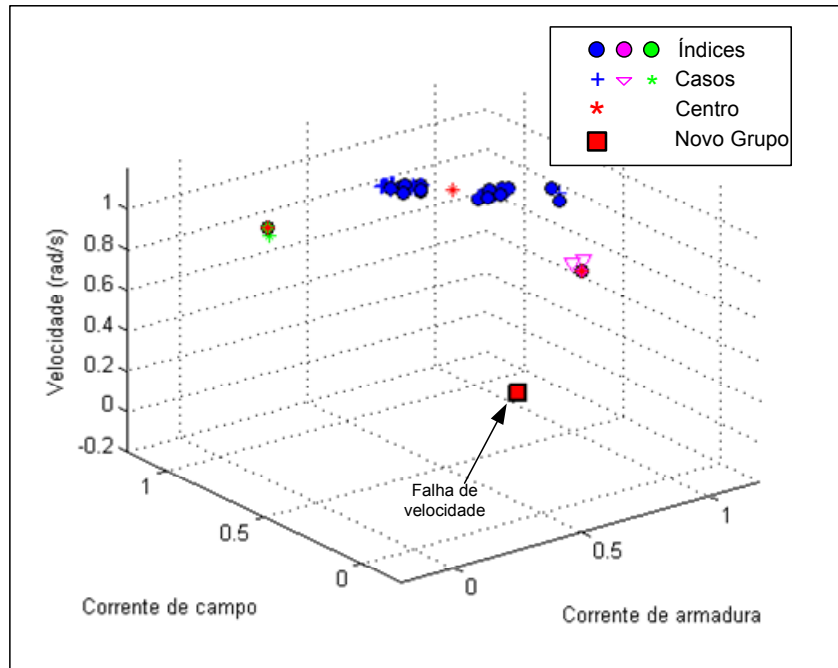


Figura 5.45 - Diagnóstico da falha de velocidade pelo supervisor.

É relevante colocar que o algoritmo de agrupamento utilizado nas duas modalidades dos experimentos é um algoritmo sensível à ordem da entrada de dados, ou seja, dependendo da seqüência de casos apresentados, os agrupamentos se organizarão de maneira diferente em relação a outras execuções, podendo até mesmo não apresentar o mesmo resultado final, indicando incoerência em relação ao padrão (agrupamentos) natural dos casos. Antes de se obter um conjunto ótimo de parâmetros do algoritmo de agrupamento, foram realizados diversos testes, os quais confirmaram a sensibilidade do algoritmo à ordem de entrada dos casos, mas ele foi mantido mesmo assim, devido à sua característica de tratamento ponto a ponto e à amplitude das operações com grupos. Os testes preliminares resultaram na formação de grupos de casos não condizentes com seu padrão natural, o que pode ser um problema para situações que possuam agrupamentos não isotrópicos (formas diferentes) e sobreposição de casos em grupos vizinhos. Assim, os problemas passíveis de formação de grupos de casos devem ter o seu algoritmo de agrupamento escolhido com base nas suas características, o que aumentará a probabilidade de sucesso na formação de grupos coerentes com os casos e a conseqüente melhora da eficiência na resolução de novos casos.

5.7. Principais dificuldades, limitações e avanços

A conjugação do RBC e SIA apresentou diversos desafios, descritos nos pontos a seguir:

- Algoritmo de agrupamento: como a detecção e diagnóstico de falhas são realizados ponto a ponto, foi necessária a seleção de um algoritmo que também tratasse ponto a ponto as entradas. Isto significa que a partir da leitura de cada ponto, o algoritmo realizava o seu agrupamento, podendo, ao longo do tempo, até mesmo alterar o grupo do ponto previamente classificado. Para que fosse obtida uma execução parcimoniosa, resultante da apresentação aleatória dos pontos e ao mesmo tempo da necessidade de detectar e diagnosticar corretamente os pontos apresentados, diversos parâmetros do algoritmo de agrupamento participativo foram refinados, como o β , α , τ e λ . A dificuldade para estabilizar (conseguir que ele fosse executado várias vezes com o mesmo resultado final) o algoritmo foi maior devido à utilização dos índices (em menor número que os casos), ou anticorpos, para realização dos cálculos de agrupamento. Outro quesito que merece atenção é a sensibilidade do algoritmo de agrupamento à ordem dos pontos apresentados, constituindo-se de mais um fator que influenciou no refinamento detalhado dos parâmetros supracitados. Ao final, o resultado positivo indicou que o uso dos anticorpos como entrada para processamento do algoritmo foi satisfatório, classificando corretamente os pontos de operação como falha (e qual delas) ou comportamento normal. A dificuldade de ajuste dos parâmetros sugere a seleção de outros algoritmos de agrupamentos que não sejam tão sensíveis à ordem de apresentação. Cada problema deve implicar na seleção de um algoritmo de agrupamento específico.
- Distribuição e quantidade dos índices: a execução da aiNet, que resultou no posicionamento representativo dos anticorpos, ou índices, no espaço de casos, apresentou uma alta taxa de compressão em relação aos casos, mas a um custo quadrático da execução do cálculo de afinidade entre os anticorpos. À medida que o número de casos de um agrupamento aumenta, maior é o tempo de execução da aiNet neste agrupamento, podendo tornar-se inviável em plataformas computacionais com poucos recursos. Uma alternativa foi aperfeiçoar os parâmetros da aiNet para que houvesse um equilíbrio entre a quantidade dos anticorpos e sua distribuição. Mesmo esta abordagem pode ser tornar custosa se o agrupamento for muito grande, necessitando remover casos muito semelhantes ou redundantes.
- Conjugação de RBC e SIA: mesmo com a existência de um trabalho de treze anos atrás correlacionando as áreas, não existia até o atual momento um trabalho que pontuasse os avanços individuais dos dois paradigmas com o objetivo de serem utilizados de forma sinérgica. Sob este ponto de vista, esta dissertação foi um esforço de retomada do tema para que uma nova conformação pudesse ser apresentada e outros trabalhos possam ser desenvolvidos com menor esforço.

Sob o ponto de vista dos avanços, os mais significativos estão expostos a seguir:

- Algoritmo de agrupamento: o uso de um algoritmo de agrupamento em caráter misto com o RBC e SIA abre possibilidades para que outros algoritmos tradicionalmente eficazes em suas áreas de aplicação possam ser adaptados para funcionar com uma implementação híbrida de RBC e SIA.
- Busca mais eficaz: a busca proposta no trabalho de Hunt era aleatória, o que torna o método inviável em uma grande base de casos, ao passo que a busca aqui proposta mapeia o espaço de casos para que os índices possam estabelecer pontos de referência e permitam aprofundar a busca sem um esforço exaustivo. Para facilitar o mecanismo, a base de casos é dividida em grupos, que ajudam a particionar mais ainda a o processo de busca e recuperação de casos.
- Exploração do espaço: com o posicionamento dos índices em regiões representativas dos casos, torna-se possível buscar por relações de conhecimento entre áreas contíguas da base de casos, potencializando a descoberta de novas interações entre os casos e seus agrupamentos.
- Redução com precisão: à medida que os casos são resolvidos e apresentados ao sistema, a base de casos aumenta em tamanho, mas não necessariamente em diversidade. Os métodos utilizados de eliminação de casos desnecessários garantem o expurgo de casos menos representativos, e por conseqüência, o aumento da eficiência do processo de busca.
- Mapeamento do ciclo de vida do RBC com SIA: para que seja formada uma idéia integrada da utilização do RBC e SIA, foram apresentadas contribuições em cada um dos processos de RBC, desde a recuperação até a retenção, que culmina na retenção dos casos e de suas relações com os casos correlatos.

Capítulo 6 - Conclusão e Perspectivas Futuras

O RBC é um poderoso método de resolução de problemas baseado em um aspecto específico de como os humanos resolvem problemas, reaproveitando experiências egressas para sua solução. O RBC não propõe algoritmos de recuperação, reutilização, revisão e retenção de casos mas, sim, uma metodologia para a resolução de problemas baseada na utilização do conhecimento adquirido, estando, desta maneira, aberto aos mais diferentes métodos, técnicas ou ferramentas que melhor possam ajudar a cumprir seu objetivo. Isto implica que o RBC pode e deve ser alvo de estudo de paradigmas de *Soft Computing* e Inteligência Computacional, pois demanda esforços em pontos como formação da memória, recuperação da informação, aprendizado de máquina e reconhecimento de padrões, dentre outros.

Este trabalho endereçou essa demanda através da apresentação inicial do Raciocínio Baseado em Casos (RBC) e do Sistema Imune Artificial (SIA) como dois métodos independentes de Inteligência Computacional. Em seguida, foi proposto e implementado um modelo híbrido dos dois métodos, de tal modo que o RBC pudesse usufruir de avanços nas pesquisas de SIA, desde a publicação dos trabalhos precursores que relacionaram as duas áreas (Hunt, Cooke e Holstein 1995) (Hunt e Fellows 1996). O objetivo desta abordagem conjunta é o de fornecer um leque de novas possibilidades para que o RBC possa abordar problemas e situações de modo mais eficiente e preciso em seu ciclo de vida.

Os resultados obtidos no protótipo híbrido do RBC e SIA indicam um caminho positivo a ser trilhado por esta abordagem, já que a busca, um dos processos mais usados e custosos do RBC, teve sua eficiência sensivelmente melhorada pela divisão dos casos em grupos com índices mapeando as suas distribuições, o que apresenta desdobramentos favoráveis na recuperação, inserção e manutenção de casos. Mesmo que este trabalho se baseie fortemente no agrupamento de casos, bases de casos que formem um único grupo também podem fazer uso deste modelo híbrido, visto que a execução da aiNet resultaria em uma base indexada de casos de grupo único. Em seguida, os casos seriam recuperados utilizando os índices, recém posicionados no espaço junto com suas relações, aumentando assim a eficiência da busca, pois se evita uma busca completa na base de casos.

É importante ressaltar que o algoritmo de agrupamento é selecionado com base no problema alvo e na sua capacidade de adaptação para funcionar

em conjunto com a aiNet. O algoritmo de agrupamento participativo foi escolhido com base na sua característica de aprendizado, ponto a ponto, e capacidade de criação de agrupamentos de maneira flexível, utilizando indicadores como a competência e o alerta. Deve-se ter em mente que o SIA e seus algoritmos aqui utilizados não competem com algoritmos de agrupamentos estabelecidos, mas tem com eles uma relação de complementaridade, beneficiando-se do melhor que eles possam oferecer ao mesmo tempo em que o SIA fornece os anticorpos como entrada para seu processamento. Deste modo, assim como a escolha do algoritmo de agrupamento é dependente do problema, também o são as técnicas de recuperação, adaptação e retenção derivadas do SIA, já que cada situação apresenta a sua especificidade, como distribuição de casos, tipo de solução, adaptação e critérios de armazenamento.

Diversos problemas de RBC demandam soluções mais complexas, além das contidas no experimento aqui apresentado, mas o arcabouço proposto neste trabalho lança as bases para que técnicas presentes no SIA como operadores genéticos e a rede imune sejam utilizadas para explorar o espaço de soluções, identificar áreas de conhecimento diferenciado, prover facilidades na busca, exploração de casos e manutenção da base de casos.

Retornando ao início deste trabalho, no tocante à fonte inspiradora do RBC, na psicologia cognitiva, é pertinente colocar que o suporte fornecido pela aiNet para o RBC é semelhante à estrutura da memória dinâmica proposta por Schank (1982), onde os casos são representados pelos antígenos, as estruturas genéricas de memória pelos anticorpos, funcionando como índices, e as conexões entre os casos da memória análogas às conexões da rede imunológica, conectando anticorpos com antígenos, ou índices com casos, respectivamente. Assim como a memória dinâmica se adapta constantemente aos novos estímulos, com o objetivo de absorver e reorganizar o conhecimento adquirido, a rede imunológica reposiciona seus anticorpos e altera suas conexões para adequar-se aos estímulos externos representados pelos novos casos apresentados.

As contribuições presentes neste trabalho tiveram o objetivo de criar um modelo híbrido de RBC e SIA que focasse mais a amplitude que a profundidade, o que naturalmente abre espaços para melhorias e propostas de trabalhos futuros. Dentre as melhorias e propostas identificadas, até o momento, estão:

1. Uso dos mecanismos de defesa em camadas do sistema imune: assim como o sistema imune biológico apresenta seus mecanismos de defesa em camadas, começando pelas barreiras físicas, bioquímicas, sistema inato e terminando no adaptativo, o modelo híbrido de RBC e SIA também pode usufruir desta característica. Uma abordagem diferente seria a seleção dos casos mais representativos da base de casos como integrantes do modelo defesa inato (macrófagos e granulócitos), sendo que cada novo caso seria primeiro apresentado a estes casos representativos. Se estes casos fossem capazes de resolver o novo caso, uma busca completa no grupo ou na base de casos seria evitada, melhorando a eficiência do sistema.

1. Execução setorizada da aiNet nos agrupamentos: mesmo com o aumento significativo da eficiência através da indexação de casos com anticorpos e a segmentação da busca por grupos, o processo de recuperação ainda pode se mostrar lento, se a aiNet for executada em agrupamentos com muitos casos. Uma opção para minimizar esta limitação é executar a aiNet em um setor ou área específicos do agrupamento e não integralmente, já que a posição do novo caso no espaço geralmente não gera um distúrbio na vizinhança dos casos e índices distantes dele.
2. Avaliação de algoritmos e mudança no cálculo de seus parâmetros: a abordagem aqui realizada da aiNet optou por efetuar os cálculos dos índices do algoritmo de agrupamento, utilizando os anticorpos, ou índices, gerados na sua execução, o que apresentou resultados favoráveis ao problema abordado. Mais experimentos são necessários com outros tipos de algoritmos de agrupamento para mostrar o seu potencial de uso com a aiNet. Também é importante realizar a avaliação dos resultados, partindo do ponto que os parâmetros do algoritmo de agrupamento seriam obtidos dos casos e não dos anticorpos gerados pela aiNet, podendo caber aos anticorpos somente o papel de índice nas buscas.
3. Exploração direcionada do espaço: o espaço de casos possui regiões povoadas e não povoadas de casos. A exploração das áreas povoadas é natural, visto que se pretende encontrar casos na base de casos que forneçam a solução para um caso candidato. A pesquisa proposital em áreas não povoadas (Pal e Shiu 2004) apresenta um interessante potencial para a descoberta de regiões do espaço que podem ser extremamente relevantes ou que não fazem sentido algum. O resultado positivo pode potencializar uma alteração no sistema de RBC e, até mesmo, fomentar a fabricação de casos nesta região do espaço, ao passo que a identificação de áreas irrelevantes pode ser útil para descartar a análise de algum caso que possua as características desta região de interesse periférico.
4. Aplicação do modelo proposto em casos que demandem uma solução explícita: o experimento realizado abordou um problema de detecção e diagnóstico, mas que pode desdobrar-se e abranger a resolução da falha encontrada, o que demandaria o uso mais sofisticado das atividades de adaptação propostas.
5. Adaptação conjugada: a adaptação é uma característica importante de RBC, pois nem sempre os casos encontrados possuem as características necessárias para prover uma solução direta, mas podem ser utilizados de maneira gradativa para fornecer a solução. Sob o ponto de vista do SIA e como colocado neste trabalho, a mutação é uma opção interessante para lidar com a adaptação, já que ela trabalha na exploração do espaço, o que pode levar a situações diferentes, favoráveis ou não. Para que o processo seja controlado, é importante que o controle ou função objetivo sejam utilizados em separado ou em conjunto com um modelo genérico do domínio, com a

meta de fornecer as direções para os caminhos encontrados por este tipo de adaptação. Nada impede que o modelo híbrido do SIA deixe de utilizar a adaptação por mutação ou a conjugue com outro método que se julgar necessário para que o objetivo do sistema de RBC seja cumprido, pois ele ainda poderá fazer uso de outras características relevantes, como índices e conexões da rede de casos. Quanto maior o emprego de opções com potencial para encontrar soluções efetivas e minimizar intervenções humanas de forma segura, mais completo será um sistema RBC, caminhando em direção à criação de uma máquina inteligente que seja capaz de resolver problemas e aprender com sua própria experiência.

Referências Bibliográficas

Aamodt, A., e E. Plaza. "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches." *AI Communications*, Março de 1994: 39-59.

Aha, D. W., D. Kibler, e M. K. Albert. "Instance-Based learning algorithms." *Machine Learning*, 1991: 37-66.

Aickelin, U., e J. Greensmith. "Sensing danger: Innate immunology for intrusion detection." *Information Security Technical Report*, 2007: 218-227.

Aickelin, U., e S. Cayzer. "The Danger Theory and Its Application to Artificial Immune Systems." *1st International Conference on Artificial Immune Systems (ICARIS-02)*. Canterbury, 2002. 141-148.

Aickelin, U., P. Bentley, S. Cayzer, J. Kim, e J. McLeod. "Danger Theory: The Link between AIS and IDS?" *2st International Conference on Artificial Immune Systems (ICARIS-2003)*. Springer, 2003. 147-155.

Althoff, K., e R. O. Weber. "Knowledge management in case-based reasoning." *The Knowledge Engineering Review*, 2005: 305-310.

Amit, K. *Computational Intelligence: Principles, Techniques and Applications*. 1a. Edição. Berlin: Springer, 2005.

Ayara, M., J. Timmis, L. de Lemos, L. de Castro, e R. Duncan. "Negative selection: How to generate detectors." *1st International Conference on Artificial Immune Systems (ICARIS-02)*. Canterbury, 2002. 89-98.

Behn, U. "Idiotypic networks: toward a renaissance?" *Immunological Review*, 2007: 142-152.

Bergmann, Ralph. *Experience Management: Foundations, Development Methodology, and Internet-Based Applications*. 1a. Edição. Berlin: Springer, 2002.

Bergmann, Ralph. *Introduction to Case-Based Reasoning*. Kaiserslautern, 2000.

Bezdek, J. C. "What is computational intelligence?" *Computational Intelligence: Imitating Life*, 1994: 1-12.

Bezdek, J. C., e S. K. Pal. "Fuzzy Models for Pattern Recognition." 1992: 1-27.

Bezerra, G. B., T. V. Barra, H. M. Ferreira, H. Knidel, L. N. de Castro, e F. J. Von Zuben. "An immunological filter for SPAM." *5th International Conference on Artificial Immune Systems (ICARIS-2006)*. Springer, 2006. 446-458.

Borner, K. "Structural Similarity as Guidance in Case-Based Design." *Topics in Case-Based Reasoning*, 1994: 197-208.

Bridge, D., M. Goker, L. McGinty, e B. Smyth. "Case-based recommender systems." *The Knowledge Engineering Review*, 2005: 315-320.

Calich, V. L.G., e C. A.C. Vaz. *Imunologia*. 1a. Edição. Rio de Janeiro: Revinter, 2001.

Caminhas, W. M. "Estratégias de Detecção e Diagnostico de Falhas em Sistemas Dinâmico." Tese de Doutorado, Engenharia Elétrica, Universidade Estadual de Campinas (UNICAMP), 1997.

Cardie, C. "Using Decision Trees to Improve Case-Based Learning." *Proceedings of the Tenth International Conference on Machine Learning*, 1993: 25-32.

Cheetham, W., e I. Watson. "Fielded applications of case-based reasoning." *Knowledge Engineering Review*, 2005: 321-323.

Cheetham, W., e J. Graf. "Case-based reasoning in colour matching." *2th International Conference on Case-Based Reasoning (ICCBR-97)*. Springer Verlag, 1997. 1-12.

Cheetham, W., Simon C.K. Shiu, e Rosina O. Weber. "Soft Case-Based Reasoning." *Knowledge Engineering Review*, 2005: 255-260.

Coutinho, A. "Will the idiotypic network help to solve natural tolerance?" *TRENDS in Immunology*, Fevereiro de 2003: 53-54.

Cox, M. T., H. Muñoz-Avila, e R. Bergmann. "Case-based planning." *The Knowledge Engineering Review*, 2005: 277-281.

Craenen, B. C., e A. E. Eiben. "Artificial Intelligence: Definition, Trends, Techniques and Cases - Computational Intelligence." *Encyclopedia of Life Support System*. 2003. <http://www.eolss.net> (acesso em Outubro de 2008).

da Silva, L. R.S. "Aprendizagem Participativa em Agrupamento Nebuloso de Dados." Dissertação de Mestrado, Engenharia Elétrica e Computação, Universidade Estadual de Campinas (UNICAMP), 2003.

Dasgupta, D. *Artificial Immune Systems and Their Applications*. Springer-Verlag, 1999.

Dasgupta, D., e S. Forrest. "Artificial immune systems in industrial applications." *2st International Conference on Intelligent Processing and Manufacturing of Materials (IPMM-99)*. IEEE press, 1999. 257-267.

de Castro, L. N. "Engenharia Imunológica: Desenvolvimento e Aplicação de Ferramentas Computacionais Inspiradas em Sistemas Imunológicos Artificiais." Tese de Doutorado, Engenharia Elétrica, UNICAMP, Campinas, 2001.

de Castro, L. N., e F. J. Von Zuben. "An Evolutionary Immune Network for Data Clustering." *Brazilian Symposium on Artificial Neural Networks (IEEE SBRN-2000)*, 2000: 84-89.

—. "The clonal selection algorithm with engineering applications." *Genetic and Evolutionary Computation Conference (GECCO-2000)*. Las Vegas: Morgan Kaufmann, 2000. 36–37.

de Castro, L. N., e J. Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach*. 1a. Edição. Springer-Verlag, 2002.

de Lemos, R., J. Timmis, S. Forrest, e M. Ayara. "Immune-inspired adaptable error detection for automated teller machines." *IEEE Trans. on Systems, Man and Cybernetics Part C: Applications and Reviews*, 2007: 873–886.

de Lima, E. M. "Modelagem Fuzzy Funcional Evolutiva Participativa." Dissertação de Mestrado, Engenharia Elétrica, Universidade Estadual de Campinas (UNICAMP), 2008.

Devi, V. S., e M. N. Murty. "An incremental prototype set building technique." *Pattern Recognition*, 2002: 505-513.

Encyclopaedia Britannica. *Immune System*. <http://www.britannica.com/EBchecked/topic/283636/immune-system/215576/Activation-of-T-and-B-lymphocytes#ref=ref708694> (acesso em Outubro de 2008).

Engelbrecht, A. P. *Computational Intelligence: An Introduction*. 2a. Edição. Chichester: Wiley & Sons, 2007.

Farmer, J. D., N. H. Packard, e A. S. Perelson. "The immune system, adaptation and machine learning." *Physica*, 1986: 187–204.

Farmer, J. D., S. A. Kauffman, N. H. Packard, e A. S. Perelson. "Adaptive dynamic networks as models for the immune system and auto-catalytic sets." *Annals of the New York Academy of Sciences*, 1987: 118–131.

Fogel, D. B. *Evolutionary computation: toward a new philosophy of machine intelligence*. 3a. Edição. Hoboken: John Wiley and Sons, 2006.

Forgy, E. "Cluster analysis of multivariate data: Efficiency vs. interpretability of classifications." *Biometrics*, 1965: 768-780.

Forrest, S., A. S. Perelson, L. Allen, e R. Cherukuri. "Self-Nonself discrimination in a computer." *IEEE Symposium on Research in Security and Privacy*. Los Alamos: IEEE Computer Society Press, 1994. 202–212.

Freitas, A. A., e J. Timmis. "Revisiting the foundations of artificial immune systems for data mining." *IEEE Transactions on Evolutionary Computation*, 2007: 521–540.

Fulcher, J., e L. C. Jain. *Computational Intelligence: A Compendium*. 1a. Edição. Berlin: Springer, 2008.

Gabel, T., e A. Stahl. "Exploiting background knowledge when learning similarity measures." *7th European Conference on Case-Based Reasoning (ECCBR-04)*. Berlin: Springer, 2004. 169-183.

Garain, U. "Prototype reduction using an artificial immune model." *Pattern Analysis and Applications*, 2008: 353-363.

García-Pedrajas, N., e C. Fyfe. "Immune network based ensembles." *Neurocomputing*, 2007: 1155–1166.

Garrett, S. "How do we evaluate artificial immune systems?" *Evolutionary Computing*, 2005: 145–178.

Gath, I., e A. B. Geva. "Unsupervised Optimal Fuzzy Clustering." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1989: 773-781.

Goker, M. H., R. J. Howlett, e J. E. Price. "Case-based reasoning for diagnosis applications." *The Knowledge Engineering Review*, 2005: 277-281.

Guzella, T. S., T. A. Mota-Santos, e W. M. Caminhas. "Regulatory T Cells: Inspiration for Artificial Immune Systems." *6st International Conference on Artificial Immune Systems (ICARIS-07)*. Santos: Springer, 2007. 312-323.

Guzella, T. S., T. A. Mota-Santos, J. Q. Uchôa, e W. M. Caminhas. "Identification of SPAM messages using an approach inspired on the immune system." *BioSystems*, 2008: 215-225 .

Hart, E., e J. Timmis. "Application areas of AIS: The past, the present and the future." *Applied Soft Computing*, 2008: 191–201.

Hart, E., H Bersini, e F. C. Santos. "How affinity influences tolerance in an idiotypic network." *Journal of Theoretical Biology*, 2007: 422–436.

Hart, Peter E. "The Condensed Nearest Neighbor Rule." *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, 1968: 515-516.

Hennessy, D., e D. Hinkle. "Applying case-based reasoning to autoclave loading." *IEEE Expert*, 1992: 21-26.

Hoffmann, G. W. "A neural network model based on the analogy with the immune system." *Journal of Theoretical Biology*, 1986: 33–67.

Hoffmann, G. W., M. W. Benson, G. M. Bree, e P. E. Kinahan. "A teachable neural network based on an unorthodox neuron." *Physica*, 1986: 233-246.

Hofmeyr, S. A., e S. Forrest. "Architecture for an Artificial Immune System." *Evolutionary Computation*, 2000: 45-68.

Holloway, Todd. *A Beautiful WWW*. 17 de Agosto de 2007. <http://abeautifulwww.com/2007/08/17/iccbr-2007-highlights> (acesso em 2008).

Holt, A., I. Bichindaritz, R. Schmidt, e P. Perner. "Medical applications in case-based reasoning." *The Knowledge Engineering Review*, 2005: 289-292.

Hunt, John E., Denise E. Cooke, e Horst Holstein. "Case Memory and Retrieval Based on the Immune System." *1th International Conference on Case-Based Reasoning (ICCB-95)*. Sesimbra: Springer, 1995. 205-216.

Hunt, John E., e A. Fellows. "Introducing an Immune Response into a CBR System for Data Mining." 1996: 35-42.

Jain, A. K., M. N. Murty, e P. J. Flynn. "Data clustering: A review." *ACM Computing Surveys*, 1999: 264-323.

Janeway, C.A., P. Travers, M. Walport, e J. D. Capra. *Imunobiologia: O Sistema Imunológico na Saúde e na Doença*. 4a. Edição. Artes Médicas, 2000.

Jang, J. -S., C. -T. Sun, e E. Mizutani. *Neuro-Fuzzy and Soft Computing*. 1a. Edição. Prentice Hall, 1997.

Jerne, N. K. "Towards a Network Theory of the Immune System." *Ann. Immunol. (Institute Pasteur)*, 1974: 373-389.

—. "Clonal Selection in a Lymphocyte Network." *Cellular Selection and Regulation in the Immune Response*, 1974: 39.

Jinsheng, S., Y. Jinchuan, e M. et al Ru. "Case-Based Real-Time Controller and its Application in Combustion Control of Hot Blast Stoves." *6th World Congress on Intelligent Control and Automation (WCICA-06)*. 2006. 7792-7796.

Jinsheng, S., Y. Jinchuan, H. Guoqiang, e Y. Jingtao. "Estimation of Blast Duration and Coordinated Control Strategy for Hot-Blast-Stove System." *1st IEEE Conference on Industrial Electronics and Applications*. 2006. 1-5.

Kim, J., e P. J. Bentley. "Evaluating Negative Selection in an Artificial Immune System for Network Intrusion Detection." *Genetic and Evolutionary Computation Conference (GECCO-01)*. San Francisco, 2001. 1330 - 1337.

Klein, J. *Immunology*. Blackwell Scientific Publications, 1990.

Kolodner, J. L. *Case-Based Reasoning*. 1a. Edição. San Mateo: Morgan Kaufmann, 1993.

Kolodner, J. L., M. T. Cox, e P. A. González-Calero. "Case-based reasoning-inspired approaches to Education." *The Knowledge Engineering Review*, 2005: 299-303.

Liao, T. W., Z. Zhang, e C. R. Mount. "Similarity measures for retrieval in case-based reasoning systems." *Applied Artificial Intelligence*, 1998: 267-288.

Lópes De Mántaras, R., P. Cunningham, e P. Perner. "Emergent case-based reasoning applications." *The Knowledge Engineering Review*, 2005: 325-328.

López de Mántaras, R., et al. "Retrieval, reuse, revision, and retention in case-based reasoning." *Knowledge Engineering Review*, 2005: 215–240.

Marchiori, E. "Hit Miss Networks with Applications to Instance Selection." *Journal of Machine Learning Research*, 2008: 997-1017.

Marling, C., E. Rissland, e A. Aamodt. "Integrations with case-based reasoning." *The Knowledge Engineering Review*, 2005: 241-245.

Matzinger, P. "Tolerance, danger, and the extended family." *Annu. Rev. Immunol.*, 1994: 991-1045.

Matzinger, P. "The danger model: a renewed sense of self." *Science* 296 (2002): 301-305.

McCarthy, J. *John McCarthy's Home Page*. 1995. <http://www-formal.stanford.edu/jmc/> (acesso em Outubro de 2008).

Menegazzo, Cinara T. "Raciocínio Baseado em Casos Aplicado a Diversos Domínios de Problema." Dissertação de Mestrado em Ciência da Computação, PPGCC, UFRGS, Porto Alegre, 2000.

Nassif, L. N., J. M. Nogueira, F. V. Andrade, A. Ahmed, A. Karmouch, e R. Impey. "Job completion prediction in grid using distributed case-based reasoning." *14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise*, 2005: 249–254.

Norvig, Peter, e Stuart Russel. *Artificial Intelligence: A Modern Approach*. 2a. Edição. Upper Saddle River, NJ: Prentice Hall, 2003.

Osborne, H., e D. Bridge. "Models of Similarity for Case-Based Reasoning." *Proceedings of the Interdisciplinary Workshop on Similarity and Categorisation*, 1997: 173-179.

Pal, Sankar K., e Simon C. K. Shiu. *Foundations of Soft Case-Based Reasoning*. 1a. Edição. John Wiley & Sons, 2004.

Patterson, D., N. Rooney, e M. Galusshka. "Efficient Similarity Determination and Case Construction Techniques For Case-Based Reasoning." *4th European Conference on Case-Based Reasoning (ECCBR-02)*. Aberdeen: Springer, 2002. 292–305.

Patterson, D., N. Rooney, M. Galushka, e S. Anand. "Towards Dynamic Maintenance of Retrieval Knowledge in CBR." *15th International FLAIRS Conference*. Florida: AAAI Press, 2002.

Perelson, A. S. "Immune network theory." *Immunology Review*, 1989: 5-36.

Perner, P., A. Holt, e M. Richter. "Image processing in case-based reasoning." *The Knowledge Engineering Review*, 2005: 311-314.

Reinartz, T., I. Iglezakis, e T. Roth-Berghofer. "Review and restore for case-based maintenance." *Computational Intelligence*. 2001. 214-234.

Rensberger, B. "In Self-Defense." *Life Itself*, 1996: 212-228.

Richter, Michael M., e Agnar Aamodt. "Soft Case-Based Reasoning." *Knowledge Engineering Review*, 2005: 203-207.

Rissland, E. L., K. D. Ashley, e L. K. Branting. "Case-based reasoning and law." *The Knowledge Engineering Review*, 2005: 293-298.

Schank, R. *Dynamic Memory: A Theory of Reminding and Learning in Computers and People*. 1a. Edição. New York: Cambridge University Press, 1982.

Schank, R., e R. Abelson. *Scripts, Plans, Goals and Understanding*. Hillsdale, New Jersey: Lawrence Erlbaum Associates, 1977.

Secker, A., A. A. Freitas, e J. Timmis. "AISEC: an Artificial Immune System for E-mail Classification." *Congress on Evolutionary Computation (CEC-2003)*. Canberra: IEEE Press, 2003. 131-138.

—. "AISIID: An artificial immune system for interesting information discovery on the web." *Applied Soft Computing*, 2008: 885–905.

Silvino, J. L., C. P. Bottura, e P. Resende. "Observadores de Fluxo e Estimaco de Parâmetros da Mquina de Induo Via Modelamento Discreto No-Estacionrio." *Controle & Automao* 4 (6) (1993): 11-22.

Simoudis, E. "Using Case-Based Retrieval for Customer Technical Support." *IEEE Expert*, 1992: 7-13.

Smyth, B. "Case-base maintenance." *11th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE-98)*. Berlin: Springer-Verlag, 1998. 507-516.

Smyth, B., e E. McKenna. "Footprint-based retrieval." *3th International Conference on Case-Based Reasoning (ICCBR-99)*. Berlin: Springer-Verlag, 1999. 343–357.

—. "Modelling the competence of case-bases." *4th European Workshop on Case-Based Reasoning (EWCBR-98)*. Berlin: Springer, 1998. 208–220.

Smyth, B., e M. T. Keane. "Adaptation-Guided Retrieval: Questioning the Similarity Assumption in Reasoning." *Artificial Intelligence*, 1998: 249-293.

—. "Remembering to forget: A competence-preserving case deletion policy for case-based reasoning systems." *14th International Joint Conference on Artificial Intelligence*. 1995. 377-382.

Stahl, A. "Learning similarity measures: A formal view based on a generalized CBR model." *6th International Conference on Case-Based Reasoning (ICCBR-05)*. Springer, 2005. 507–521.

Stahl, A., e T. Gabel. "Optimizing Similarity Assessment in Case-Based Reasoning." *21th National Conference on Artificial Intelligence (AAAI-06)*. Boston: AAAI Press, 2006.

Timmis, J. "Artificial Immune Systems: A Novel Data Analysis Technique Inspired by the Immune Network Theory." Tese de Doutorado, Department of Computer Science, University of Whales, 2000.

Timmis, J., e M. Neal. "A Resource Limited Artificial Immune System." *Knowledge Based Systems*, 2001: 121–130.

Timmis, J., M. Neal, e J. Hunt. "An Artificial Immune System for Data Analysis." *Biosystems*, 2000: 143–150.

Tizard, I. R. *Immunology An Introduction*. 4a. Edição. Saunders College Publishing, 1995.

U.S. National Institutes of Health. "Publication Order Form." Vers. NIH Publication No. 03-5423. *National Institute of Allergy and Infectious Diseases*. 2003. <http://www3.niaid.nih.gov/PubsOrderForm/default.asp> (acesso em Outubro de 2008).

Varela, F., e A. Coutinho. "Second generation immune networks." *Immunology Today*, 1991: 159-166.

Venkatasubramanian, V., R. Rengaswamy, K. Yin, e S. N. Kavuri. "A review of process fault detection and diagnosis Part I: Quantitative model-based methods." *Computers and Chemical Engineering*, 2003: 293-311.

Wang, X., X. Z. Gao, e S. J. Ovaska. "Artificial immune optimization methods and applications – A survey." *IEEE International Conference on Systems, Man, and Cybernetics*. The Hague, 2004. 3415-3420.

Watkins, A., J. Timmis, e L. Boggess. "Artificial immune recognition system (AIRS): An immune inspired supervised machine learning algorithm." *Genetic Programming and Evolvable Machines*, 2004: 291–318.

Watson, Ian, e Farhi Marir. "Case-Based Reasoning: A Review." *The Knowledge Engineering Review*, 1994: 355-381.

Wess, Stefan, Klaus-Dieter Althoff, e Guido Derwand. "Using k-d Trees to Improve the Retrieval Step in Case-Based Reasoning." *Topics in Case-Based Reasoning*, 1994: 167-181.

Whitbrook, A. M., U. Aickelin, e J. M. Garibaldi. "Idiotypic Immune Networks in Mobile Robot Control." *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, 2007: 1581–1598.

Wikipedia. *Motor elétrico*. 2005.
http://pt.wikipedia.org/wiki/Motor_el%C3%A9trico (acesso em Outubro de 2008).

—. *Somatic hypermutation*. 2005.
http://en.wikipedia.org/wiki/Somatic_hypermutation (acesso em Outubro de 2008).

—. *V(D)J recombination*. 2006.
[http://en.wikipedia.org/wiki/V\(D\)J_recombination](http://en.wikipedia.org/wiki/V(D)J_recombination) (acesso em Outubro de 2008).

Wilson, D. R., e T. R. Martinez. "Instance Pruning Techniques." *14th International Conference of Machine Learning*, 97: 404-411.

—. "Reduction techniques for instance-based learning algorithms." *Machine Learning*, 2000: 257-286.

Wilson, D., e D. Leake. "Maintaining cased-based reasoners: Dimensions and directions." *Computational Intelligence*, 2001: 196-213.

Xu, R., e D. Wunsch. "Survey of clustering algorithms." *IEEE Transactions on Neural Networks*, May de 2005: 645–678.

Yager, R. "A model of participatory learning." *IEEE Transactions on Systems, Man and Cybernetics*, 1990: 1229–1234.

Yoo, J., e P. Hajela. "Immune network simulations in multicriterion design." *Structural Optimization*, 1999: 85-94.

Yue, X., A. Abraham, Z. Chi, Y. Hao, e H. Mo. "Artificial immune system inspired behavior-based anti-spam filter." *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 2007: 729-740.

Zadeh, L. A. "Fuzzy logic, neural networks and soft computing." *Communications of the ACM*, 1994: 77–84.

Zhou, J., e D. Dasgupta. "Revisiting negative selection algorithms." *Evolutionary Computation*, 2007: 223–251.

Apêndice A - Máquina de corrente contínua

O conteúdo deste apêndice foi retirado assim como publicado em notas de aula de uma máquina de corrente contínua, cujo conteúdo foi originalmente produzido pelo mesmo autor na sua tese de doutorado (Caminhas 1997).

A.1. Modelo do Sistema de Acionamento de Corrente Contínua

O sistema de acionamento de CC é composto por duas fontes de alimentação, conversores estáticos controlados, uma máquina de corrente contínua e uma carga mecânica, Figura A.1. O diagrama de blocos do sistema completo, incluindo o controle, é mostrado na Figura A.2. O controlador de velocidade, tipo proporcional integral, fornece o valor da corrente de armadura de referência. O controle da corrente da armadura é feito através da variação da tensão de alimentação. Esta tensão é função do ângulo de disparo do conversor, fornecido pelo controlador de corrente. A malha de campo, além de proporcionar o controle da corrente, permite que o sistema de acionamento opere à velocidade acima da nominal com potência constante, via enfraquecimento de campo.

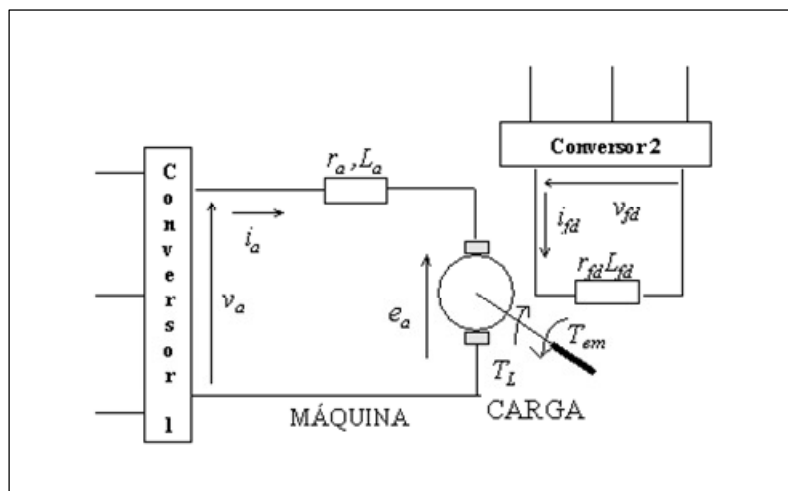


Figura A.1 - Representação do sistema de acionamento de CC.

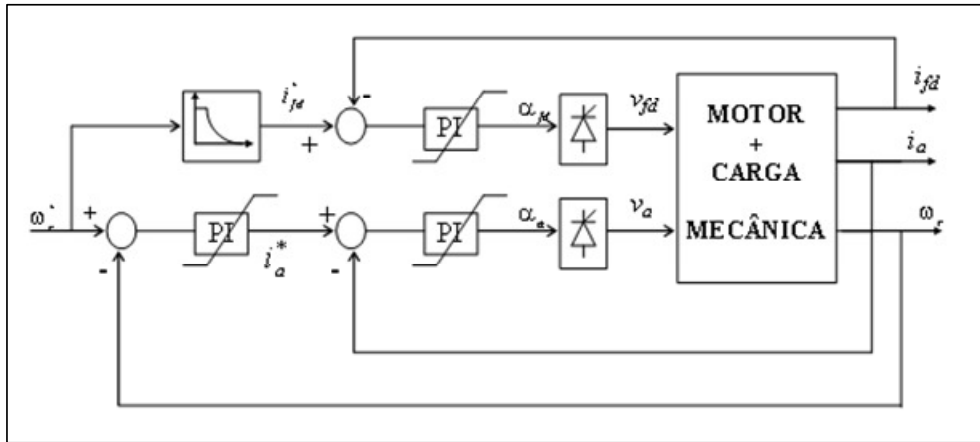


Figura A.2 - Diagrama de blocos do sistema de acionamento CC.

Onde

v_a : tensão de alimentação do circuito da armadura;

v_{fd} : tensão de alimentação do circuito de campo;

i_a : corrente do circuito de armadura;

i_{fd} : corrente do circuito de campo;

ω_r : velocidade mecânica de rotação em rad/s;

α_{fd} : ângulo de disparo do conversor do circuito de campo;

α_a : ângulo de disparo do conversor do circuito de armadura.

sendo que as variáveis indexadas por n são referentes a valores nominais da máquina.

A corrente de campo de referência é calculada da seguinte forma:

$$i_{fd}^* = \begin{cases} i_{fd}^n, & \text{para } \omega_r^* \leq \omega_r^n \\ i_{fd}^n \cdot \frac{\omega_r^n}{\omega_r^*}, & \text{para } \omega_r^* > \omega_r^n \end{cases} \quad (\text{A.1})$$

A.1.1. Modelo de Máquina de Corrente Contínua

A máquina de corrente contínua possui o campo (excitação) independente da armadura, como mostrado na Figura A.1. Considerando as correntes i_a e i_{fd} e a velocidade ω_r como variáveis de estado, o modelo dinâmico para a máquina funcionando como motor é representado por:

$$\begin{cases} \frac{di_a}{dt} = \frac{1}{L_a}(v_a - r_a \cdot i_a - e_a) \\ \frac{di_{fd}}{dt} = \frac{1}{L_{fd}}(v_{fd} - r_{fd} \cdot i_{fd}) \\ \frac{d\omega_r}{dt} = \frac{1}{J_m}(T_{em} - B_m \cdot \omega_r - T_L) \end{cases} \quad (\text{A.2})$$

Sendo

$$\begin{cases} e_a = L_{afd} \cdot i_{fd} \cdot \omega_r \\ T_{em} = L_{afd} \cdot i_{fd} \cdot i_a \end{cases} \quad (\text{A.3})$$

Onde

r_a : resistência do circuito da armadura;

r_{fd} : resistência do circuito de campo;

L_a indutância do circuito da armadura;

L_{fd} : indutância do circuito de campo;

L_{afd} : indutância mútua armadura/campo;

e_a força-contraeletromotriz da armadura;

T_{em} : conjugado eletromagnético;

T_L conjugado mecânico exigido pela carga;

B_m : coeficiente de atrito viscoso;

J_m momento de inércia do sistema motor/carga.

Considerando que i_a , i_{fd} e ω_r são as variáveis medidas, colocando na forma de equação de estado, tem-se o modelo contínuo para a máquina de corrente contínua, representado por:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -\frac{r_a}{L_a} & -\frac{L_{afd}}{L_a} \cdot x_3 & 0 \\ 0 & \frac{r_{fd}}{L_{fd}} & 0 \\ \frac{L_{afd}}{J_m} \cdot x_2 & 0 & -\frac{B_m}{J_m} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & \frac{1}{L_{fd}} \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_a \\ v_{fd} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \cdot [T_L] \quad (\text{A.4})$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad \text{onde} \quad \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} i_a \\ i_{fd} \\ \omega_r \end{bmatrix}$$

Nota-se que se trata de um sistema dinâmico não linear, resultante do produto de variáveis de estado.

Um modelo discreto para o sistema acima é representado pela Equação (A.5). Para obtê-lo foi usada a mesma metodologia adotada por Silvino *et alii* (1993).

$$\begin{aligned} \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} &= \begin{bmatrix} a_1 & a_2(k) & 0 \\ 0 & a_3 & 0 \\ a_4(k) & 0 & a_5 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} b_1 & 0 \\ 0 & b_2 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_a(k) \\ v_{fd}(k) \\ 0 \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ 0 \\ d_1 \end{bmatrix} \cdot [T_L(k)] \end{aligned} \quad (A.5)$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}$$

Onde

$$\left\{ \begin{aligned} a_1 &= a_1(r_a, L_a) = e^{-\frac{r_a h}{L_a}} & a_3 &= a_3(r_{fd}, L_{fd}) = e^{-\frac{r_{fd} h}{L_{fd}}} \\ a_5 &= a_5(B_m, J_m) = e^{-\frac{B_m h}{J_m}} & a_4 &= a_4(B_m, J_m, x_2(k)) = \frac{(1 - a_5)}{B_m} \cdot x_2(k) \\ b_1 &= b_1(r_a, L_a) = \frac{1 - a_1}{r_a} \\ a_2(k) &= a_2(r_a, L_a, r_{fd}, L_{fd}, x_3(k)) = \\ &= \frac{1}{r_{fd} \cdot L_a - r_a \cdot L_{fd}} [L_{afd} \cdot L_{fd} \cdot (a_3 - a_1) \cdot x_3(k) + (r_a \cdot L_{fd} \cdot a_1 - r_{fd} \cdot L_a \cdot a_3)] \\ b_2 &= b_2(r_{fd}, L_{fd}) = \frac{1 - a_3}{r_{fd}} d_1 = d_1(B_m, J_m) = -\frac{1 - a_5}{B_m} \end{aligned} \right. \quad (A.6)$$

A.1.2. Modelo de Carga Dinâmica

O modelo da carga mecânica é estabelecido pela equação de conjugado. Considerou-se uma carga genérica, cujo conjugado é:

$$T_L = C_0 + C_1 \cdot \omega_r + C_2 \cdot \omega_r^2 \quad (A.7)$$

A.1.3. Modelo de Conversores Estáticos

Os dois conversores, da armadura e do circuito de campo, são tipo Ponte de Graetz a Tiristor. É adotado o modelo estático para calcular a tensão de saída em função do ângulo de disparo (Bose, 1996):

$$v_a = 2.34V_{oa} \cos \alpha_a \quad (A.8)$$

$$v_{fd} = 2.34V_{ofd} \cos \alpha_{fd}$$

sendo V_{oa} e V_{ofd} os valores eficazes das tensões nas entradas dos conversores da armadura e campo, respectivamente.

A.2. Modelagem das Falhas para o Sistema de Acionamento de Corrente Contínua

Do ponto de vista de classificação de falhas, o sistema de acionamento C.C. pode ser dividido em três grandes grupos: atuadores (conversores da armadura e do campo); planta ou processo (máquina e carga mecânica) e sensores (medidores de corrente e velocidade). Serão abordadas falhas relacionadas a estes três grupos.

A.2.1. Falhas em Atuadores

O número de falhas que podem ocorrer nos conversores estáticos é muito grande, como, por exemplo, queima de componentes nos circuitos de potência e comando. Mas, do ponto de vista da máquina, as duas falhas mais graves são curto-circuito e desconexão do conversor. Estas falhas podem ser modeladas pelas chaves ideais CH_{aa} , CH_{afd} , CH_{cca} e CH_{ccfd} , Figura A.3.

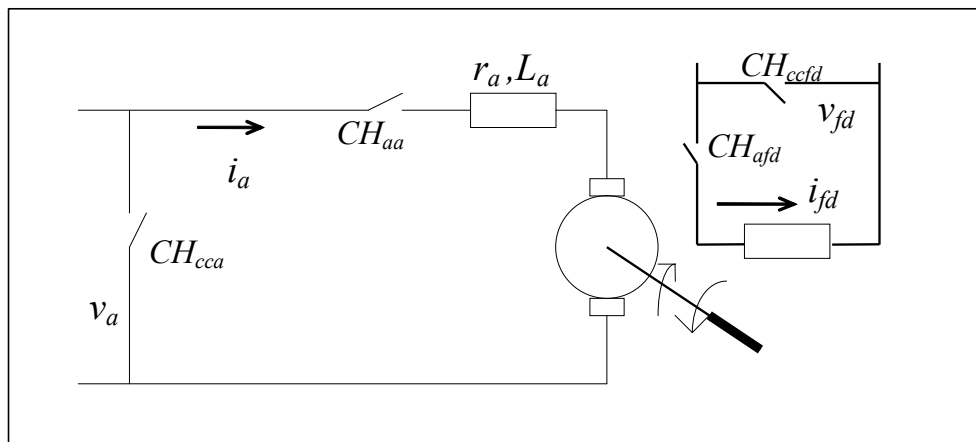


Figura A.3 - Representação do sistema de acionamento C.C. para falhas nos atuadores.

Associando os parâmetros:

$k_{aa} \in \{0,1\}$ $\{1\}$ indica a desconexão do conversor da armadura;

$k_{afd} \in \{0,1\}$ $\{1\}$ indica a desconexão do conversor do campo;

$k_{cca} \in \{0,1\}$ $\{1\}$ indica curto-circuito do conversor da armadura;

$k_{ccfd} \in \{0,1\}$ $\{1\}$ indica curto-circuito do conversor do campo.

aos estados das chaves, tal que:

$k_i = 0$ se CH_i estiver aberta e

$k_i = 1$ se CH_i estiver fechada, $i = aa, afd, cca$ e $ccfd$.

Considerando que uma falta de alimentação é caracterizada por uma corrente e um curto-circuito, por uma tensão nula, tem-se o modelo para falhas no circuito de alimentação da máquina dado por (A.9).

$$\begin{aligned} \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} &= \begin{bmatrix} k_{aa} \cdot a_1 & k_{aa} \cdot a_2(k) & 0 \\ 0 & k_{afd} \cdot a_3 & 0 \\ a_4(k) & 0 & a_5 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \\ &+ \begin{bmatrix} b_1 & 0 \\ 0 & b_2 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} k_{aa} \cdot \bar{k}_{cca} \cdot v_a(k) \\ k_{afd} \cdot \bar{k}_{ccfd} \cdot v_{fd}(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ d_1 \end{bmatrix} \cdot [T_L(K)] \end{aligned} \quad (A.9)$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \text{ onde } \bar{k}_i = 1 - k_i$$

A.2.2. Falhas em Componentes da Planta

As falhas no conjunto máquina/carga consideradas são: curto-circuito entre espiras na bobina da armadura; curto-circuito entre espiras na bobina do campo; falha no sistema de ventilação e falha de lubrificação. As duas primeiras podem ser modeladas por variações nas resistências e indutâncias da armadura e do campo. Estas variações são modeladas pelos parâmetros:

$k_{ca}^{ra} \in [0,1]$ indica a variação na resistência r_a quando ocorre curto-circuito na bobina da armadura;

$k_{ca}^{La} \in [0,1]$ indica a variação na indutância L_a quando ocorre curto-circuito na bobina da armadura;

$k_{cfd}^{rfd} \in [0,1]$ indica a variação na resistência r_{fd} quando ocorre curto-circuito na bobina do campo;

$k_{cfd}^{Lfd} \in [0,1]$ indica a variação na indutância L_{fd} quando ocorre curto-circuito na bobina do campo.

A falha no sistema de ventilação se traduz em variações das resistências da armadura e do campo, provocadas pela elevação de temperatura da máquina. Já uma má lubrificação dos mancais, falha de lubrificação, pode ser modelada por uma variação do coeficiente de atrito.

Os parâmetros que representam estas duas falhas são:

k_{fv}^{ra} indica a variação na resistência r_a quando ocorre falha no sistema de ventilação;

k_{fv}^{rfd} indica a variação na resistência r_{fd} quando ocorre falha no sistema de ventilação;

k_{fl} indica a variação no coeficiente de atrito, B_m , quando ocorre falha de lubrificação dos mancais.

O modelo que incorpora somente as falhas em componentes da planta é composto pelas Equações de (A.5) a (A.8), onde, no cálculo dos coeficientes de (A.6), a_1 , $a_2(k)$, a_3 , $a_4(k)$, b_1 , b_2 e d_1 , os parâmetros nominais são substituídos por (A.10).

$$\begin{cases} r_a^f = k_{ca}^{ra} \cdot k_{fv}^{ra} \cdot r_a \\ L_a^f = k_{ca}^{La} \cdot k_{fv}^{La} \cdot L_a \\ r_{fd}^f = k_{cfd}^{rfd} \cdot k_{fv}^{rfd} \cdot r_{fd} \\ L_{fd}^f = k_{cfd}^{Lfd} \cdot k_{fv}^{Lfd} \cdot L_{fd} \\ B_m^f = k_{fl} \cdot B_m \end{cases} \quad (\text{A.10})$$

A.2.3. Falhas de Instrumentação

Aqui são consideradas falhas bruscas (desligamentos) dos sensores de corrente (armadura e campo) e de velocidade, modeladas pelos parâmetros:

$k_{ia}^f \in \{0,1\}$ {1} indica falha no sensor da corrente da armadura;

$k_{ifd}^f \in \{0,1\}$ {1} indica falha no sensor da corrente do campo;

$k_{\omega_r}^f \in \{0,1\}$ {1} indica falha no sensor de velocidade da máquina.

Com relação ao modelo das falhas nos sensores, este difere do modelo representado pelas Equações (A.5) a (A.8) somente com relação à equação de saída, que passa a ser calculada por:

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \end{bmatrix} = \begin{bmatrix} \bar{k}_{ia}^f & 0 & 0 \\ 0 & \bar{k}_{ifd}^f & 0 \\ 0 & 0 & \bar{k}_{\omega_r}^f \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \quad \text{onde } \bar{k}_i = 1 - k_i \quad (\text{A.11})$$

A.3. Modelo Geral das Falhas no Sistema de Acionamento de Corrente Contínua

Considerando os dois tipos de falhas abordados anteriormente, Tabela A.1, tem-se o modelo completo dado pelas Equações (A.12), (A.6), (A.7) e (A.8). O circuito representativo deste modelo é mostrado na Figura A.3.

$$\begin{aligned} \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} &= \begin{bmatrix} k_{aa} \cdot a_1^f & k_{aa} \cdot a_2^f(k) & 0 \\ 0 & k_{af} \cdot a_3^f & 0 \\ a_4^f(k) & 0 & a_5^f \end{bmatrix} \cdot \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \\ &+ \begin{bmatrix} b_1^f & 0 \\ 0 & b_2^f \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} k_{aa} \cdot \bar{k}_{cca} \cdot v_a(k) \\ k_{af} \cdot \bar{k}_{ccfd} \cdot v_{fd}(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ d_1^f \end{bmatrix} \cdot [T_L(k)] \end{aligned} \quad (A.12)$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \end{bmatrix} = \begin{bmatrix} \bar{k}_{i_a}^f & 0 & 0 \\ 0 & \bar{k}_{i_{fd}}^f & 0 \\ 0 & 0 & \bar{k}_{\omega_r}^f \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}, \text{ onde } \bar{k}_i = 1 - k_i$$

sendo que os coeficientes indexados pelo sobrescrito “f” são calculados substituindo os parâmetros nominais da máquina por (2.10).

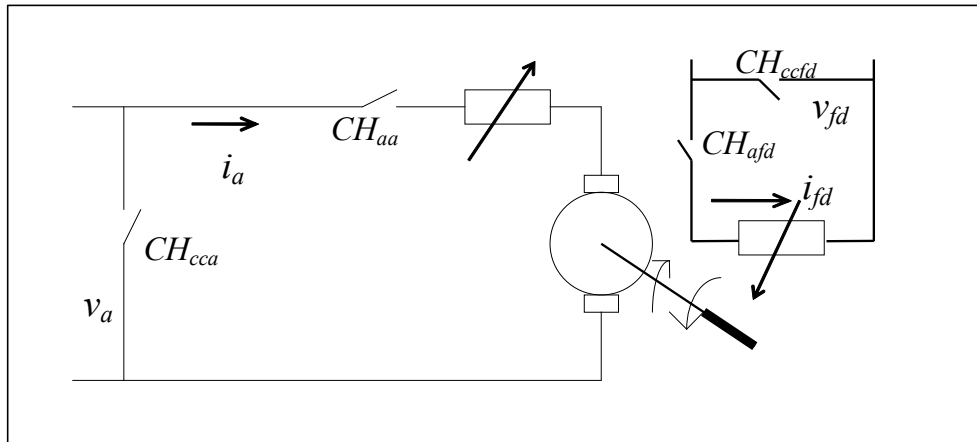


Figura A.4 - Representação do sistema de acionamento C.C. para os diversos tipos de falhas.

Tipo de Falha	Variáveis Afetadas	Parâmetros Indicadores da Falha	Definição dos Parâmetros
Abertura da bobina da armadura	$i_a = 0$	k_{aa}	$\{0,1\}$
Abertura da bobina de campo	$i_{fd} = 0$	k_{af}	$\{0,1\}$
Curto-circuito do conversor da armadura	$v_a = 0$	k_{cFa}	$\{0,1\}$
Curto-circuito do conversor do	$v_{fd} = 0$	k_{cFd}	$\{0,1\}$

Tipo de Falha	Variáveis Afetadas	Parâmetros Indicadores da Falha	Definição dos Parâmetros
campo			
Curto-circuito entre espiras da bobina de armadura	r_a e L_a	$k_{ca}^{r_a}$ e $k_{ca}^{L_a}$	[0,1]
Curto-circuito entre espiras da bobina de campo	r_{fd} e L_{fd}	$k_{cf}^{r_{fd}}$ e $k_{cf}^{L_{fd}}$	[0,1]
Falha no sistema de ventilação	r_a e r_{fd}	$k_{fv}^{r_a}$ e $k_{fv}^{r_{fd}}$	[1, ∞)
Falha de lubrificação dos mancais	B_m	k_{fl}	[0, ∞)
Falha no sensor de corrente da armadura	i_a	$k_{i_a}^f$	
Falha no sensor de corrente da fonte	i_{fd}	$k_{i_{fd}}^f$	{0,1}
Falha no sensor de velocidade	ω_r	$k_{\omega_r}^f$	{0,1}

Tabela A.1 - Resumo das falhas no sistema de acionamento CC.

Apêndice B – Agrupamento Participativo

A seguir, serão expostos o aprendizado e o agrupamento participativo (Yager 1990) e como este trabalha em conjunto com o SIA.

O termo participativo remonta ao aprendizado participativo, que é

um mecanismo de aprendizagem no qual a percepção acerca de um sistema depende de quanto o mecanismo é capaz de representar o conhecimento já estabelecido sobre o sistema. Em linhas gerais, o conhecimento atual é parte de um processo de aprendizagem que depende da coerência entre o estado atual do sistema e a observação mais recente. A renovação do conhecimento, isto é, o aprendizado, é caracterizado pela interpretação da nova informação no contexto atual (de Lima 2008).

Com isto, torna-se necessário que exista uma conexão entre os novos pontos observados e o estado atual, atualizando o sistema, sempre que necessário. Como o algoritmo de agrupamento participativo é um desdobramento do aprendizado participativo, este será inicialmente formulado baseado em propostas que já que trabalham fortemente o algoritmo de agrupamento participativo (da Silva 2003) (de Lima 2008).

B.1. Aprendizado Participativo

O aprendizado participativo utiliza a noção de que, quanto maior a compatibilidade entre uma nova observação e o estado atual do sistema, maior será o impacto dessa informação na aprendizagem do mesmo. Nesse sentido, se $\bar{v} \in [0,1]^p$ é a representação do conhecimento, pode-se dizer que o objetivo é aprender o valor de \bar{v} a partir de um conjunto de observações $x^k \in [0,1]^p$. Essa aprendizagem é feita via atualização de estimativas v^k feitas de acordo com sua compatibilidade em relação a cada x^k observado.

Em outras palavras, busca-se definir uma seqüência $v^k = v(G^k, x^{k-1}, v^{k-1})$ que, sob certas condições, possua um comportamento convergente e consistente com o sistema objeto de aprendizagem. Para tal, devem ser consideradas as informações, a experiência (ou percepção) acumulada, e uma função de compatibilidade que sirva para medir o quanto o conhecimento está de acordo com as observações. A função $\rho: [0,1]^p \times$

$[0,1]^p \rightarrow [0,1]$ de compatibilidade, de forma geral, pode ser definida como uma função tal que seja:

1. Reflexiva: $\rho(x, x) = 1$;
2. Simétrica: $\rho(x, v) = \rho(v, x)$.

No contexto deste trabalho, $\rho(x, v) = 1$ define compatibilidade plena entre x e v , e $\rho(x, v) = 0$ define incompatibilidade entre x e v .

Com isso, a função de compatibilidade ou, simplesmente, compatibilidade, representa o quanto o dado observado está de acordo com o modelo atual, e uma construção para a seqüência $\{v^k\}$ deve ser tal que, para valores altos de ρ , o sistema esteja mais propenso a aprender, ocorrendo o oposto para baixos valores. Um mecanismo que implementa essa idéia está descrito a seguir.

Considere x^k uma variável real em $[0,1]^p$ e $\alpha \in [0,1]$ um parâmetro chamado de taxa de aprendizagem. Então a dinâmica associada ao aprendizado participativo é definida por uma seqüência $\{v^k\}$ na forma:

$$v^{k+1} = v^k + \alpha G^k(x^k - v^k) \quad (\text{B.1})$$

no qual $v^1 \in [0,1]^p$ e $G^k = G(\rho(x^k, v^k))$ é uma função que depende da compatibilidade entre os vetores x^k e v^k em $k = 1, 2, \dots$

Nesse processo, o conceito participativo é caracterizado pela contribuição de cada observação de acordo com um critério de aceitação, obtido a fim de determinar a coerência entre o conhecimento atual expresso por v^k e um novo dado x^k no instante k . Em outras palavras, busca-se definir o quanto a última observação pode contribuir na atualização do conhecimento representado pela variável v^k .

Espera-se que a compatibilidade seja tão maior quanto mais próximo x^k for de v^k . Dentre as diversas escolhas possíveis, uma primeira seria escolher $G^k = \rho^k$, gerando a seguinte expressão:

$$v^{k+1} = v^k + \alpha \rho^k(x^k - v^k) \quad (\text{B.2})$$

sendo $\alpha \in [0,1]$ a taxa de aprendizagem e $\rho \in [0,1]$ uma função de compatibilidade.

Há uma infinidade de maneiras para satisfação da definição de uma função de compatibilidade. Em particular, utilizar a distância como uma medida da proximidade entre x^k e v^k é uma escolha intuitiva para decidir se as mesmas são ou não compatíveis. Uma escolha simples e direta é:

$$\rho = 1 - d(x^k - v^k) \quad (\text{B.3})$$

na qual $d(x^k, v^k)$, simplesmente d^k , é a distância entre x^k e v^k . Vale notar que essa abordagem permite o cálculo de diferentes medidas de compatibilidade para cada componente de x^k .

No entanto, tal abordagem não leva em conta situações onde x^k represente uma informação conflitante com v^k , isto é, casos onde observações com baixo índice de compatibilidade são recebidas durante instantes consecutivos. Com isso, é importante que o mecanismo possa detectar também inconsistências no aprendizado, minimizando a influência do índice de compatibilidade na atualização de v^k em casos onde o índice de alerta é alto. Esse índice de alerta é definido de forma que, quanto maior for o seu valor, menor deve ser a influência do índice de compatibilidade na atualização do sistema.

De fato, um baixo valor para o índice de compatibilidade em um determinado instante pode ser visto como uma aversão ao aprendizado nesse instante. Por outro lado, baixos valores de compatibilidade durante vários instantes consecutivos podem representar uma necessidade de revisão no aprendizado e não, necessariamente, uma aversão. Nesse sentido, o índice de alerta deve detectar se uma observação pouco compatível será subestimada, ou se a baixa compatibilidade caracteriza a necessidade de revisão do sistema.

Dentre outras possibilidades, considere o índice de alerta dado por:

$$a^{k+1} = a^k + \beta(1 - \rho^k - a^k) \quad (\text{B.4})$$

Sendo $\beta \in [0,1]$ um parâmetro e $a^k \in [0,1]$ com $k = 1,2, \dots$

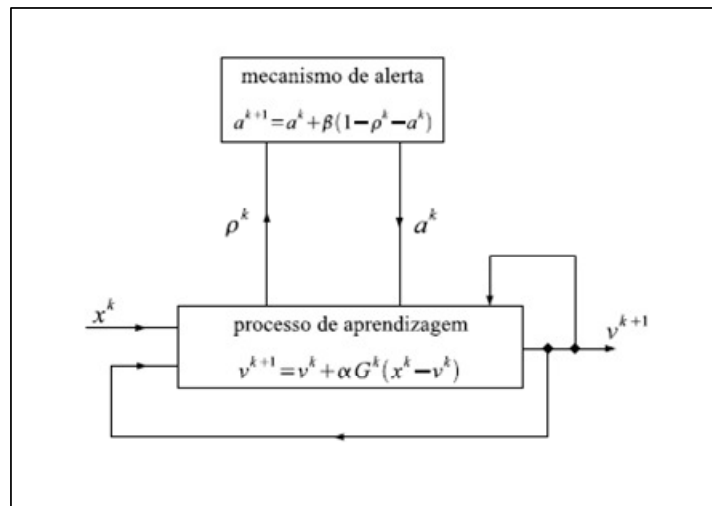


Figura B.1 – Mecanismo do aprendizado participativo

Como mostrado na Figura B.1, para cada entrada x^k o índice de compatibilidade ρ^k é obtido e enviado ao mecanismo de alerta, que monitora o desempenho do aprendizado. O índice de alerta a^k é calculado e o aprendizado é atualizado na direção do vetor $(x^k - v^k)$, onde o tamanho do passo depende do termo αG^k . Quanto maior for o índice de alerta, menor será a influência do índice de compatibilidade no cálculo de v^{k+1} . Ainda pela Figura

B.1, o mecanismo é considerado participativo, pois a atualização do sistema é feita de acordo com o conhecimento acumulado, ou seja, a variação no aprendizado depende do quanto já se conhece do sistema.

Observando a expressão B.4, percebe-se que o parâmetro β atua como um controlador da variação do alerta em relação a uma dada observação. Para valores de β próximos de 0, as variações do índice de alerta são menores, ao passo que para valores de β próximos de 1 estas variações são maiores. Em contrapartida, o índice de compatibilidade ρ atua na aceitação de uma observação. Juntos, estes dois índices trabalham na construção de agrupamentos coerentes com a distribuição dos dados. Desta maneira, a atualização v^{k+1} do aprendizado pode ser ampliada, levando em conta o índice de alerta. O princípio é que o alerta possa atuar na aprendizagem, alterando a influência da compatibilidade na expressão B.5. Dentre as várias maneiras de colocar estas possibilidades, Yager (1990) sugere $G^k = (\rho^k)^{1-a^k}$ coloca:

$$v^{k+1} = v^k + \alpha(\rho^k)^{1-a^k}(x^k - v^k) \quad (\text{B.5})$$

Assim, o índice de alerta regula o efeito do índice de compatibilidade no processo de aprendizagem.

B.2. Agrupamento Participativo

O algoritmo de agrupamento participativo, aqui utilizado, é um algoritmo de partição, e objetiva a divisão do conjunto de dados em subconjuntos e a associação de um ponto representativo a cada subconjunto, denominado centro do grupo ou protótipo (Bezdek e Pal 1992). No agrupamento participativo aqui adotado, os índices de alerta e compatibilidade controlam o número de grupos, que, em última instância, são identificados pelos centros dos grupos (da Silva 2003). O objetivo é encontrar um conjunto de grupos que seja representativo dos dados analisados. Encontrar este conjunto de grupos não é uma tarefa simples e depende fortemente do conjunto de dados, função de validação e parâmetros do algoritmo.

De maneira geral, um agrupamento pode ser do tipo supervisionado ou não supervisionado. O primeiro tipo se dá, quando o número de grupos é um parâmetro de entrada e, o segundo, quando o número de grupos é definido de acordo com os dados de entrada (Gath e Geva 1989).

As modalidades dos experimentos realizados se dividem no aprendizado supervisionado, que remete ao agrupamento supervisionado, e aprendizado não supervisionado, relativo ao agrupamento não supervisionado, onde a quantidade de grupos é inicialmente desconhecida. O algoritmo apresentado a seguir é um algoritmo de agrupamento não supervisionado, chamado agrupamento participativo. Para atender à primeira modalidade do experimento, este algoritmo será alterado para que possa funcionar como um algoritmo de agrupamento supervisionado, podendo assim aproveitar-se de um cenário onde existe conhecimento dos agrupamentos de casos. Em comum, o algoritmo original e o alterado têm o fato que seus centros não são encontrados

como resultado de um problema de otimização, mas sim de modificações gradativas baseando-se nos indicadores de alerta e compatibilidade.

No agrupamento participativo, cada grupo possui um centro que é atualizado de acordo com os índices do aprendizado participativo (de Lima 2008). Sendo c^k o número de grupos, o centro de cada um deles é definido como $v_i^k \in [0,1]^p$, onde $k = 1,2, \dots$ são os instantes nos quais os dados são fornecidos e $i = 1,2, \dots, c^k$, sendo os centros dos grupos atualizados baseando-se na expressão B.5.

Os índices de alerta e compatibilidade modelam o comportamento do algoritmo de agrupamento participativo. O índice de alerta é responsável por regular o efeito da compatibilidade e, também, por servir como critério de criação de novos grupos, pois sempre que os índices de alerta dos grupos existentes ultrapassarem um determinado limiar, um novo grupo é criado, Figura B.3, contribuindo, assim, para a ampliação do conhecimento do sistema (da Silva 2003). Em contrapartida, se os índices de alerta não ultrapassarem o limiar τ , então o ponto apresentado fará parte de um grupo existente, como na Figura B.2.

Um aspecto interessante é que o agrupamento participativo também analisa a redundância dos agrupamentos, pois, dependendo da seqüência de entrada, dados que na realidade pertencem ao mesmo agrupamento podem se formar em áreas diferentes do espaço, sendo inicialmente separados em grupos diferentes. A detecção da redundância de grupos atua justamente na união ou fusão de grupos que, na realidade, possuem um padrão de dados mais amplo em comum, como na Figura B.4. A detecção de grupos redundantes é calculada, utilizando o índice de compatibilidade. Dados dois centros diferentes, v_i^k e v_j^k , tem-se que a compatibilidade entre eles é:

$$\rho_{ij}^k = \rho_{ij}^k(v_i^k, v_j^k) = 1 - d(v_i^k, v_j^k) \quad (\text{B.6})$$

sendo que, quanto mais próximo de 1 for $\rho_{ij}^k \in [0,1]$, maior será o grau de redundância entre estes centros. O parâmetro $\lambda \in [0,1]$ é definido no algoritmo, de forma que dois grupos, i, j distintos são considerados redundantes, se o índice de compatibilidade entre os dois for maior ou igual a λ , ou seja, se $\rho_{ij}^k \geq \lambda$, o que implica $d(v_i^k, v_j^k) > (1 - \lambda)$.

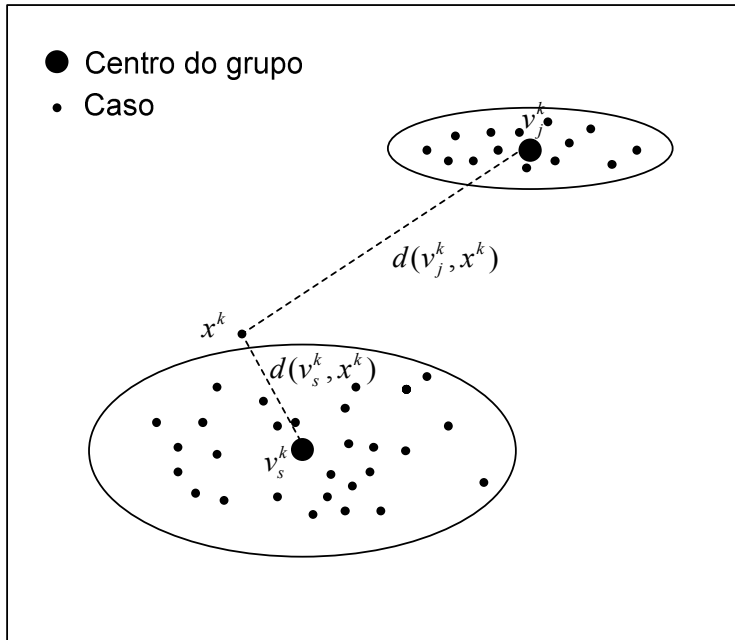


Figura B.2 – O ponto x^k fará parte do grupo i , pois apresenta maior compatibilidade com i do que com j .

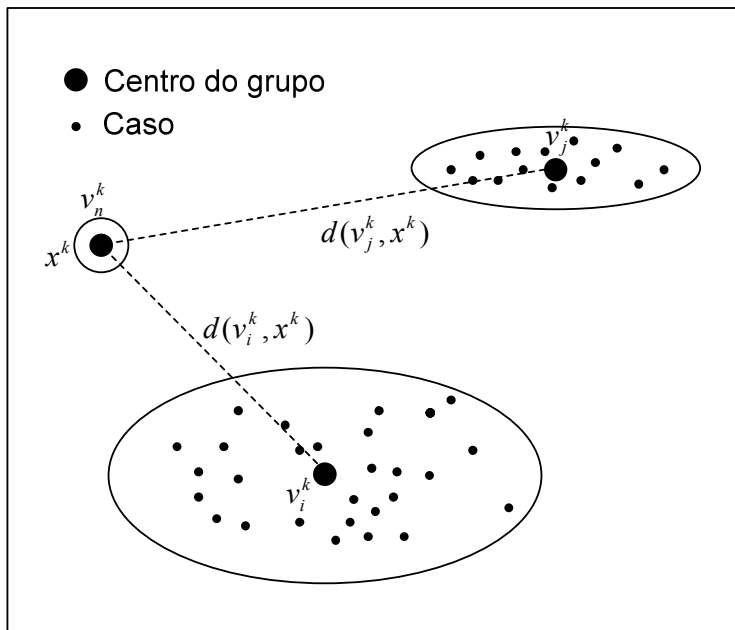


Figura B.3 – O ponto x^k origina a criação do grupo de centro v_n^k ($n = \text{novo}$), pois a_s^{k+1} e $a_j^{k+1} \geq \tau$.

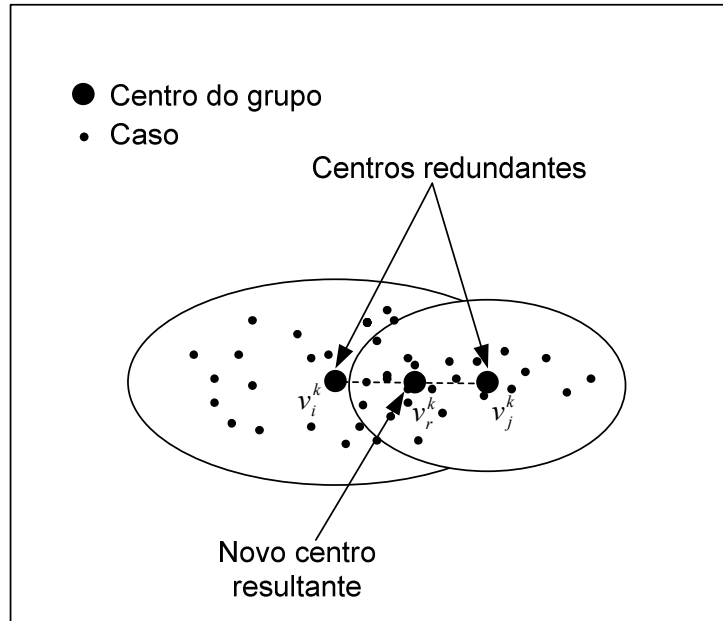


Figura B.4 – O ponto v_r^k ($r =$ resultante) é o centro do novo grupo originado da união dos grupos cujos centros são v_i^k e v_j^k , já que $\rho_{ij}^k = 1 - d(v_i^k, v_j^k) \geq \lambda$.

Colocados os princípios de funcionamento do algoritmo de agrupamento participativo, este é o momento de introduzi-lo. Este algoritmo é a versão apresentada por Lima (2008), onde não há critério de parada especificado.

Algoritmo de agrupamento participativo

1. Inicializar o número de grupos $c^1 \geq 2$ e os centros $v_i \in [0,1]^p, i = 1, 2, \dots, c^1$.
2. Escolher os parâmetros $\alpha \in [0,1], \beta \in [0,1], \tau \in [0,1]$ e $\lambda \in [0,1]$.
3. **Para** $x^k \in [0,1]^p, k = 1, 2, \dots$
 - 3.1. **Para** $i = 1, 2, \dots, c^k$
 - 3.1.1. Calcular compatibilidade: $\rho_i^k = 1 - d(v_i^k, x^k)$;
 - 3.1.2. Calcular alerta: $a_i^{k+1} = a_i^k + \beta(1 - \rho_i^k - a_i^k)$;
 - 3.2. **Fim Para**
 - 3.3. **Se** $a_i^{k+1} \geq \tau, \forall i = 1, 2, \dots, c^k$ **então**
 - 3.3.1. $c^k = c^k + 1$;
 - 3.3.2. x^k é o centro do novo grupo: $v_{c^k}^k = x^k$;
 - 3.4. **Senão**
 - 3.4.1. Obter o grupo mais compatível $s = \max_{j=1, \dots, c^k} \{\rho_j^k\}$;
 - 3.4.2. Atualizar o grupo s : $v_s^{k+1} = v_s^k + \alpha(\rho_s^k)^{1-a_s^{k+1}}(x^k - v_s^k)$;
 - 3.5. **Fim Se**
 - 3.6. **Para** $i = 1, 2, \dots, (c^k - 1)$ e $j = (i + 1), \dots, c^k$
 - 3.6.1. Calcular compatibilidade entre centros v_i e v_j : $\rho_{ij}^k = 1 - d(v_i^k, v_j^k)$;
 - 3.6.2. **Se** $\rho_{ij}^k \geq \lambda$ para algum j **então**
 - 3.6.2.1. excluir v_i^k e redefinir v_j^k usando os grupos i e j ;
 - 3.6.2.2. $c^k = c^k - 1$;
 - 3.6.3. **Fim Se**
 - 3.7. **Fim Para**

$$3.8. c^{k+1} = c^k;$$

4. Fim Para

Algoritmo B.1 – Insere pontos, cria, altera e remove grupos.

A inicialização do algoritmo pode ser feita definindo alguns pontos conhecidos e tornando-os centro de seu próprio grupo, onde cada grupo possui um único membro e um índice de alerta inicializado como $a_i^0 = 0$. Desta maneira, o algoritmo começa completamente receptivo ao aprendizado.

Para cada iteração k existem c^k agrupamentos distintos representados pelos centros $v_j^k \in [0,1]^p, j = 1, 2, \dots, c^k$, assim o número de grupos pode permanecer o mesmo, aumentar ou diminuir, dependendo dos valores dos índices de compatibilidade e alerta em relação aos limiares, que por sua vez dependem de τ e λ .

É importante notar que α e β ditam como os centros serão alterados, enquanto τ e λ estão ligados à abrangência e à quantidade de grupos.