

UNIVERSIDADE DO MINHO  
ESCOLA DE ENGENHARIA  
DEPARTAMENTO DE INFORMÁTICA



RACIOCÍNIO BASEADO EM CASOS

*Paulo Novais e José Neves*

Braga  
Abril 1998

## Resumo

O *Raciocínio Baseado em Casos (RBC)* traduz-se numa técnica de resolução de problemas, uma poderosa e rica ferramenta que teve a sua génese na área científica da Inteligência Artificial.

O *RBC* é apresentado como sendo um processo susceptível de utilizar conhecimento dum experiência passada, de um problema ou de um caso similar, para resolver um novo problema. Por outro lado, o *RBC* dá corpo a uma forma de aprendizagem que se quer incremental e sustentada.

Neste trabalho serão atendidas as principais características, áreas de aplicabilidade, vantagens e desvantagens do *RBC*:

- a representação de casos;
- a *recuperação de casos*;
- a *reutilização de casos*; vários são os métodos de adaptação de casos, sendo aqui abordados os procedimentos por *Analogia Transformacional* e *Analogia Derivacional*;
- a *aprendizagem*; que passa pela incorporação do conhecimento a que se teve acesso via a resolução de problemas (casos), na memória de casos.

# Índice

1	INTRODUÇÃO	6
2	RACIOCÍNIO BASEADO EM CASOS	7
2.1	Fundamentos do Raciocínio Baseado em Casos.....	10
2.2	Um Modelo para o Raciocínio Baseado em Casos.....	12
2.3	Vantagens e Desvantagens .....	15
2.4	Aplicabilidade .....	15
2.5	Aplicações Comerciais .....	17
2.6	O Raciocínio baseado em Casos versus outros métodos de Resolução de Problemas .....	17
2.6.1	Indução .....	17
2.6.2	Sistemas Baseados em Regras.....	18
2.6.3	Bases de Dados .....	19
2.6.4	Redes Neurais.....	20
2.7	Combinando o Raciocínio Baseado em Casos com outros métodos de Raciocínio.....	20
3	IMPLEMENTAÇÃO DO SISTEMA DE RACIOCÍNIO BASEADO EM CASOS	21
3.1	Um Caso.....	21
3.1.1	Descrição do Problema.....	22
3.1.2	Solução do Problema.....	22
3.1.3	Justificação da Solução .....	22
3.1.4	Vocabulário .....	23

3.1.5 Representação de um Caso .....	25
3.2 Repositório de Casos .....	26
3.2.1 Modelo de Memória Dinâmica.....	26
3.2.2 Modelo Baseado em Exemplos e Categorias .....	29
3.2.3 Memória não Hierarquizada.....	31
3.3 Recuperação de Casos .....	31
3.4 Reutilização de Casos .....	35
3.5 Aprendizagem.....	39
4 CONCLUSÕES .....	41
5 REFERÊNCIAS .....	42
6 ANEXO .....	46

## Índice de Figuras

Figura 1: Modelo Ilustrativo do Raciocínio Baseado em Casos.....	6
Figura 2: Resolução de Problemas.....	8
Figura 3: Modelo de Raciocínio Baseado em Casos.....	13
Figura 4: Algoritmo para o Raciocínio Baseado em Casos.....	14
Figura 5: Exemplos de Casos .....	23
Figura 6: Exemplo de um Caso Estruturado .....	26
Figura 7: Exemplo de um Episódio Generalizado .....	27
Figura 8: Aranhão de Protos .....	29
Figura 9: Uma instância do Protos.....	30
Figura 10: Recuperação de Casos.....	33
Figura 11: Algoritmo para a Recuperação de Casos .....	35
Figura 12: Adaptação de Casos por Transformação.....	38
Figura 13: Algoritmo de Reutilização de Casos .....	39
Figura 14: Algoritmo de Aprendizagem de Casos .....	40

## Índice de Definições

Definição 1: O Caso.....	7
Definição 2: Raciocínio Baseado em Casos .....	8
Definição 3: Um Caso.....	21
Definição 4: Caso.....	24
Definição 5: Repositório de Casos.....	26
Definição 6: Modelo Mínimo de Casos.....	28
Definição 7: Formulação do Problema.....	32
Definição 8: Similaridade .....	34
Definição 9: Raciocínio por Analogia.....	37

## Glossário

BC	Base de Conhecimento
BDs	Base de Dados
E-MOP	<i>Episodic Memory Organization Packets</i>
MC	Memória de Casos
MM	Modelo Mínimo
MOP	<i>Memory Organization Packets</i>
RBC	Raciocínio Baseado em Casos
RBR	<i>Raciocínio Baseado em Regras</i>
RC	Representação de Conhecimento
RN	Rede Neuronal

# 1 Introdução

A possibilidade de um sistema aprender a partir da observação e da experiência é uma funcionalidade básica de qualquer sistema inteligente; i.e., quaisquer entidades que se tornam conscientes do ambiente que as cercam, que desenvolvem crenças sobre os objectos e os acontecimentos no seu espaço de percepção, dão corpo a um sistema inteligente.

Ora, esta faculdade de acumular conhecimento como ajuda à resolução de problemas, apresenta-se como das facetas mais marcantes da actividade humana. Facultar aos sistemas computacionais este *talento* é um dos objectivos desta área do conhecimento.

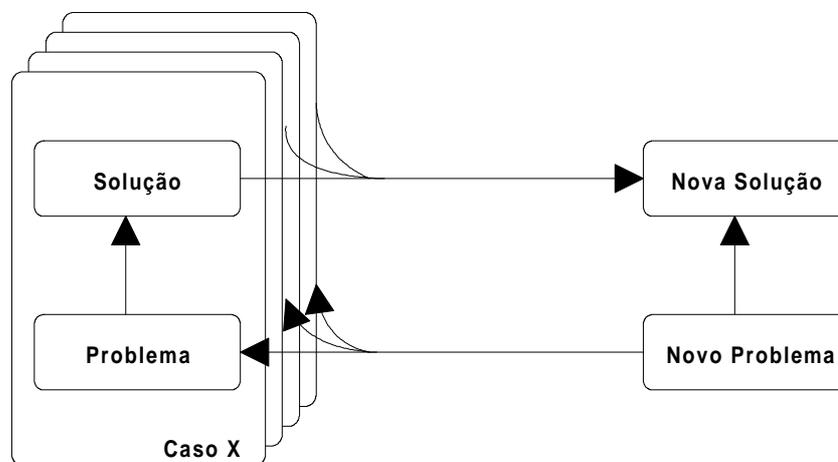


Figura 1: Modelo Ilustrativo do Raciocínio Baseado em Casos

O estudo do *Raciocínio Baseado em Casos (RBC)*, surge com duas motivações [Sc82]: uma motivação cognitiva, que consiste em desenvolver sistemas que emulem o comportamento humano, e uma motivação pragmática, que

consiste em desenvolver sistemas inteligentes que se mostrem eficientes.

Por outro lado, o senso comum leva-nos a concluir que o homem utiliza experiências passadas na resolução de problemas do presente, que a experiência melhora o seu desempenho, e que o conhecimento que tem sobre um assunto é por definição incompleto. Não é por conseguinte de admirar que o *RBC* se assuma como um dos paradigmas analíticos, para a resolução de problemas, mais promissores no campo da *Aprendizagem Automática*.

O *RBC* procura a partir da análise do problema encontrar na memória de casos um caso(s) similar(es), usa-o(s) para sugerir uma solução para o problema, avalia a solução proposta e actualiza a memória de casos, aprendendo com esta experiência (Figura 1).

## 2 Raciocínio Baseado em Casos

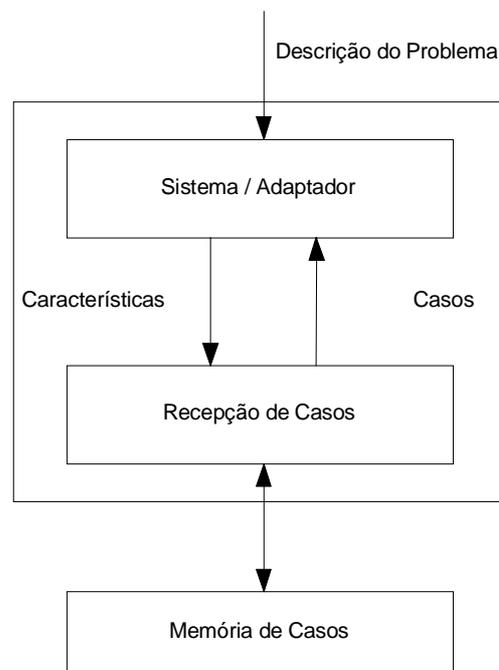
No *Raciocínio Baseado em Casos (RBC)* o ponto de convergência ou foco é colocado no conhecimento disponível, nas associações de casos, na generalização das relações entre os descritores do problema e os casos para que apontam na memória de casos.

Os sistemas de *RBC* são por conseguinte susceptíveis de utilizar conhecimento de uma experiência ou situação passada, ou de um caso similar, para resolver novos problemas (Figura 2). O *RBC* implementa uma vertente da aprendizagem automática que se pretende incremental e sustentada.

*Definição 1: O Caso*

*Os casos traduzem situações que ocorreram no processo de resolução de um problema, consistindo na descrição do problema em questão, da solução encontrada, assim como*

*na sua justificação ou explicação.*



*Figura 2: Resolução de Problemas*

*Definição 2: Raciocínio Baseado em Casos*

*O processo de Raciocínio Baseado em Casos consiste, dado um problema, em encontrar e justificar a solução encontrada, o que passa pela reconstituição de uma experiência ou situação passada, reutilizando e adaptando o conhecimento de tal experiência ou situação a um novo caso, passível de vir a ser inserido na memória de casos.*

Exemplos de RBC passam a ser mencionados:

- um médico, quando examina um doente, aconselha uma terapêutica para o tratamento da(s) enfermidade(s) detectada(s) e, então, regista esse conhecimento sobre a forma de casos. Entretanto, novos pacientes afluem ao seu consultório. Assumindo que existem analogias entre as patologias indiciadas pelos casos da memória de casos, e o novo caso, o do paciente em observação, o clínico usa o conhecimento do passado para tratar o doente do presente;
- um mecânico com experiência, com base na observação e no conhecimento do tipo de ruídos que um veículo pode apresentar, determina qual o problema e também qual a reparação a efectuar;
- um juiz faz uso da jurisprudência (utilização de deliberações sobre situações do mesmo tipo já transitadas em julgado) para se pronunciar sobre um novo caso;
- um analista financeiro quando se propõe atribuir crédito a um cliente, compara o caso presente com casos passados, envolvendo empresas ou indivíduos em situações financeiras semelhantes, passando então à atribuição ou não do crédito;
- um piloto afina a direcção do veículo que lhe está distribuído, usando a sua experiência de condução; i.e., ele sabe, por exemplo, que um desvio deste para a esquerda, quando curva, se pode dever a um mau funcionamento da respectiva suspensão.

O raciocínio por reutilização de casos passados é, sem dúvida, uma poderosa ferramenta na resolução de problemas.

No *RBC*, um caso indica uma situação, uma experiência passada cujo conhecimento foi capturado e assimilado de tal forma que pode ser usado na

resolução de problemas futuros.

Uma das características mais importante do *RBC* é a sua ligação à aprendizagem. No *RBC* opta-se não só por um método de raciocínio, mas também por um processo de aprendizagem suportado por uma actualização de uma base de conhecimento, após concluída a resolução de um problema. A aprendizagem ocorre naturalmente através da resolução de casos. Quando um problema é resolvido, a experiência é retida de forma a resolverem-se casos similares no futuro. Quando a resolução falha, a razão da falha é identificada e lembrada de modo a evitar-se, futuramente, a repetição dos mesmos erros.

## 2.1 Fundamentos do Raciocínio Baseado em Casos

O advento do *RBC* fica a dever-se em grande parte a trabalhos de *Roger Schank* [Sc82] e *Janet Kolodner* [Ko88]. A partir de então o desenvolvimento de modelos e métodos para o *RBC* tem sido constante.

Os processos de *RBC* têm como objectivo, a partir de uma análise prévia de um problema, encontrar caso(s) similar(es), usar caso(s) para sugerir uma solução para o problema, avaliar a solução proposta e actualizar o sistema (Base de Conhecimento - BC), aprendendo com esta experiência.

Existe um conjunto de diferentes métodos para organizar, recuperar, e utilizar o conhecimento passado retido em casos.

Os casos podem ser guardados como experiências isoladas ou agrupados como conjuntos de casos similares na forma de um caso geral. Os casos podem ser armazenados separadamente ou divididos em sub-unidades e apresentarem-se estruturados. A solução de um caso anterior pode ser aplicada directamente ao problema em mãos, ou modificado de acordo com as diferenças entre os (dois) casos. A adaptação de soluções e formas de aprendizagem com base em casos

pode ser suportada e guiada por formas de representação de conhecimento e raciocínio do domínio geral ou, muito simplesmente, baseado em aparentes similaridades sintácticas entre os casos. Um *RBC* pode ser automático ou interactivo, estando, nesta última variante, o utilizador com o controlo das operações.

### **Raciocínio Baseado em Memórias**

As memórias de eventos passados são usadas directamente nas tomadas de decisão. Raciocinar fica assim reduzido à percepção; i.e., a situação corrente é observada, procura-se na memória de casos os que mais se aproximam da situação corrente e, utiliza-se esse conhecimento para a resolução do problema [StWa88]. Os casos são organizados vectorialmente, denotando atributos e relações, de modo análogo ao que acontece na criação e manutenção das bases de dados. As formas de organização e acesso à memória de casos são peças centrais para a análise de desempenho deste tipo de sistemas.

### **Raciocínio Baseado em Exemplos**

Resolver um problema, que é passado como um caso, não passa além de um simples processo de classificação; i.e., há que procurar a classe que irá reter o caso não classificado. Nessa classe ir-se-á procurar, então, uma solução para o problema. O conjunto de classes torna-se, nesta situação, no conjunto das possíveis soluções. As tarefas de classificação são aqui efectuadas a partir de casos passados, em vez de o serem através de regras, como é habitual nos sistemas tradicionais. Um sistema como o *Protos*, destinado ao diagnóstico de patologias auditivas, é um exemplo de aplicação deste método [BaPoWi90].

## Raciocínio Baseado em Casos

Assume-se que cada caso é rico em conhecimento e se encontra bem estruturado. Este método está preparado para modificar e adaptar uma solução recuperada a um contexto diferente daquele em que a resolução do problema inicial foi equacionada [Sc82].

## Raciocínio Baseado em Analogia

Este método equaciona a resolução de novos problemas baseando-se em casos passados oriundos de diferentes áreas ou domínios de conhecimento [Sh88] [KlCa88]. Este tipo de raciocínio é típico do método de adaptação de casos [Ca86].

## 2.2 Um Modelo para o Raciocínio Baseado em Casos

De um modo geral há que [Sl91] [Ko93] [AaPl94]:

- **recuperar** o(s) caso(s) mais similar(es);
- **adaptar** o conhecimento e a informação contida no(s) caso(s) para resolver o problema;
- **rever e reparar** a solução proposta;
- **apreender** com a experiência presente, por forma a que esta possa ser útil na resolução de problemas futuros.

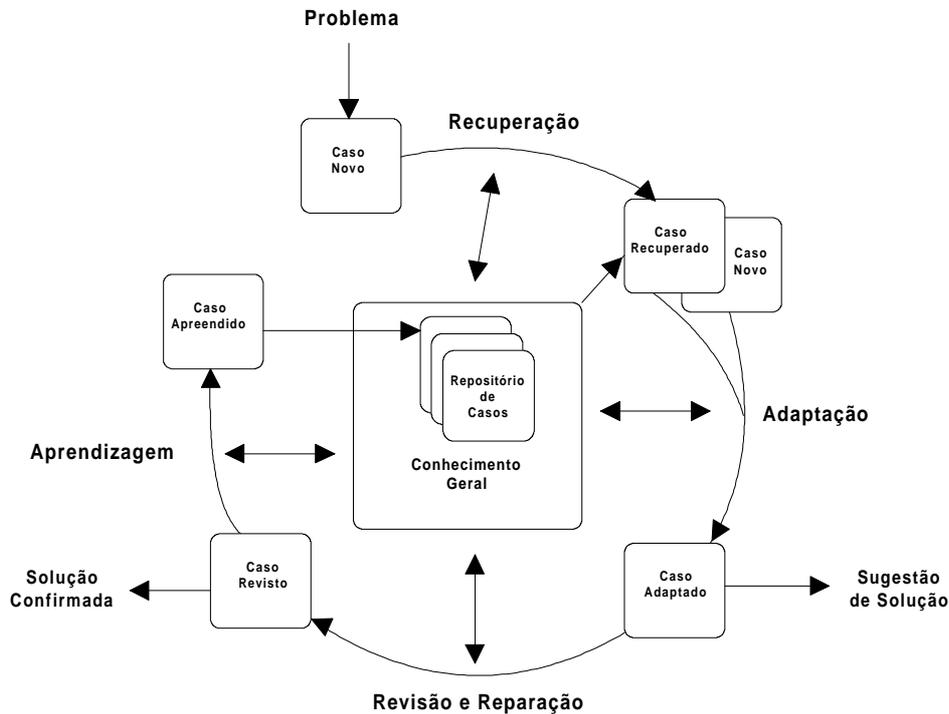


Figura 3: Modelo de Raciocínio Baseado em Casos

Como se pode ver a partir da Figura 3, a descrição inicial do problema traduz-se num caso, o qual por sua vez é usado na recuperação de um ou mais casos a partir do conjunto de casos passados. Esta recuperação tanto pode ser baseada em similaridades sintáticas entre os casos (abordagem mais comum), como feita a partir de similaridades semânticas; i.e., pela forma ou pelo contexto. Uma solução para o caso proposto surge da combinação do caso recuperado com o caso proposto, através de um processo de adaptação. O processo de revisão testa a solução, aplicando a solução proposta à situação concreta, ou evoluindo com base num controlo exterior, corrigindo-se a solução se necessário (*reparação*). Durante o processo de aprendizagem o caso é assimilado para futuras utilizações e a base de conhecimento é actualizada com o novo caso (Figura 4).

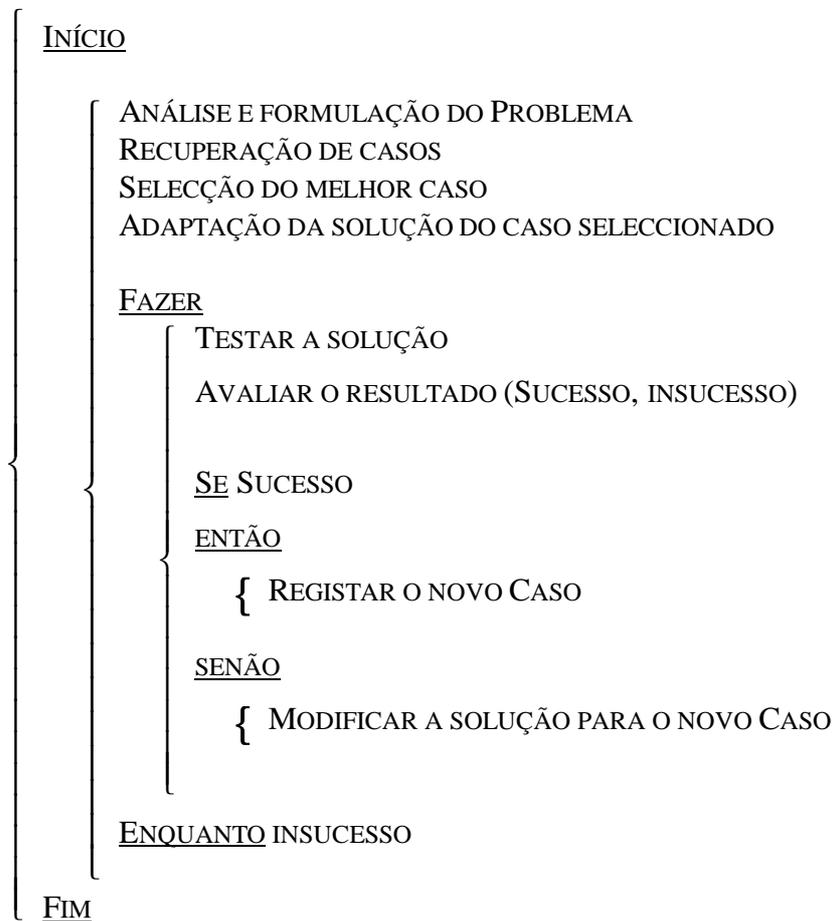
**ALGORITMO O RBC**

Figura 4: Algoritmo para o Raciocínio Baseado em Casos

O *conhecimento geral* suporta o processo, representando os *casos* o conhecimento específico; i.e., um determinado conjunto de situações que já transitaram em observação e experiência. O *conhecimento geral* representa o conhecimento numa determinada área. No exemplo do *meccânico*, o *conhecimento geral* pode ser entendido como contendo as descrições técnicas de um determinado veículo, sendo os *casos* as descrições das reparações já efectuadas em veículos desse tipo.

### 2.3 Vantagens e Desvantagens

Tradicionalmente, os procedimentos em sistemas inteligentes eram descritos através do uso de regras, dando corpo a uma forma de raciocínio denominada de *Raciocínio Baseado em Regras - (RBR)*. Expressar o conhecimento nestes termos, em situações reais, é difícil e moroso [Ha85]. Em contraste, num qualquer sistema de *RBC*, os casos são o conhecimento, o que leva a que a manutenção e actualização do conhecimento em tais sistemas seja feita automaticamente, enquanto que em sistemas de *RBR* isso implica a escrita de novas regras.

Este tipo de sistema propõe uma solução para um dado problema, a partir de conhecimento não completamente definido, pouco estruturado ou desconhecido; permite ainda que se atenda durante o processo de construção de uma solução, aos aspectos/características do problema considerados determinantes para a construção dessa mesma solução.

Por outro lado, o facto de se estarem a relembrar experiências passadas dá uma alerta para potenciais problemas que aí porventura tenham ocorrido, e que há que ultrapassar.

Há, contudo, um senão. Com o tempo, o repositório de casos aumenta e a sua manutenção torna-se difícil, assim como nem sempre é possível garantir que se parte do conjunto de casos passados mais apropriado, quando há que abordar um novo problema.

A utilização de métodos incrementais [Aa93], explicativos [Aa94], tem contudo constituído uma saída para este tipo de situações.

### 2.4 Aplicabilidade

Existem domínios do conhecimento nos quais o processo de *RBC* na resolução de problemas é particularmente atractivo, sendo de referir, entre outras as áreas do

diagnóstico, planeamento e desenho são, entre outras, áreas de aplicação relevantes para o *RBC*.

### **Diagnóstico**

A partir de um conjunto de sintomas há que elaborar sobre as causas. Os sistemas de diagnóstico baseados em casos tentam recuperar aqueles cuja lista de sintomas é análoga à do novo caso, sendo a solução dada com base nessas experiências [De95]. São exemplos deste tipo de sistemas o sistema o CASEY [Ko93], e o *Sistema de Diagnóstico Multiagente* [No96].

### **Planeamento**

Partindo do pressuposto de que é mais difícil criar um artefacto a partir de um conjunto de especificações e técnicas, do que classificar um artefacto existente com base num conjunto de protótipos conhecidos, pode-se caminhar para a simplificação do acto da criação.

Planos passados são guardados na memória de casos e são recuperados em função do estado inicial do problema e do objectivo a alcançar (planeamento de tarefas). Um exemplo deste tipo de sistemas é-nos dado pelo sistema PRODIGY [Ve+94].

### **Desenho**

O desenho é uma tarefa complexa, onde o processo de criação se assume na sua plenitude. Existem similaridades com sistemas de planeamento, no entanto a adaptação no desenho é substancialmente mais complexa, na medida em que os passos intermédios não são condicionantes do produto final.

Estes sistemas tendem a limitar-se à assistência ao utilizador, fornecendo

desenhos passados (recuperação de casos).

## **Interpretação e Classificação**

A interpretação caracteriza-se pela existência de uma situação nova, que é apreciada tendo em conta situações passadas. É útil quando não existem outros meios de obter uma solução (eg. em problemas em que o grau de incerteza e subjectividade é tal que é praticamente impossível a sua resolução por utilização de outras vias, dado que os universos de discurso são pouco estruturados ou de difícil conceptualização). É usada em tarefas de classificação, justificação ou previsão [St94], assim como na avaliação de efeitos de uma decisão ou plano [Al+95]. Em particular, a jurisprudência posiciona-se como uma das áreas, de maior potencial para a aplicação deste método.

## **2.5 Aplicações Comerciais**

As aplicações conhecidas não implementam o modelo de *RBC* em todas as suas vertentes, ficando-se, normalmente, pela recuperação de casos. Não fornecem, em particular, suporte à adaptação de casos [Al+95].

## **2.6 O Raciocínio baseado em Casos versus outros métodos de Resolução de Problemas**

### **2.6.1 Indução**

Por indução pode extrair-se informação por aplicação de árvores de decisão (generalização das árvores de prova) ou de regras (resolução por inversão) [So94], ao conhecimento presente na memória de casos.

A grande diferença entre uma abordagem puramente indutiva e um sistema

de *RBC*, está em que a indução cria uma abstracção da memória de casos; i.e., parte de uma generalização da memória de casos e utiliza (somente) este conhecimento na resolução de problemas, enquanto que o *RBC* utiliza directamente os casos passados (experiência).

Um sistema puramente indutivo apresenta não só alguns senões na manipulação de informação incompleta durante o processo de generalização, mas também obsta a que a informação sobre os exemplos dados não seja mais acessível depois de passada a fase de generalização.

Uma possível utilização da prática indutiva em *RBC* está na redefinição do processo de assimilação de conhecimento, tal como é apresentado por *Kowalski* [Kow90]; i.e., o conhecimento obtido por uma generalizado da memória de casos é uma *Teoria*, na qual se irá tentar inserir um novo termo (o novo caso).

### 2.6.2 Sistemas Baseados em Regras

Desenvolver sistemas inteligentes baseados em regras é uma tarefa difícil. Descortinar as regras relevantes para um dado domínio do conhecimento é um processo deveras complexo, na medida em que é mais fácil a um especialista resolver um problema prático do que explicar como o resolveu. As regras são difíceis de articular e avaliar, enquanto que os casos são facilmente estruturáveis e fáceis de encontrar.

Através de regras têm-se grandes dificuldades em representar conceitos em abstracto, enquanto que o *RBC* fornece um bom fundamento para o armazenamento de conceitos acompanhados de exemplos (não raras vezes com a ajuda de Bases de Dados), dando uma visão mais aproximada do mundo real.

Nos processos de inferência os sistemas baseados em regras disparam um número muito elevado destas, afectando, por conseguinte, e de forma drástica, o

seu desempenho. Os sistemas baseados em *RBC*, por outro lado, não enfermando da limitação anterior, podem, ainda, evitar erros do passado, além de terem a capacidade de gerar novas soluções através de um processo de generalização.

As regras são dirigidas para a representação de situações e para os processos de manipulação dessas mesmas situações; i.e., actuam com base no poder do raciocínio sobre o conhecimento. Nos sistemas de *RBC* usam-se as situações do passado como modelos para a concepção de explicações no presente; i.e., as justificações são parte constituinte dos casos.

Os métodos baseados em casos são incrementais. Os novos casos são introduzidos na memória de casos. A tarefa de manutenção de sistemas baseados em regras é mais complicada na medida em que para cobrir um determinado domínio, é necessário um grande número destas.

### 2.6.3 Bases de Dados

As Bases de Dados (BDs) e os sistemas baseados em *RBC* têm muito em comum.

Sendo o armazenamento e a recuperação os aspectos principais do *RBC*, uma questão natural é a relação que existe entre sistemas de *RBC* e BDs.

Um sistema *completo* de *RBC* adapta o caso recuperado, as BDs simplesmente ignoram este processo.

As BDs deixam o ónus da interrogação ao utilizador. No *RBC*, o processo de resolução de problemas inicia-se com uma descrição do problema, a partir de uma caracterização dos descritores.

As BDs estão preparadas para encontrar a combinação entre a componente da interrogação feita à base e a informação armazenada, enquanto que o objectivo do *RBC* é recuperar o caso mais similar com o novo caso ou problema.

As BDs manipulam grandes quantidades de informação, enquanto que os

sistemas *RBC* são comparativamente mais comedidos.

As BDs operam sem qualquer conhecimento do problema que estão a resolver, limitando-se a fornecer os mecanismos de pesquisa e recuperação de informação. Os sistemas de *RBC* usam o conhecimento acerca do problema em equação para construir os mecanismos a utilizar na recuperação de casos.

#### **2.6.4 Redes Neurais**

As Redes Neurais (RNs) e o *RBC* utilizam casos, no entanto nas RNs os casos servem apenas para o treino e teste da rede.

As RNs têm melhor desempenho que os sistemas *RBC* em domínios com informação incompleta ou mesmo de falta total de informação.

Nas RNs o utilizador não pode, em princípio, julgar (ou avaliar) as decisões da rede.

Os sistemas de *RBC* lidam com informação que se encontra, comparativamente, melhor estruturada.

### **2.7 Combinando o Raciocínio Baseado em Casos com outros métodos de Raciocínio**

Diferentes combinações do *RBC* com outras formas de raciocínio têm vindo a ser estudadas, sendo de reter entre outras as associações:

- com técnicas indutivas de aprendizagem, em tarefas de diagnóstico [Aur+95], [ArPI94];
- com sistemas de *RBR*, em que os casos podem guiar a interpretação de regras [Le95]; ou integrarem-se com estas [Ló92] [Ne+95];
- em que os sistemas de *RBC* surgem como componentes de um sistema

de estratégias múltiplas de aprendizagem [RisSk89] [PIArc93];

- com RNs na fase de recuperação de casos [No96];
- com arquitecturas integradas [Ve+94].

Concluindo, os sistemas de *RBC* podem aparecer associados a outros métodos ou ser suportados por outros métodos; i.e., desempenhar um papel de charneira entre diferentes sistemas.

### 3 Implementação do Sistema de Raciocínio Baseado em Casos

A implementação<sup>1</sup> do sistema *RBC* vai depender do tipo de funcionalidades a considerar, das técnicas de representação do conhecimento a utilizar, da arquitectura do sistema computacional. A matéria prima é o caso, sendo de vital importância o aprofundamento da representação do conhecimento que transporta, assim como dos modos de raciocinar que lhe podem ser aplicáveis.

#### 3.1 Um Caso

*Casos são exemplos de situações que ocorreram no passado* [AlWe91]; i.e., reportam-se a eventos ou processos (problemas e suas soluções) que ocorreram e que podem ser limitados no espaço e no tempo, e que foram resolvidos (Definição 3).

*Definição 3: Um Caso*

*Um caso é uma tripla  $\langle P, S, J \rangle$ , onde  $P$  é a descrição do*

---

<sup>1</sup> Não existem soluções universais; i.e., uma implementação que cubra todos os domínios do conhecimento. Pretende-se dar uma ideia global de como se pode implementar um sistema deste tipo.

*problema, S é a solução para o problema, e J é uma justificação para a solução do problema. Um caso corresponde a um evento real ou a um processo que pode ser limitado no tempo e no espaço. As justificações são uma representação do processo de resolução do problema e/ou dos métodos utilizados na sua resolução.*

### **3.1.1 Descrição do Problema**

A formulação do problema deve contemplar o contexto em que o mesmo é colocado, e ser caracterizada por uma descrição deste que tenha em conta:

- os objectivos a alcançar com a resolução do problema; i.e., o que se pretende obter dando corpo a uma solução;
- as restrições (invariantes) que condicionam os objectivos a alcançar.

### **3.1.2 Solução do Problema**

A solução para um problema deve ser constituída por:

- uma instância de uma solução possível;
- o conjunto de passos usados para a sua concretização;
- porventura outras instâncias de soluções, assim como possíveis excepções (soluções não possíveis) [Ko93];
- o(s) resultado(s) obtido(s) com a aplicação da solução.

### **3.1.3 Justificação da Solução**

Uma justificação de uma solução pode ser dada na forma:

- pelos métodos de resolução de problemas utilizados;

- pelas justificações ou explicações das decisões tomadas, quer por parte do sistema computacional, quer por parte do utilizador, quando são chamados a intervir.

	Marca	Modelo	....	Problema	....	Solução	....
Caso 1	Renault	12	....	Falta de travões	....	....	....
Caso 2	Ford	Fiesta 1.8 D	....	Motor não pega	....	....	....
Caso 3	Ford	Fiesta 1.6	....	Motor não pega	....	....	....
....							

Figura 5: Exemplos de Casos

Embora nem sempre todos estes aspectos sejam tomados em consideração, quando há que justificar uma solução (nomeadamente nas situações de insucesso na procura de uma solução), são no entanto um ponto de referência.

### 3.1.4 Vocabulário

A escolha do vocabulário ou linguagem utilizada na descrição dos casos é função da abordagem seguida para a resolução do(s) problema(s) [Ko93] [Ve94]:

- abordagem *funcional*; a escolha do vocabulário passa pela análise do tipo de casos disponíveis e das tarefas que têm que ser suportadas pelo sistema;
- abordagem por *captura*; a escolha do vocabulário passa pela análise das características dos casos que são utilizados pelo utilizador para a certificação de tarefas a cargo do sistema, tendo em conta a necessidade de estabelecer pontes entre casos presentes e casos passados. Esta abordagem está ligada à captura

das nuances que caracterizam um determinado domínio ou universo de discurso.

O conjunto de características relevantes deve ser o mais reduzido possível, por forma a permitir que o processo de recuperação de casos seja eficiente. Estão nesta situação os índices dos registos dos casos, que são um subconjunto de um conjunto mais vasto de factores que descrevem e representam os casos.

*Definição 4: Caso*

*Seja  $A = \{A_1, \dots, A_n\}$   $n \geq 1$ , um conjunto de variáveis denominadas atributos; para cada atributo  $A_i$  ( $i = 1, \dots, n$ ), seja  $R_{ij}$  ( $j = 1, \dots, m$ ), um conjunto não vazio de valores denominado domínio de  $A_i$ ; seja  $S$  uma variável denominada solução; e seja  $D$  um conjunto não vazio de valores denominado domínio de  $S$ . Um caso é uma tripla  $C = (CD, Sol, Jus)$  em que:*

- *$CD \subseteq \{(A_i, V) \mid A_i \in A \text{ e } V \in R_{ij}\}$  em que cada  $A_i$  ocorre pelo menos uma vez em  $CD$ ; em que  $CD$  é denominada de descrição de um caso  $C$ .*
- *$Sol = (D, S)$  onde  $S \in D$ .  $Sol$  é denominado solução do caso  $C$ .*
- *$Jus$  é a justificação da solução dada para o caso.*

Um exemplo de um caso é dado a seguir:

Imagine que se encontra perante um cidadão que procura alívio para uma ligeira indisposição, tendo-se informação acerca do sexo, idade, e estado febril. Por sua vez para cada um destes parâmetros tem-se um domínio: o domínio  $R_1$  do atributo *sexo* é

o conjunto  $\{masculino, feminino\}$ , o domínio  $R_2$  do atributo *idade* é o intervalo  $[0...120]$ , o domínio  $R_3$  do atributo *febre* é o intervalo  $[34...43]$ . Uma possível descrição deste caso é-nos então dada por:  $CD=\{(Sexo, feminino), (Idade, 25), (Febre, 40)\}$ . A solução para o problema tomará então a forma:  $Sol=(Diagnóstico, gripe)$ . Uma possível justificação para a solução passa a ser  $Jus=Tomar antibióticos$ .

### 3.1.5 Representação de um Caso

#### Representação não Hierarquizada

Um caso é representado por tuplos (Figura 5 e Definição 4). Esta forma de representação acarreta problemas, nomeadamente no que se refere a fenómenos que sejam desconhecidos, irrelevantes ou incompletos, não permitindo ainda a partição de casos. É no entanto a representação mais utilizada.

#### Representação Estruturada

Uma possível solução para o problema está na representação estruturada de casos, através da divisão de um caso em termos das suas características ou propriedades, o que potencia a conceptualização do caso via objectos [P195]. A cada objecto estão associados atributos que, por si, podem desqualificar ou não os seus pares (eg. quando se é colocado face a um certo tipo de habitação, constata-se que os atributos que qualificam umas, desqualificam outras (Figura 6)).

	Tipo	Morada	....	Quarto 1	Quarto 2	....
Caso 1	T1	Porto	....	4 x 5	Irrelevante.	....
Caso 2	T2	Guimarães	....	4 x 5	3 x 5	....
Caso 3	T0	Braga	....	Irrelevante	Irrelevante	....
....						

Figura 6: Exemplo de um Caso Estruturado

A este conhecimento, aquele que está associado a casos passados (repositório de casos), dá-se o nome de *Conhecimento Episódico*.

### 3.2 Repositório de Casos

Um dos factores de sucesso deste tipo de sistemas está no manuseamento fácil e rápido de casos. Um dos modelos mais conhecidos dá pelo nome de *Modelo de Memória Dinâmica* [Sc82], utilizado no sistema *CYRUS* de *Kolodner*, um sistema de pergunta/resposta [Ko93].

*Definição 5: Repositório de Casos*

*Um Repositório de Casos (RC) ou Base de Casos (BC) é um conjunto  $\{C_1, \dots, C_m\}$ ,  $m \geq 0$ , onde  $C_i$  é um caso sobre  $A$ , sendo  $A$  um conjunto de atributos do caso.*

#### 3.2.1 Modelo de Memória Dinâmica

Esta forma de modelar o repositório de casos surge em resultado do trabalho desenvolvido por *Roger Schank* [Sc82], dirigido à compreensão do funcionamento da mente humana, de modo a simulá-la em computador. *Schank* partiu do princípio de que o conhecimento deve derivar da experiência.

O *Modelo de Memória Dinâmica* dirige-se em particular à forma de processar no presente o conhecimento obtido com situações do passado, assim como ao modo em como é feita a integração no repositório de casos do conhecimento derivado de novas experiências.

*Schank* propôs uma solução baseada numa estrutura hierárquica, organizada através de *MOPs* (*Memory Organization Packets*), a qual passou por organizar os casos atendendo às propriedades que têm em comum, agrupando-os.

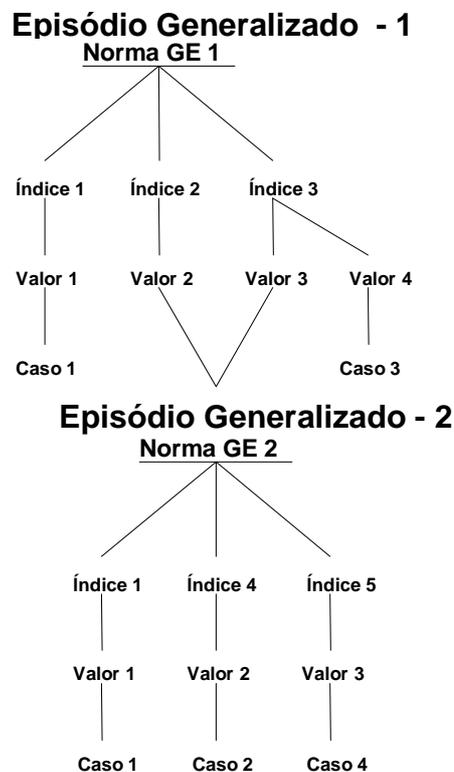


Figura 7: Exemplo de um Episódio Generalizado

No sistema *CYRUS* (já referido) a memória de casos está organizada em *E-MOPs* (*Episodic Memory Organization Packets*), ou seja, segundo uma estrutura

em que os casos que foram objecto de estudo passaram por processo de generalização. A memória de casos é, por conseguinte, uma estrutura hierárquica de *Episódios Generalizados*.

Um *Episódio Generalizado* contém três tipos de termos: *normas*, *casos* e *índices*. As *normas* são termos que dão corpo às características comuns a todos os casos indexados a um certo *Episódio Generalizado*. Os *índices* são termos que se referem a características que permitem distinguir cada um dos casos na estrutura, podendo indiciar um *Episódio Generalizado* ou directamente um *caso*, sendo compostos por um par (*atributo*, *valor*).

A estruturação da memória de casos é, por definição, dinâmica, no sentido em que dois casos, em princípio complementares, dão corpo a um *Episódio Generalizado* e, dentro deste, as partes são indexadas segundo a sua especificidade. A memória de casos traduz-se numa rede, onde cada nodo é um *Episódio Generalizado* ou um *caso*. A estrutura de índices é de certo modo imaterial (Figura 7), na medida múltiplos são os caminhos que nos conduzem a cada *caso* [AaPI94].

A recuperação de um *caso* passa por se encontrar o *Episódio Generalizado* que melhor se identifique com a descrição do problema, partindo-se para a pesquisa a partir do nodo raiz. Para a retenção de um *caso* o processo a seguir é análogo, havendo que criar-se, porventura e sempre que se justificar, um novo *Episódio Generalizado*. Apesar da aparente simplicidade, a implementação desta estrutura é uma tarefa algo complexa e os senãos a ultrapassar são diversos. A solução para este problema pode ser encontrada na ligação entre a *Memória de Casos* (*MC*), o seu *Modelo Mínimo* (*MM(MC)*) e cada um dos *Casos* (*C*) concretos em presença [NeNo98]:

*Definição 6: Modelo Mínimo de Casos*

$MC \models C$  se e somente se  $C \in MM(MC)$ ; i.e.,  $MM(MC)$  é

*exactamente o conjunto de todos os casos que são implicados por MC.*

*Sendo  $\models$  a relação de implicação lógica.*

### 3.2.2 Modelo Baseado em Exemplos e Categorias

No sistema *PROTOS* [BaPoWi90] faz-se uma proposta, considera-se a aprendizagem e o raciocínio a partir de *exemplos* (aqui os *casos* são os *exemplos*) (Figura 8).

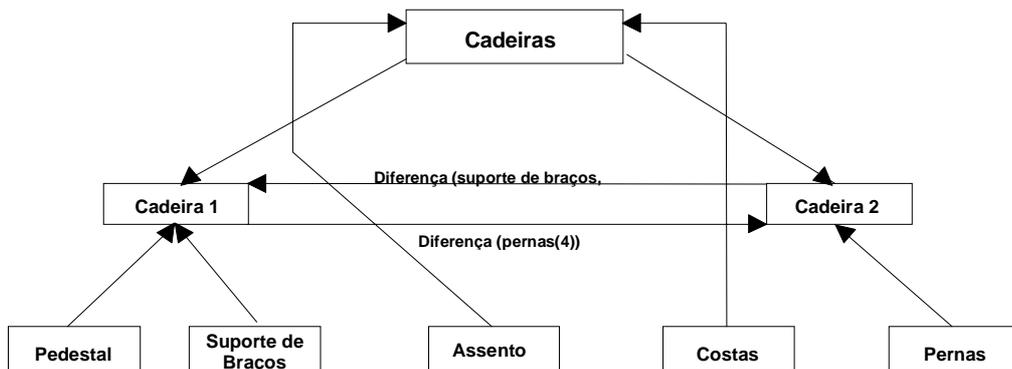


Figura 8: Aranhaço de Protos

A organização do conhecimento é dada por uma rede semântica que se interliga com uma estrutura de categorias ou conjuntos de casos similares (Figura 9). Esta interligação cria a *memória de casos*, sendo cada exemplo ou caso dentro de uma categoria utilizado de acordo com o seu grau de usabilidade (i.e., existem *pares* e *primus inter pares* dentro de uma mesma categoria).

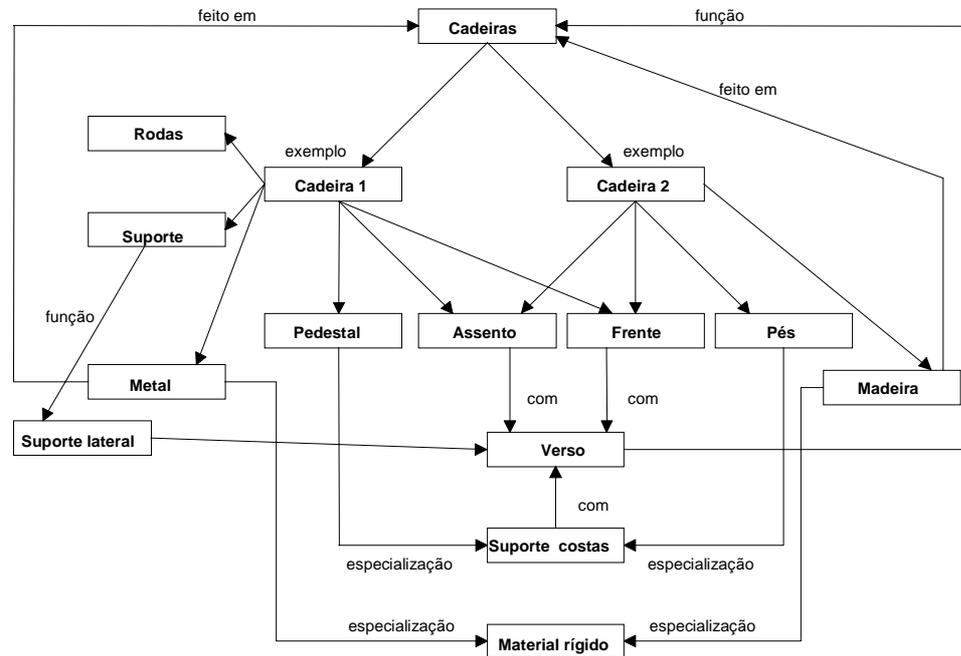


Figura 9: Uma instância do Protos

Consideram-se três tipos de mecanismos para o tratamento de índices em *Protos*:

- o tipo *associativo*, que faz a ligação, no aranhão, entre um descritor do problema e um exemplo ou categoria;
- o tipo *protótipo*, que faz a ligação, no aranhão, entre um descritor do problema, uma categoria e, dentro desta, entre a raiz e os casos seleccionados;
- o tipo *diferenças*, que faz a ligação, no aranhão, entre um descritor do problema e os exemplos ou casos mais próximos, dando particular realce aos pormenores que, entre estes, mais os diferenciam.

As diferenças de traço na Figura 8 traduzem o facto de aí nem todas as ligações terem o mesmo peso.

Para procurar um caso na *memória de casos*, há que tratar os descritores do problema por forma a gerar um apontador para um caso ou categoria que, no mínimo, partilhe o máximo de descritores com o primeiro. É um processo de classificação. Se a ligação for do tipo *associativo*, aponta para uma categoria, e o caso que é seleccionado apresenta-se como uma possível solução para o problema (em termos de um protótipo).

### 3.2.3 Memória não Hierarquizada

Existem outros tipos de organização da *memória de casos*, embora não tão complexos, como os de *memória horizontal*, em que os casos são armazenados sequencialmente, não obedecendo a qualquer tipo de estruturação. Não há, neste caso, qualquer possibilidade de indexação.

## 3.3 Recuperação de Casos

A recuperação de casos tem como objectivo seleccionar o caso (ou casos) mais consequente(s) com a descrição do problema (Figura 10), podendo-se dividir este procedimento em três partes, nomeadamente:

- na *formulação* do problema;
- na *procura* dos casos que combinam com estes descritores;
- na *selecção* destes últimos.

### Formulação do Problema

Para a formulação do problema procuram-se os descritores, a utilizar dentro de um

certo contexto, na procura de casos passados.

*Definição 7: Formulação do Problema*

*Seja  $A=\{A_1, \dots, A_n\}$ , um conjunto de variáveis denominadas atributos (Definição 4). Para cada atributo  $A_i$ , seja  $R_i$  o domínio de  $A_i, i=1, \dots, n$ . Um problema, ou novo caso, pode então ser equacionado em termos do conjunto  $P=\{(A_i, v_j) \mid A_i \in A, v_j \in R_i, j=1, \dots, m\}$ , em que  $A_i$  ocorre pelo menos uma vez em  $P$ .*

## **Procura**

Na fase de *procura* tem-se como objectivo obter o conjunto de casos que estão *suficientemente* próximos do novo caso (Figura 10). Esta operação pode ser feita com base nos descritores do problema ou em descritores obtidos a partir destes, assim como a partir da forma de estruturação dos casos, com eventual recurso a conhecimento do senso comum para validar opções.

Neste processo várias são as estratégias que podem ser seguidas, indo-se desde uma escolha de casos que partilhem o maior número possível de descritores com o problema [Ko88], à quantificação da qualidade dos descritores [BaPoWi90].

## **Seleccção**

Neste processo quando mais do que um caso é recuperado da memória de casos, há que escolher, por seleccção, qual o caso(s) mais relevante(s).

A cada caso é associado uma medida de similaridade (que pode já ter sido calculada na fase de **procura**), sendo seleccionado o caso que apresente o maior valor para esta.

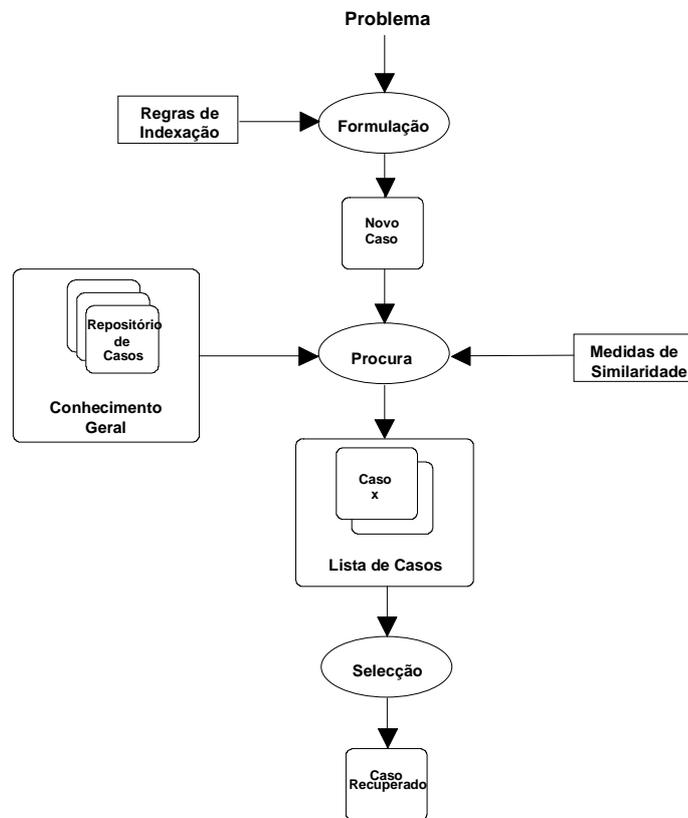


Figura 10: Recuperação de Casos

## Similaridade

Na recuperação de casos a palavra chave é *similaridade*. A medida de similaridade pode ser obtida tendo em conta preferências heurísticas ou o cálculo de uma métrica.

Quando se estrutura a memória de casos (tipo modelo de memória dinâmico), normalmente utilizam-se medidas de similaridades que têm em conta preferências heurísticas; i.e., há que seleccionar o caso mais conveniente para a operação de recuperação, o que nem sempre nos leva ao caso com mais similar

com o novo caso.

A similaridade entre dois casos [Al+95] (o novo caso e um dos casos recuperados) baseia-se no cálculo da medida de similaridade entre cada atributo de cada caso (similaridade local), agrupando-se então as medidas de similaridades locais numa medida global, normalmente tomadas no intervalo [0,1].

*Definição 8: Similaridade*

***Seja  $CD=\{cd_1, cd_2, \dots, cd_n\}$  a descrição de um caso; seja  $P=\{p_1, p_2, \dots, p_n\}$  a descrição de um problema. A similaridade entre  $P$  e  $CD$  é então dada por:***

$$SIMILARIDADE(P, CD) = F(Sim_1(p_1, cd_1), Sim_2(p_2, cd_2), \dots, Sim_n(p_n, cd_n))$$

***em que  $F()$  é uma medida de similaridade [RiWe91].***

Em termos dos modelos de organização da memória de casos, viu-se o modelo de memória dinâmica (eg. no sistema *CYRUS*), em que existem similaridades sintáticas entre os descritores do problema e os dos casos passados, assim como o modelo baseado em exemplos e categorias (eg. no sistema *PROTOS*), em que ocorrem similaridades do foro semântico ou interpretativo (Figura 11).

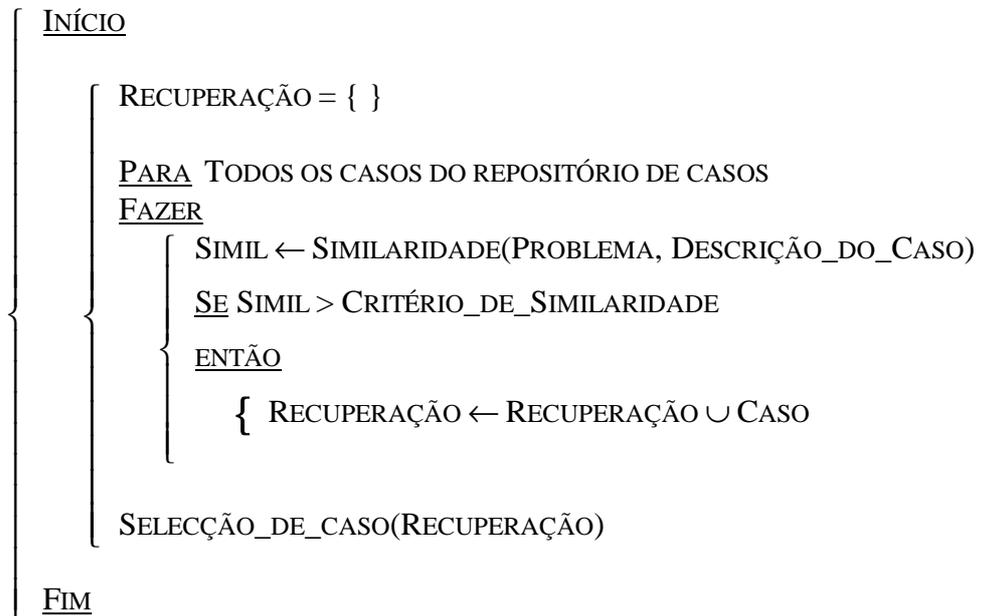
**ALGORITMO RECUPERAÇÃO DE CASOS**

Figura 11: Algoritmo para a Recuperação de Casos

### 3.4 Reutilização de Casos

O objectivo da reutilização de casos está no uso do conhecimento do(s) caso(s) passado(s), devolvido pela recuperação de casos, na resolução de um novo problema. A reutilização de casos pode envolver alguma modificação da solução associada a um ou mais casos passados.

Como já foi mencionado (Capítulo 2), este procedimento passa pela junção do processo de *Adaptação* de casos aos processos de *Revisão* e *Reparação*. Por outro lado, pode-se restringir a *reutilização de casos*:

- à *adaptação* do conhecimento deste(s) último(s) face a informação trazida pelo novo caso;
- ao *teste* da solução;
- à *reparação* da solução.

## **Adaptação**

A solução protagonizada pelo caso recuperado não é, normalmente, directamente aplicável ao novo caso, tendo de ser adaptada à nova situação.

A adaptação da solução pode passar por uma simples transferência de conhecimento do caso recuperado para o novo caso (solução e justificações), ou então ser objecto de processamento prévio com base, por exemplo, em processos de pensamento ou raciocínio por analogia (eg. produtos como a *Analogia Transformacional* [Ca83] ou a *Analogia Derivacional* [Ca86]).

## **Teste**

O teste da solução proposta é normalmente realizado fora do sistema de *RBC*, e passa por simular a aplicação da solução encontrada à resolução do problema. Se a operação tiver sucesso, então o caso pode ser assimilado na memória de casos, caso contrário o caso passa à fase de recuperação.

## **Reparação**

A reparação da solução proposta como solução para o problema, envolve a detecção de erros ou falhas que porventura nesta sejam detectadas. Esta operação de reparação faz, por outro lado, apelo ao uso de conhecimento específico (eg. fontes de conhecimento, regras de reparação) ou ao utilizador do sistema (eg. sugestões).

## **Adaptação por Substituição**

Em problemas em que o universo de discurso é conhecido e o caso recuperado aproxima-se da descrição do novo caso, em geral a adaptação da solução proposta não é uma tarefa complicada; i.e., a alteração de valores de atributos da solução

será suficiente, em princípio, para a sua consolidação, não havendo por isso que recorrer a alterações na estrutura da solução.

## **Raciocínio por Analogia**

A resolução de problemas por analogia passa pela transferência de conhecimento respeitante a soluções de problemas passados, para a solução do novo problema, o que é o caso quer na *Analogia Transformacional* [Ca83] quer na *Analogia Derivacional* [Ca86].

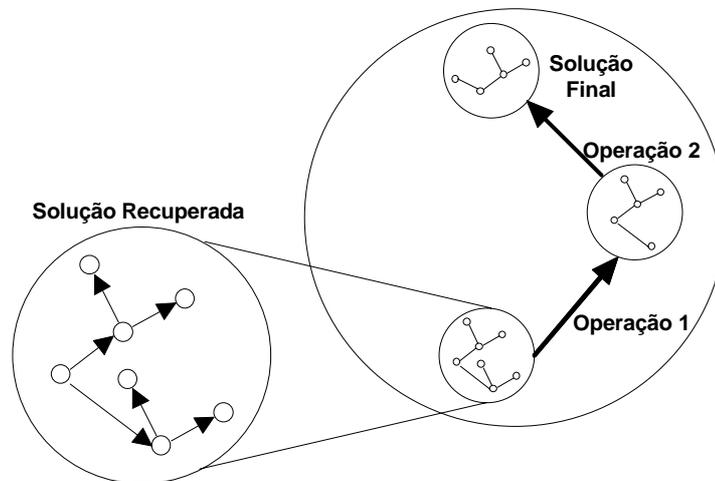
*Definição 9: Raciocínio por Analogia*

*Na resolução de problemas por analogia usa-se o conhecimento de soluções de casos passados para construírem-se as soluções de novos problemas.*

No que respeita à *Analogia Transformacional*, o procedimento de transformação da solução é implementado com recurso a um conjunto de operadores atómicos, num processo que passa pela selecção das acções que conduzam um caso recuperado a transformar-se na solução para o novo problema; i.e., a tarefa fundamental da adaptação de soluções por transformação passa por encontrar a adequada sequência de operações que transformem a solução do caso recuperado na solução desejada para o novo caso (Figura 12).

São exemplos de operadores de transformação os operadores de *substituição, inserção, remoção* ou *reordenação*.

A *Analogia Transformacional* não olha à forma de resolução do problema, mas à equivalência entre as soluções encontradas.



*Figura 12: Adaptação de Casos por Transformação*

Ao contrário da *Analogia Transformacional*, a *Analogia Derivacional* [Ca86] toma em atenção a forma de resolução de problema. Tudo passa por se reconstruírem os mais relevantes aspectos do processo de resolução do problema, com base em conhecimento de casos passados, e aplicá-los ao contexto actual.

Para esta reconstrução ser possível, é necessário que o caso recuperado contenha (normalmente na justificação), informação do tipo (Figura 13):

- sobre a resolução do problema (eg. árvores de prova, objectivos a atingir,...);
- das decisões tomadas com vista à obtenção de uma solução (eg. alternativas consideradas e rejeitadas, operadores usados, etc.);
- dos apontadores para o conhecimento usado na solução do problema;
- qualquer outro tipo de informação do senso comum que se considere relevante ou pertinente para a reconstrução do problema.

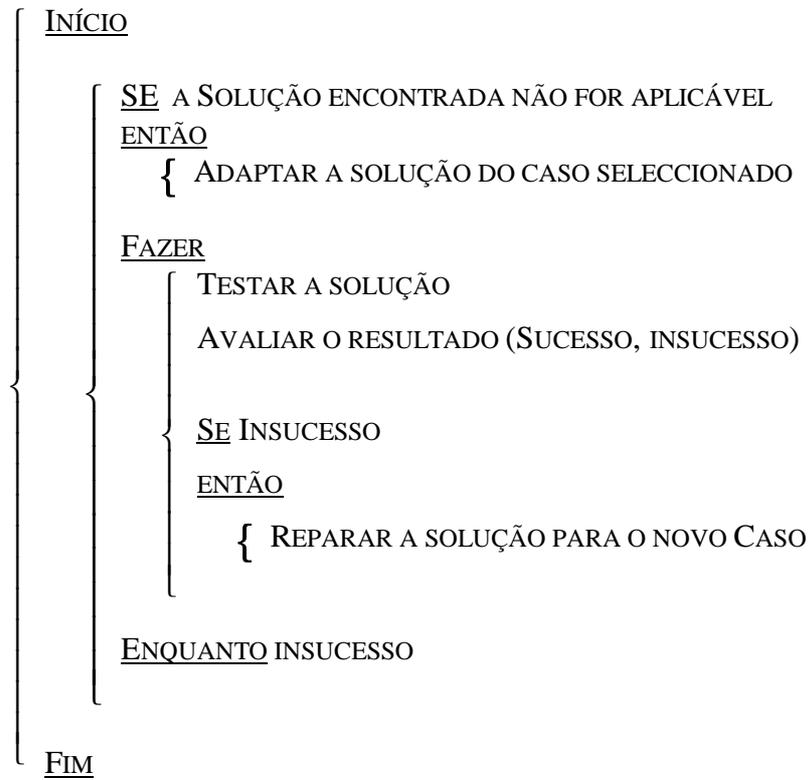
**ALGORITMO REUTILIZAÇÃO DE CASOS**

Figura 13: Algoritmo de Reutilização de Casos

### 3.5 Aprendizagem

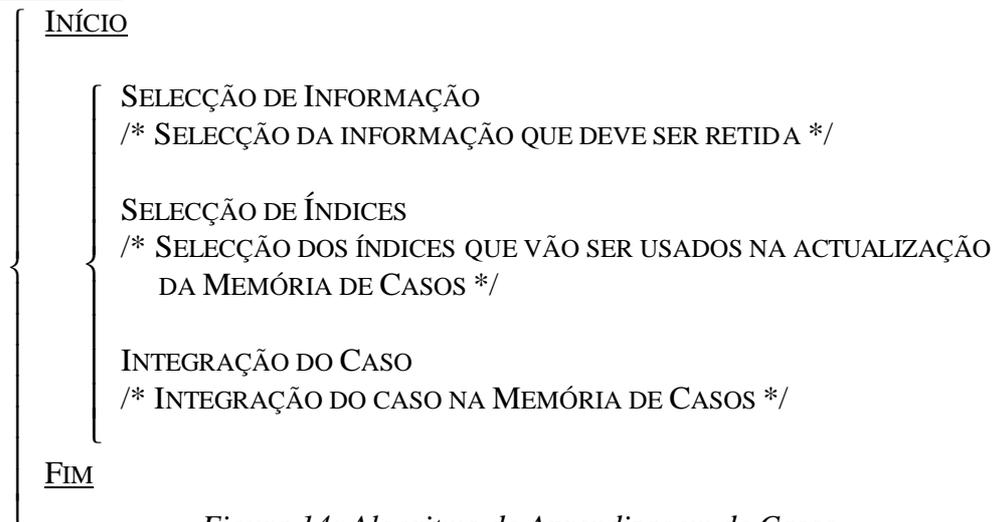
O objectivo da aprendizagem está em adquirir, em incorporar conhecimento de experiências passadas na memória de casos, quer através da construção de um novo caso, quer modificando a base de conhecimento de um caso já existente.

O processo de aprendizagem passa pela selecção da informação a respeito do caso recuperado que deve ser retida, de que forma deve ser retida, assim como de que forma se integrará o novo caso na memória de casos. O processo de aprendizagem passa então a ser descrito nos termos (Figura 14):

- *selecção* de informação;

- *selecção* de índices;
- *integração* do caso.

#### ALGORITMO APRENDIZAGEM EM RBC



*Figura 14: Algoritmo de Aprendizagem de Casos*

### **Seleccção de Informação**

O repositório de casos é sempre actualizado seja qual for a forma de resolução do problema. Por outro lado há que dar resposta a situações do género:

- se o problema foi resolvido com base num caso passado, então poder-se-á passar à generalização dessa situação ou mesmo criar um novo caso;
- se o caso foi resolvido com base em outros métodos (não teve como ponto de partida um caso passado), ou foi resolvido com base em intervenções do utilizador, então um novo caso deve ser adicionado à memória de casos.

## Seleção de Índices

Para a selecção dos índices que vão ser usados na actualização da memória de casos, usa-se um conjunto de regras, regras de indexação, que operam na forma:

- tratando os descritores do novo caso como índices;
- escolhendo os descritores mais relevantes;
- utilizando os descritores que distinguem os casos.

## Integração do Caso

Se a memória de casos é constituída por uma estrutura hierarquizada, então os índices são ajustados e a estrutura é actualizada, caso contrário há apenas que acrescentar o novo caso ao repositório de casos.

## 4 Conclusões

O *Raciocínio Baseado em Casos (RBC)* é apresentado como sendo um processo susceptível de utilizar conhecimento duma experiência passada, de um problema ou de um caso similar, para resolver um novo problema. O *RBC* dá corpo a uma forma de aprendizagem que se quer incremental e sustentada.

Algumas observações tornam-se neste momento pertinentes:

- um sistema de *RBC* é necessário quando o critério(s) de similaridade entre tuplos ou relações, não for(em) quantificável(eis) a partir de interrogações à base de dados. Uma base de dados pode apenas ser usada para a recuperação de casos se as características de similaridade entre estes forem objectiva e claramente definidas;
- o(s) critério(s) de similaridade deve(m) ser claramente equacionado(s).

Que características devem ser usadas para determinar se um dado problema é ou não equacionável em termos de um ou mais casos do passado? Poderão estas características ser claramente identificadas e quantificadas? Será que tudo isto depende do contexto em que o problema é colocado e da experiência do utilizador?;

- se as tradicionais formas de representação de conhecimento não são apropriadas, então um sistema baseado em casos deve ser considerado (eg. partes do Direito podem ser expressas em termos de regras; porém em termos de jurisprudência, o processo já não é tão linear);
- este tipo de sistema é um bom candidato para domínios do conhecimento nos quais este último é adquirido através de experiências passadas; i.e., o sistema ajusta-se a domínios ricos em experiências, mas pobres em conhecimento (eg. na gestão empresarial);
- este tipo de sistema é útil em situações em que a informação disponível é incompleta.

## 5 Referências

- [Aa93] Aamodt A.,  
A Case-Based Answer to some Problems of Knowledge-Based Systems, in Technical Report, University of Trondheim, Norway, 1993.
- [Aa94] Aamodt A.,  
*Explanation-Driven Case-Based Reasoning*, in Technical Report, University of Trondheim, Norway, 1994.
- [AaPl94] Aamodt A., Plaza E.,  
*Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches*, in AI Communications, Vol. 7, Nº 1, pages 39-59, 1994.

- [Al+95] Althoff K., Auriol E., Barletta R., Manago M.,  
*A Review of Industrial Case-Based Reasoning Tools*, in AI Perspectives Report Series, 1995.
- [AlWe91] Althoff K., Web S.,  
*Case-Based Reasoning and Expert System Development*, in Proceedings of The First Joint Workshop in Contemporary Knowledge Engineering and Cognition, Schmalhofer F., Strube G., Wetter T. (eds.), Kaiserlautern, Germany. Lectures Notes in Artificial Intelligence n° 622, pages 146-158, 1991.
- [ArPl94] Armengol E., Plaza E.,  
*Integrating Induction in Case-Based Reasoner*, in Proceedings of The Second European Workshop on Case-Based Reasoning, EWCBR-95, pages 243-251, 1994.
- [Aur+95] Auriol E., Wess S., Manago M., Althoff H., Traphöner R.,  
*INRECA: A Seamlessly Integrated System based on Inductive Inference and Case-Based Reasoning*, in Proceedings of The First International Conference on Case-Based Reasoning, ICCBR-95, Manuela Veloso and Agnar Aamodt (eds), pages 371-380, Portugal, 1995.
- [BaPoWi90] Bareiss E. R., Porter B., Wier C.,  
*PROTOS: An Exemplar-Based Learning Apprentice*, in Machine Learning - An Artificial Intelligence Approach, Vol. III, Kodratoff Y., Michalski R. (eds.), Morgan Kaufman, pages 112-127, 1990.
- [Ca83] Carbonell J. G.,  
*Learning by Analogy: Formulating and Generalizing Plans from Past Experience*, in Machine Learning - An Artificial Intelligence Approach, Michalski R., Carbonell J. G., Mitchell T. M. (eds.), Tioga, pages 137-161, 1983.
- [Ca86] Carbonell J. G.,  
*Derivational Analogy: A Theory of Reconstructive Problem Solving and Expertise Acquisition*, in Machine Learning - An Artificial Intelligence Approach, Vol. II, Michalski R., Carbonell J. G., Mitchell T. M. (eds.), Morgan Kaufman, pages 371-392, 1986.
- [Ca90] Carbonell J. G.,  
*Machine Learning Paradigms and Methods*, in MIT Press, London, 1990.

- [Ha85] Hayes-Roth F.,  
*Rule-Based Systems*, in Communications of The ACM, Vol. 28, nº9, 1985.
- [klCa88] Klein G., Calderwood R.,  
*How Do People Use Analogues To Make Decisions?*, in Proceedings of The DARPA Case-Based Reasoning Workshop, Janet Kolodner (ed), pages 209-223, Florida, 1988.
- [Ko88] Kolodner J.,  
*Extending Problem Solver Capabilities Through Case-Based Inference*, in Proceedings of The DARPA Case-Based Reasoning Workshop, Janet Kolodner (ed), pages 21-30, Florida, 1988.
- [Ko93] Kolodner J.,  
*Case-Based Reasoning*, in Morgan Kaufmann, 1993.
- [Le95] Leake D.,  
Combining Rules and Cases to Learn Case Adaptation, in Technical Report, Indiana University, 1995.
- [Ló92] López B.,  
*Reactive Planning through the Integration of a Case-Based System and a Rule-Based System*, in Prospects for Artificial Intelligence, Sloman A., et al. (eds), IOS Press, pages 189-198, 1992.
- [Ne+95] Neves J., Machado J., Novais P., Abelha A.,  
*Integrating Case-Based and Rule-Based Reasoning - A Logic Programming View*, in Research Report, Universidade do Minho, Portugal, 1995.
- [NeNo98] Neves J., Novais P.,  
*Minimal Models and Logic Consequence in Case-Based Reasoning*, in Research Report, Universidade do Minho, Portugal, 1998.
- [No96] Novais P.,  
*Sistema de Diagnóstico Multiagente*, Dissertação para o Mestrado, Universidade do Minho, Braga, Portugal, 1996.
- [Pl95] Plaza E.,  
*Cases as Terms: A Feature Term Approach to the Representation of Cases*, in Proceedings of The First International Conference on Case-Based Reasoning, ICCBR-95, Manuela Veloso and Agnar Aamodt (eds), pages 265-276, Portugal, 1995.

- [PlArc93] Plaza E., Arcos E.,  
*Flexible Integration of Multiple Learning Methods into a Problem Solving Architecture*, in Technical Report, Institut d'Investigació en Intel·ligència Artificial, Spain, 1993.
- [Ri95] Richter M.,  
*The Knowledge Contained in Similarity Measures*, in invited talk of The First International Conference on Case-Based Reasoning, Portugal, 1995.
- [RiWe91] Richter M., Wess S.,  
*Similarity, Uncertainty and Case-Based Reasoning in PATDEX*, in Technical Report, University of Kaiserslauten, 1991.
- [RisSk89] Rissland E., Skalak D.,  
*Combining Case-Based and Rule-Based Reasoning: A Heuristic Approach*, in Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, Sridharan (ed), USA, 1989.
- [Sc82] Schank R.,  
*Dynamic Memory: A Theory of Reminding and Learning in Computers and People*, in Cambridge University Press, UK, 1982.
- [Sh88] Shinn H.,  
*Abstractional Analogy: a Model of Analogical Reasoning*, in Proceedings of The DARPA Case-Based Reasoning Workshop, Janet Kolodner (ed), pages 370-387, Florida, 1988.
- [Sl91] Slade S.,  
*Case-Based Reasoning: A Research Paradigm*, in AI Magazine, 1991.
- [St94] Stottler R.,  
*CBR for Cost and Sales Prediction*, in AI Expert, pages 25-33, August 1994.
- [StWa88] Stanfill C., Waltz D.,  
*The Memory-Based Reasoning Paradigm*, in Proceedings of The DARPA Case-Based Reasoning Workshop, Janet Kolodner (ed), pages 414-424, Florida, 1988.
- [Ve94] Veloso M.,  
*Planning and Learning by Analogical Reasoning*, in Lectures Notes in Artificial Intelligence, vol. 886, Springer-Verlag, 1994.

- [Ve+94] Veloso M., Carbonell J., Pérez A., Borrajo D., Fink E., Blythe J., *Integrating Planning and Learning: The PRODIGY Architecture*, in Technical Report, Carnegie Mellon University, 1994.

## 6 Anexo

Para além da bibliografia dada, outros elementos de trabalho podem ser obtidos via a *World Wide Web (WWW)*, nomeadamente:

- Listas de correio  
AI-CBR: aqui discutem-se muitos das questões de RBC ainda em aberto. Para se juntar à lista envie uma mensagem para *mailbase@mailbase.ac.uk*, com join ai-cbr *nome*, como corpo.  
CBR-MED: Para aqueles que se interessam por RBC na área da medicina. Para se juntar a lista envie uma mensagem para *listproc@cs.uchicago.edu*, com subscribe CBR-MED *nome*, como corpo.
- *CBR-Newsletter*  
Jornal electrónico de RBC, que pode ser acedido em:  
<http://wwwagr.informatik.uni-kl.de/~lsa/CBR/cbrwNewsletter.html>.
- Repositórios na *WWW* de informações sobre RBC:  
<http://www.aic.nrl.navy.mil/~aha/research/case-based-reasoning.html>.  
<http://wwwagr.informatik.uni-kl.de/~lsa/CBR/CBR-Homepage.html>.  
<http://www.salford.ac.uk/survey/staff/IWatson/cbr01.html>.

