

Les connaissances : formaliser, raisonner, apprendre

Alain Mille*
Université Claude Bernard
Laboratoire d'ingénierie des systèmes d'information
Équipe Cognition et Expérience
EDIIS - DEA ECD
Module de Base

14 janvier 2001

Table des matières

1	Introduction	3
1.1	Intelligence Artificielle / Intelligence	3
1.2	Connaissances et compétences	3
2	Représenter les connaissances pour « raisonner logiquement »	5
2.1	Systèmes formels	5
2.1.1	Définition	5
2.1.2	Exemple	5
2.1.3	Deuxième exemple	6
2.1.4	Moralité	6
2.1.5	Interprétation	7
2.1.6	Synthèse	7
2.2	Rappels de logique	8
2.2.1	Calcul des propositions	8
2.2.2	Calcul des prédicats	16
2.2.3	Représentation des connaissances par objets	19
2.3	Un système à base de connaissances	22
2.3.1	Un système d'aide au voyageur	22
2.3.2	Déroulement	23
2.3.3	Autre exemple	25
2.3.4	Généralisations	26
2.3.5	Synthèse	28
2.3.6	Programmation par contraintes	28
3	Connaissances : expertise et expérience	33
3.1	Nouvelle génération de systèmes informatiques?	33
3.2	Tentative de définitions pour les concepts d'expertise et d'expérience	34

*Ce document profite largement de supports précédemment réalisés par Jean-Marc Fouet, disparu prématurément qui avait préparé ce cours en 2000. Je remercie aussi André Michel à qui j'ai emprunté sa façon de présenter la logique.

4	L'acquisition des connaissances : charnière du passage de l'expérience à l'expertise ou comment ne pas oublier l'expérience	38
4.1	Les méthodes, techniques et outils de la littérature : une approche pratiquement toujours tournée vers l'expertise	38
4.2	Synthèse de notre expérience de terrain : deux projets impliquant l'acquisition de connaissances	40
4.2.1	Supervision en qualité d'une production vaccinale : système expert associée à des cartes de contrôle multivariées (Mérieux 93)	40
4.2.2	Programmation de fours à micro-ondes guidées pour la minéralisation : système d'aide à la programmation basée sur la réutilisation directe de l'expérience (Prolabo 96)	44
4.3	Apprendre : associer expertise et expérience dans un réseau d'explications mutuelles?	45
5	État de l'art sur les recherches en raisonnement à partir de cas	48
5.1	Mise en contexte de la recherche	48
5.2	Le RàPC : Tendances de la recherche.	49
5.2.1	Principales tendances au niveau international	49
5.2.2	La recherche sur le raisonnement à partir de cas en France	53
5.3	La recherche dans le domaine RàPC par rapport aux directions stratégiques de la recherche en Intelligence Artificielle	56
5.4	Synthèse concernant les axes à développer en raisonnement à partir de cas	59

1 Introduction

1.1 Intelligence Artificielle / Intelligence

« L'Intelligence » est associée intuitivement à la capacité à résoudre des problèmes. C'est à ce titre que le jeu d'échecs, par exemple, a été particulièrement utilisé pour essayer de caractériser cette « intelligence » dont fait preuve l'homme quand il détermine son coup. En effet, si l'on imagine l'écriture d'un programme qui de manière systématique examine toutes les solutions ouvertes par les coups légaux à chaque nouvel état du jeu, on arrive à une complexité de calcul impressionnante incompatible avec sa mise en oeuvre par un calculateur quelconque (il faudrait qu'il soit plus de 1 million de fois plus puissant que le plus puissant des calculateurs actuels). C'est une sorte de démonstration que l'homme fait « autrement » et démontre ainsi son « intelligence ». Vous trouverez dans les documents préparés par Jean-Marc Fouet (<http://www710.univ-lyon1.fr/~fouet/DEA/index.html>) une illustration de ces problèmes de complexité. Il introduit la notion d'heuristique permettant de couper drastiquement la complexité par la « connaissance » de certaines propriétés qui simplifient très efficacement le problème posé. Ce recours à des heuristiques permet de revenir à des résolutions combinatoires réalisables car explorant un espace fini (et petit) de solutions. Les systèmes à base de connaissances sont fondés sur ce principe d'exploiter la connaissance de certaines propriétés pour résoudre simplement des problèmes complexes (en simplifiant le problème!). C'est ainsi qu'un système « intelligent » est supposé exploiter des connaissances pour fonder son comportement ¹.

Ces systèmes à base de connaissances démontrent donc des compétences en faisant reposer leur fonctionnement sur la mise en oeuvre d'un raisonnement s'appuyant sur ces connaissances. Ces notions de connaissances et compétences seront donc omniprésentes dans l'étude du champ de l'intelligence artificielle. Différencier connaissances et compétences permet de couvrir la plus grande part du domaine de l'intelligence artificielle.

1.2 Connaissances et compétences

Il n'existe naturellement pas de définition canonique des notions de connaissances et compétences et ce sont plutôt des considérations sur ces notions que nous allons tenter de rendre claires pour situer les différentes acceptions de ces termes que l'on retrouvera classiquement dans le domaine de l'intelligence artificielle.

Les connaissances sont supposées explicites, c'est-à-dire qu'il doit exister une façon de les rendre explicites. La capacité à « conceptualiser » est alors intimement liée à la notion de connaissances. Les connaissances sont explicitées pour être communiquées, partagées, combinées, etc. Le coeur historique de l'Intelligence Artificielle [Crevier, 1993], [Kirsh, 1991] est constitué de l'étude de cette conceptualisation du monde destinée à servir de connaissances exploitables par les mécanismes de raisonnement des systèmes. De plus, l'hypothèse est faite qu'il est possible de mener l'étude de ces connaissances et des mécanismes de raisonnement en faisant abstraction des sous-systèmes « sensitifs » et des sous-systèmes « moteurs » des systèmes « intelligents ». Comme la conceptualisation suppose une terminologie nommant et organisant ces concepts, l'hypothèse a été faite qu'il était possible de décrire les différents états traversés pendant le raisonnement sous la forme d'un langage logico-mathématique utilisant un vocabulaire connu (l'anglais pour l'essentiel). Enfin, il est souvent admis que l'on peut étudier séparément la cognition du proces-

1. Nous verrons qu'un comportement peut sembler parfaitement « intelligent » sans reposer sur l'exploitation de connaissances explicites.

sus d'apprentissage de concepts, du développement psychologique, et des changements de type évolutionnaires. Ces approches relèvent toutes d'une manière ou d'une autre à une approche « logicienne » qui a fondé les techniques de l'intelligence artificielle [Newell, 1990]. Nous verrons toutefois qu'il s'agit maintenant d'un enjeu majeur pour la recherche d'intégrer cette approche avec des approches plus fondées sur l'émergence de compétences et de connaissances à partir d'éléments beaucoup plus liés aux systèmes « sensitifs » et « moteurs ». Les sciences cognitives sont au coeur du débat.

Les compétences, quant à elles, se constatent, sans donner forcément lieu à une explicitation des connaissances mises en oeuvre. Les compétences peuvent être le résultat de l'exploitation de connaissances sur la tâche à réaliser et sur le contexte de cette tâche par un « moteur de raisonnement » mais il est parfaitement possible de faire preuve de compétence sans être capable d'explicitier cette compétence sous la forme de connaissances, ni même de disposer d'une représentation interne de ces connaissances [Brooks, 1991]. La manière dont ces compétences peuvent être acquises et reproduites fait l'objet de recherches importantes. La conceptualisation serait alors une tâche particulière liée à la nécessité de transmettre « noir sur blanc » des connaissances enfouies dans la « matière grise »².

2. J'emprunte ici à Jean-Marc Fouet la métaphore du passage de la « matière grise » au « noir sur blanc ».

2 Représenter les connaissances pour « raisonner logiquement »

2.1 Systèmes formels

Pour construire une langue (par exemple le français), on a besoin de 4 choses :

- un alphabet (a, b, ..., z, blanc, virgule, parenthèse ouvrante, ...),
- un procédé de formation des mots, qui est la concaténation,
- un dictionnaire, qui permet de savoir que "chat" est français, alors que "cat" ne l'est pas,
- des règles de grammaire, qui permettent de savoir que "chattes" est français, alors qu'il n'est pas dans le dictionnaire.

2.1.1 Définition

Pour construire un système formel, nous aurons besoin de 4 choses analogues :

- un alphabet, ensemble de symboles pas nécessairement réduit à des caractères,
- un procédé de formation des expressions, pas nécessairement la concaténation,
- un ensemble d'axiomes, c'est-à-dire d'expressions obéissant aux deux premiers points ci-dessus, et dont on décide arbitrairement qu'ils appartiennent au système,
- des règles de dérivation qui, à partir des axiomes, permettent de produire des théorèmes (c'est-à-dire des expressions appartenant au système), et peuvent ensuite s'appliquer aux théorèmes pour en produire d'autres.

2.1.2 Exemple

Considérons le système "peu" :

- alphabet = l'ensemble des trois symboles "p" , "e" , et "u"
- p.f.e. = concaténation
- axiome = upueuu
- règles :
 - R1 : si une expression de la forme AeB est un théorème (où "A" désigne n'importe quelle suite de "u", de "p", et de "e", et B de même), alors l'expression uAeBu est aussi un théorème.
 - R2 : si une expression de la forme AeB est un théorème, alors l'expression AueuB est aussi un théorème.

Questions :

- Q1 = uupuueuuu est-il un théorème?
- Q2 = upuueuuu ?
- Q3 = upupueuuu ?

Réponses :

- Q1, oui.
- Q2, non : il y a un nombre impair de "u", ce qui n'est pas possible.
- Q3, non : il y a deux "p".

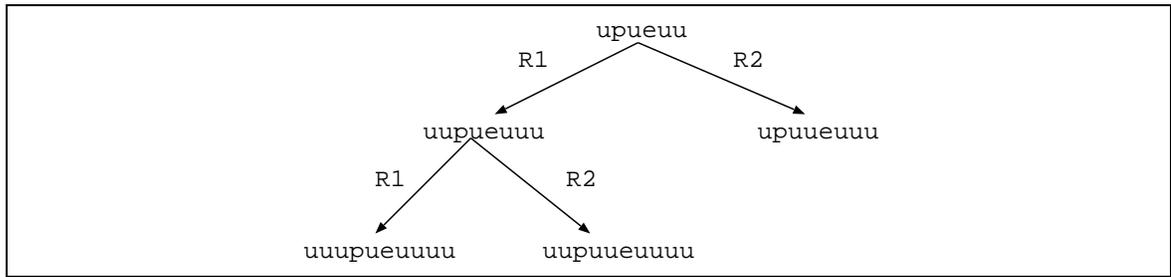


FIG. 1 – Preuve? On développe l'arbre

2.1.3 Deuxième exemple

Considérons le système :

- alphabet = l'ensemble des trois symboles "plus ", "egal ", et "un ",
- p.f.e. = concaténation,
- axiome = un plus un egal un un.
- règles :
 - R1 : si une expression de la forme A egal B est un théorème (où "A" désigne n'importe quelle suite de "un ", de "plus ", et de "egal ", et B de même), alors l'expression un A egal B un est aussi un théorème.
 - R2 : si une expression de la forme A egal B est un théorème, alors l'expression A un egal un B est aussi un théorème.

Questions :

- Q1 = un un plus un un egal un un un un est-il un théorème?
- Q2 = un plus un un egal un un un un ?
- Q3 = un plus un plus un egal un un un ?

Réponses : oui, non, et non, car les deux systèmes sont isomorphes.

Si nous remplaçons "plus " par "+", "egal " par "=", et "un " par "1", l'axiome s'écrit : $1 + 1 = 2$, en base 1 (en chiffres romains, concaténation = addition).

R1 se lit : si $A = B$, alors $1 + A = B + 1$

R2 : si $A = B$, alors $A + 1 = 1 + B$

Q1 : $2 + 2 = 4$ est un théorème

Q2 : $1 + 2 = 4$ est un non-théorème

Q3 : $1 + 1 + 1...$ est-un non-théorème !

2.1.4 Moralité

Pour démontrer qu'une expression est un théorème, nous avons une procédure (développer l'arbre comme illustré en figure 1) infaillible, qui répond en $2n/2$, n étant la longueur de la chaîne, donc en temps fini.

Par contre, pour démontrer qu'une expression est un non-théorème, il nous faut "tricher". Dans les deux exemples ci-dessus, nous avons fait de l'arithmétique, que faudrait-il faire sur un autre exemple?

Donc les notions de "être un théorème" et de "être un non-théorème" ne sont pas symétriques. Notre système "peu" est semi-décidable : nous avons une procédure qui répond "oui" en temps fini, mais pas de procédure pour répondre "non".

2.1.5 Interprétation

Ce que nous venons de manipuler n'est pas un système formel : le symbole "2" ne fait pas partie de l'alphabet. Ce que nous venons de manipuler est l'interprétation du système formel. "1+1+1=3", dont l'interprétation est « vrai », bien que ce soit un non-théorème.

Une interprétation d'un S.F. n'est en général pas équivalente au S.F., c'est-à-dire que les notions de "être un théorème" et "être un non-théorème" d'une part, et de "être vrai" et "être faux" d'autre part, peuvent ne pas se recouvrir.

Cependant, il ne faut pas rejeter l'interprétation : souvenez-vous, si vous avez étudié un peu de Géométrie, la première fois qu'on vous a demandé "démontrer que le triangle ABC est isocèle". Vous avez soigneusement mesuré les deux cotés, et vous avez dit : "Oui, les deux cotés sont égaux". Et l'enseignant vous a répondu : "NON! Ce n'est pas parce que c'est vrai sur la figure que c'est un théorème".

Mais vous n'avez pas pour autant cessé de faire des figures pour résoudre les problèmes, parce qu'elles guident le raisonnement. Nous y reviendrons.

2.1.6 Synthèse

Considérons un système formel quelconque, et l'ensemble (qui peut être infini) des expressions qu'on peut engendrer dans ce système en appliquant le procédé de formation des expressions.

Nous pouvons partitionner (voir figure 2) cet ensemble en :

- les expressions qui sont des théorèmes,
- et celles qui sont des non-théorèmes.

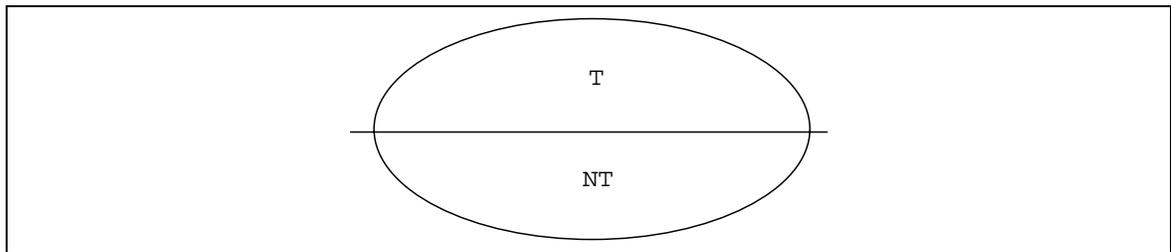


FIG. 2 – *Partition Théorème/Non Théorème*

Nous pouvons également partitionner (voir figure 3) cet ensemble en :

- ce qu'une procédure peut démontrer en temps fini,
- et ce qu'elle ne peut pas.

Si la partie ND est vide, le système est dit décidable. Si la partie TND est vide, il est semi-décidable. Si aucune partie n'est vide, il est indécidable.

Nous pouvons par ailleurs partitionner (voir figure 4) l'ensemble en :

- les expressions dont l'interprétation est "vrai",
- et celles dont l'interprétation est "faux".

Si la partie TF n'est pas vide, nous dirons que notre interprétation est sans intérêt. On montre en revanche que la partie NTV n'est, en général, pas vide.

Lorsqu'une proposition "manifestement vraie" ne peut être démontrée, on est donc en droit de se demander dans laquelle de ces zones on se trouve. Un cas célèbre était la conjecture de Fermat, qui a tenu les mathématiciens en échec pendant trois siècles : il n'existe pas d'entiers positifs x , y et z , et d'entier n supérieur à 2, tels que $x^n + y^n = z^n$.

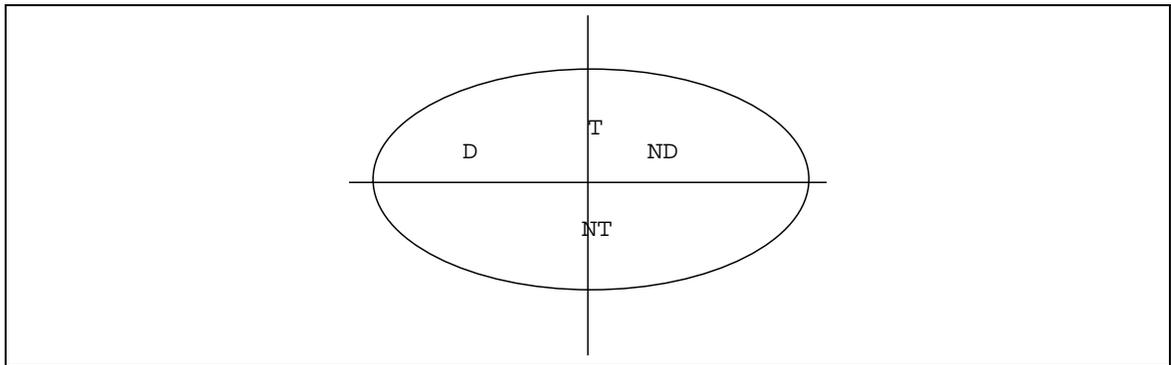


FIG. 3 – Partition Théorème/Non Théorème - Décidable/Non Décidable

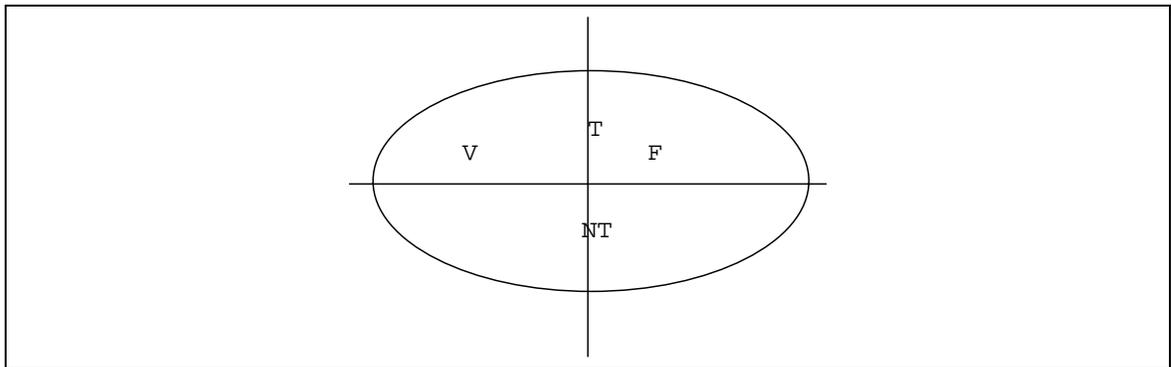


FIG. 4 – Partition Théorème/Non Théorème - Vrai/Faux

La partie suivante revient plus précisément sur les bases de la logique formelle qui est utilisée pour exprimer les mécanismes d'inférence ou encore de raisonnement qui caractériseraient un processus « intelligent ».

2.2 Rappels de logique

Cette introduction a pour but de rappeler les bases de la logique qui sous-tend les mécanismes fondamentaux de l'Intelligence Artificielle, comme déduction, abduction, etc

2.2.1 Calcul des propositions

Une *proposition* est un énoncé déclaratif (ou assertion) possédant par définition une seule *valeur de vérité* : *V* (vraie) ou *F* (fausse).

En langage naturel une proposition est en général une affirmation comme :

- la pluie mouille,
- il pleut,
- je suis dehors,
- je suis mouillé.

Ces affirmations sont des éléments d'information atomiques.

Les propositions sont représentées symboliquement par les éléments d'un ensemble dénombrable (un *alphabet*) appelés *littéraux*.

Une application *val* (valeur de vérité) de l'ensemble des propositions $\{X\}$ est définie sur l'ensemble $\{0, 1\}$:

$$\begin{aligned} \{X\} &\xrightarrow{val} \{0, 1\} \\ val(p) &\longrightarrow \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right\} \end{aligned} \quad (1)$$

Foncteurs de vérité unaires Il s'agit d'applications de l'ensemble $\{X\}$ des propositions possibles sur lui-même.

$$\begin{aligned} \{X\} &\xrightarrow{f} \{X\} \\ f(p) &\longrightarrow q \end{aligned} \quad (2)$$

A chacune de ces applications f correspond une application de $\{0, 1\}$ sur lui-même :

$$\begin{array}{ccc} p & \xrightarrow{f} & q \\ \downarrow val & & \downarrow val \\ \{0, 1\} & \xrightarrow{\varphi} & \{0, 1\} \end{array} \quad (3)$$

On a donc :

$$val(f(p)) = \varphi(val(p)) \quad (4)$$

Dans le cas de la logique bivalente 4 foncteurs unaires f sont possibles :

p	$val(p)$	$f_1(p)$	$f_2(p)$	$f_3(p)$	$f_4(p)$
V	1	1	1	0	0
F	0	1	0	1	0

FIG. 5 – Table des foncteurs unaires en logique bivalente

Dans le reste du texte, la relation d'équivalence \equiv sera remplacée par sa version "opérationnelle" \leftrightarrow .

Les foncteurs f_1 et f_4 sont les constantes 1 et 0 respectivement. Le foncteur f_3 est appelé *négarion de p* , il est dénoté par $\neg p$. f_2 est la *négarion de f_3* qui possède la même table de vérité que *val*. Cette double négarion définit la loi d'**idempotence** :

$$\neg(\neg p) \equiv p \quad (5)$$

Foncteurs de vérité binaires (interpropositionnels) Il s'agit d'applications f de l'ensemble produit $X \times X$ dans X , avec X l'ensemble des propositions possibles.

$$X \times X \xrightarrow{f} X \quad (6)$$

$$f(p, q) = r \quad (7)$$

Un tableau similaire à celui présenté en figure 5 peut être établi à partir du diagramme suivant :

$$\begin{array}{ccc} p \times q & \xrightarrow{f} & r \\ \text{val} \times \text{val} \downarrow & & \downarrow \text{val} \\ \{0, 1\} \times \{0, 1\} & \xrightarrow{\varphi} & \{0, 1\} \end{array} \quad (8)$$

avec :

$$\text{val}(f(p, q)) = \varphi(\text{val}(p), \text{val}(q)) \quad (9)$$

Ce tableau (figure 6) met en évidence les 16 foncteurs de vérité binaire. Une sémantique classique est précisée pour chacun des foncteurs.

					notation	sémantique
p	1	1	0	0		
q	1	0	1	0		
1	1	1	1	1		tautologie
2	1	1	1	0	\vee	ou inclusif
3	1	1	0	1	\Leftarrow	p si q
4	1	1	0	0		$f_4(p, q) = p$
5	1	0	1	1	\Rightarrow	p implique q
6	1	0	1	0		$f_6(p, q) = q$
7	1	0	0	1	\Leftrightarrow	p si et seulement si q
8	1	0	0	0	\wedge	p et q
9	0	1	1	1	$ $	NAND incompatibilité
10	0	1	1	0	\oplus	XOR alternative, ou exclusif
11	0	1	0	1		
12	0	1	0	0		
13	0	0	1	1		
14	0	0	1	0		
15	0	0	0	1	\downarrow	NOR - rejet
16	0	0	0	0		antilogie

FIG. 6 – Table de vérité (foncteurs binaires) $f(p, q) = r$

Les notations sont indicatives et varient selon les auteurs. La sémantique indiquée doit être considérée avec précaution. Le sens usuel des termes

“implique”, tautologie, antilogie, etc. sont souvent différents de la stricte interprétation en logique. Ainsi la proposition $[p \Rightarrow q]$ est toujours vraie sauf si $p = 1$ et $q = 0$, ce qui signifie seulement que le vrai (1) ne peut impliquer le faux (0).

Expressions Bien Formées (EBF) Les foncteurs binaires permettent de transformer des propositions simples en propositions complexes que l'on appelle le plus souvent des formules. Une formule sera libellée en majuscules. Toutes les formules possibles (combinant des littéraux de propositions simples, des opérateurs, des connecteurs, des parenthèses) ne sont pas forcément bien formées pour constituer à leur tour une expression trouvant sa valeur dans le système formel. Quatre règles doivent être respectées pour qu'une formule soit bien formée :

- règle 1 : un littéral de proposition simple est une EBF de même que les constantes logiques V (vrai) et F (faux),
- règle 2 : P étant une formule, si P est une EBF alors $\neg P$ est une EBF,
- règle 3 : P et Q étant deux formules, si P et Q sont des EBF alors $P \Rightarrow Q$ est une EBF.
- règle 4 : rien n'est une EBF sinon les suites de symboles produites aux règles 1 à 3 précédentes.

Ces règles permettent de distinguer :

- la fausseté d'une formule qui caractérise un énoncé ayant pour valeur de vérité F ,
- l'absence de signification qui caractérise un énoncé dénué de sens,

Par exemple, $[\neg(\Rightarrow pq)]$ n'est pas un énoncé faux mais un énoncé dénué de sens (n'est pas une EBF).

Calcul des propositions : évaluation des EBF Le calcul des propositions consiste à trouver la valeur de vérité d'une formule P composée de propositions atomiques $p_1 \dots p_i \dots p_n$.

Deux méthodes sont disponibles :

- la méthode des tables de vérité,
- la méthode de parcours d'arbre de solutions.

Si on considère qu'une tautologie est une formule dont les valeurs de vérité sont toutes vraies, alors une tautologie peut s'appeler une formule valide (on dit aussi qu'une formule valide constitue une loi logique).

La logique *formelle* peut alors être considérée comme la recherche de toutes les tautologies possibles.

Le chapitre suivant présente un certain nombre de lois logiques constituant la base de la logique formelle.

Principales lois logiques (tautologies) Les lois présentées dans la figure 7 sont à la base de nombreuses simplifications. L'absorption est une autre façon de simplifier les formules. Les différentes lois permettant d'autres simplifications (ou développements) avec les connecteurs \vee et \wedge sont présentées dans le tableau suivant (figure 8) :

$p \Leftrightarrow p$	loi d'identité
$p \Rightarrow p$	tautologie élémentaire
$p \Leftrightarrow (p \vee p)$ $p \Leftrightarrow (p \wedge p)$	lois d'idempotence des connecteurs \vee et \wedge
$\neg(p \wedge \neg p) \Leftrightarrow V$	loi de non contradiction deux formules p et $\neg p$ ne peuvent être toutes les deux vraies.
$p \vee \neg p \Leftrightarrow V$	loi du tiers exclu deux formules p et $\neg p$ ne peuvent toutes les deux fausses.
$\neg\neg p \Leftrightarrow p$	loi de la double négation

FIG. 7 – Lois élémentaires sur une proposition unique

$(p \vee q) \Leftrightarrow (q \vee p)$ $(p \wedge q) \Leftrightarrow (q \wedge p)$	Commutativité
$((p \vee q) \vee r) \Leftrightarrow (p \vee (q \vee r))$ $((p \wedge q) \wedge r) \Leftrightarrow (p \wedge (q \wedge r))$	Associativité
$(p \vee (q \wedge r)) \Leftrightarrow (p \vee q) \wedge (p \vee r)$ $(p \wedge (q \vee r)) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$	Double distributivité
$p \wedge (p \vee q) \Leftrightarrow p$ $p \vee (p \wedge q) \Leftrightarrow p$	Absorption

FIG. 8 – Lois de commutativité, associativité, double distributivité et absorption

Un certain nombre de lois permettent d'établir des dualités d'écriture de connecteurs en termes d'autres connecteurs. Ces lois sont connues sous le nom de lois de **DE MORGAN** (tableau de la figure 9).

Le tableau suivant (figure 10) présente quelques lois dites de l'implication.

Substitution et remplacement L'opérateur de substitution permet de formaliser la propagation d'une formule par le remplacement d'un littéral partout où il apparait dans une autre formule.

L'expression de la substitution s'écrit :

$S(Q, p, P)$ (10)	Substitution de Q à p dans P
-------------------	------------------------------------

Par exemple :

$\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$ $\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$ $(p \Rightarrow q) \Leftrightarrow (\neg p \vee q)$ $(p \Rightarrow q) \Leftrightarrow \neg(p \vee \neg q)$ $(p \Leftrightarrow q) \Leftrightarrow (p \Rightarrow q) \wedge (q \Rightarrow p)$ $(p \vee q) \Leftrightarrow (\neg p \Rightarrow q)$ $(p \wedge q) \Leftrightarrow \neg(p \Rightarrow \neg q)$	Lois de dualité ou lois DE MORGAN
--	-----------------------------------

FIG. 9 – Lois de DE MORGAN

$(p \Rightarrow q) \Leftrightarrow (\neg q \Rightarrow \neg p)$	Loi de contraposition
$((p \Rightarrow q) \wedge (q \Rightarrow r)) \Rightarrow (p \Rightarrow r)$	Loi de la transitivité et de l'implication
$((p \Rightarrow q) \wedge (p \Rightarrow \neg q)) \Rightarrow \neg p$	Loi de réduction à l'absurde
$(\neg p \Rightarrow p) \Rightarrow p$	Loi de Clavius
$(p \Rightarrow (q \Rightarrow p)) \Rightarrow p$	Loi de Pierce
$p \Rightarrow (q \Rightarrow p)$	<i>Verum sequitur ad quodlibet</i> , une proposition vraie est impliquée par n'importe quelle autre proposition.
$\neg p \Rightarrow (p \Rightarrow q)$	Ex falso sequitur quodlibet (à partir du faux on peut déduire n'importe quoi)
$((p \Rightarrow q) \wedge p) \Rightarrow q$	Modus ponens (<i>latin pono : poser ou affirmer</i>)
$((p \Rightarrow \neg q) \wedge q) \Rightarrow \neg p$	Modus tollens (<i>latin tollo : enlever ou nier</i>)
$(p \Rightarrow (q \Rightarrow r)) \Leftrightarrow ((p \wedge q) \Rightarrow r)$	Principe "d'import-export"

FIG. 10 – Lois d'implication

$$\begin{aligned}
 P &: p \Rightarrow (q \Rightarrow p) \\
 Q &: p \Rightarrow q \\
 S(Q, q, P) &= \text{Substituer } Q \text{ à toutes les occurrences de } q \text{ dans } P \\
 \text{On obtient } &p \Rightarrow ((p \Rightarrow q) \Rightarrow p)
 \end{aligned} \tag{11}$$

Théorème 2.1 Si P est une tautologie et Q une EBF alors $S(Q, q, P)$ est une tautologie.

Corollaire du théorème 2.1 Si R est une tautologie et P une formule constituante de R et si $(P \Rightarrow Q)$ est une tautologie, alors l'expression obtenue en remplaçant P par Q dans une ou plusieurs occurrences quelconques de P dans R est une tautologie.

La règle d'élimination C'est une tautologie permettant de simplifier grand nombre de formules. Cette règle (que le lecteur démontrera facilement) indique :

$$(P \vee Q) \wedge (\neg P \vee R) \Rightarrow Q \vee R \quad (P \text{ est "dissous"!) \tag{12}$$

Jeu de connecteurs fonctionnellement complet Les 16 connecteurs de la table de vérité décrite en figure 6 peuvent être tous représentés à l'aide des seuls connecteurs :

- conjonction \vee
- disjonction \wedge
- implication \Rightarrow

Ainsi, $p \Rightarrow q$ équivaut à $\neg \vee q$.

On appelle fonctionnellement complet le jeu de connecteurs qui permet d'obtenir tous les autres par définition.

En conséquence toute EBF peut être ramenée à une formule où ne figurent que les opérateurs \neg , \wedge , \vee , formant le concept des *formes normales d'une EBF*.

Formes normales d'une EBF

Définition 2.1 On appelle forme normale disjonctive distinguée (FNDD) d'une expression P une forme disjonctive équivalente à P (mêmes valeurs de vérité) et dont tous les membres disjonctifs contiennent **une et une seule fois chacun** des littéraux propositionnels figurant dans P .

Pour écrire la FNDD d'une expression donnée, il suffira d'énumérer chacun des cas où l'expression prend la valeur V (vraie).

Définition 2.2 On appelle forme normale conjonctive distinguée (FNCD) d'une expression P une forme conjonctive équivalente à P dont tous les facteurs contiennent une et une seule fois chacun des littéraux propositionnels figurant dans P , ces littéraux étant ou non affectés de négation.

Pour écrire la FNCD d'une expression, il suffira d'inventorier chacun des cas où l'expression P prend la valeur F (faux).

On notera qu'une FNCD est une conjonction de disjonctions élémentaires.

Remarque: Les disjonctions citées dans la définition sont appelées **clauses** et la FNCD est une **forme clauseale**.

Ces définitions servent à la résolution de calcul de propositions pour la démonstration automatique de tautologies.

Forme clauseale de KOWALSKI La forme clauseale de Kowalski est une forme abrégée de la notation clauseale précédente :

$$P_1, P_2, \dots, P_m \Leftarrow Q_1, Q_2, \dots, Q_n \quad (13)$$

Les virgules à **gauche** sont à lire **ou**,

Les virgules à **droite** sont à lire **et**,

La flèche orientée à gauche se lit **si**,

Les P_i et Q_j sont appelés **clauses**,

Les clauses P_i sont appelées **conclusions**,

Les clauses Q_i sont appelées **conditions**.

La forme généralisée introduit les valeurs F (faux) et V :

$$P_1, P_2, \dots, P_m, F \Leftarrow Q_1, Q_2, \dots, Q_n, V \quad (14)$$

Plusieurs cas particuliers sont distingués :

$m > 1$	les conclusions sont multiples, c'est le cas non déterministe.
$m \leq 1$	Cas particulier des clauses dites clauses de HORN du nom d'un chercheur américain (1951)
$m = 1$ $n > 0$	$P \Leftarrow Q_1, Q_2, \dots, Q_n$ clauses définies. Il existe une conclusion. C'est le cas déterministe.
$m = 1$ $n = 0$	$P \Leftarrow V$ clause définie inconditionnelle: c'est un fait ou une assertion ou encore un axiome .
$m = 0$ $n > 0$	$F \Leftarrow Q_1, Q_2, \dots, Q_n$ négation formelle de la conjonction des Q_j
$m = 0$ $n = 0$	$F \Leftarrow V$ une antilogie fondamentale

La preuve par l'absurde ou réfutation Prouver qu'une clause est vraie revient à prouver que la négation de cette clause est une contradiction (antilogie).

Ce type de preuve est bien traité dans le cas des clauses de HORN. La résolution est basée sur trois *schémas* appliqués de manière systématique.

Schéma 1

$$\begin{array}{l}
 P \Leftarrow Q, R \quad P \text{ si } Q \text{ et } R \\
 \text{et} \\
 F \Leftarrow P \quad P \text{ est } \textit{faux} \\
 \text{alors} \\
 \hline
 F \Leftarrow Q, R \quad (Q \text{ et } R) \text{ est faux, c'est à dire non vrais en même temps}
 \end{array}$$

Schéma 2

$$\begin{array}{l}
 P \Leftarrow V \quad P \text{ est vrai} \\
 \text{et} \\
 F \Leftarrow P, Q \quad (P \text{ et } Q) \text{ est faux (pas vrais en même temps)} \\
 \text{alors} \\
 \hline
 F \Leftarrow Q \quad Q \text{ est faux}
 \end{array}$$

Schéma 3

$$\begin{array}{l}
 P \Leftarrow V \quad P \text{ est vrai} \\
 \text{et} \\
 F \Leftarrow P \quad P \text{ est faux} \\
 \text{alors} \\
 \hline
 F \Leftarrow V \quad \textbf{contradiction}
 \end{array}$$

Schéma 4

En généralisant :

$$\begin{array}{l}
 P_1 \Leftarrow Q_1, Q_2, \dots, Q_n \\
 F \Leftarrow P_1, P_2, \dots, P_m \\
 \text{alors} \\
 \hline
 F \Leftarrow Q_1, Q_2, \dots, Q_n, P_1, P_2, \dots, P_m
 \end{array}$$

Les calculs ainsi présentés font apparaître les clauses opérantes au dessus de la ligne de points, et la clause résultante en dessous d'elle.

Le processus de démonstration est rarement linéaire. Les schémas sont appliqués dans l'ordre aux clauses. Les clauses résultantes sont ajoutées à la base de connaissances. En cas d'impasse (plus de schéma applicable), un retour en arrière est fait (backtracking), pour recommencer l'application des schémas aux autres clauses.

2.2.2 Calcul des prédicats

Le calcul des prédicats est une extension du calcul des propositions. Appelé également logique du premier ordre (LPO), ou logique des prédicats, il est à la base des langages de programmation logique comme PROLOG par exemple. La puissance d'expression du calcul des prédicats repose sur l'apparition des **variables**, **quantificateurs** et **fonctions**.

Termes Les constantes, les variables et les fonctions sont appelées **termes**.

- une constante est un sous-ensemble des symboles autorisés du domaine, comme - Jean, 23.56, $\sqrt{2}$, ... -³,
- une variable est un sous-ensemble de symboles autorisés du domaine qui prend des constantes comme valeurs⁴,
- une fonction est une application de l'ensemble des constantes dans lui-même, comme - Nom(Jean), SquareRoot(16), ... -

Les prédicats Un prédicat est l'application de l'ensemble des constantes dans l'ensemble (VRAI, FAUX)⁵. Par exemple, $EST_DANS(Boite, y)$, est un prédicat à deux variables. La première est instanciée par Boite, tandis que la seconde est libre. Une variable peut être instanciée par une liste de constantes. Les formules atomiques sont des combinaisons de prédicats avec leurs arguments. Les littéraux sont à la fois les formules atomiques et leur négation.

Quantificateurs Les quantificateurs sont ceux habituellement utilisés en mathématiques :

- \exists comme dans $[\exists x, OISEAU(x) \cap VERT(x)]$
- \forall comme dans $[\forall x, OISEAU(x) \rightarrow PLUME(x)]$

L'ordre des quantificateurs est très important pour l'évaluation d'une expression. Par exemple :

$$[\forall x, \exists y, CONNAIT(x, y)] \quad (15)$$

qui signifie: " Toute (personne) x connaît au moins une autre (personne) y", très différent de:

$$[\exists x, \forall y, CONNAIT(x, y)] \quad (16)$$

qui signifie: " Il existe une (personne) x qui connaît toutes les (personnes) y".

Les variables utilisées avec un quantificateur sont dites **liées** et les autres sont **libres**.

3. Les constantes "caractères" sont le plus souvent représentées par des chaînes de caractères commençant par une lettre majuscule.

4. Les noms de variables commencent le plus souvent par une lettre minuscule.

5. En langue naturelle, un prédicat est le plus souvent exprimé par un verbe.

Formules bien formées Les formules bien formées⁶ sont définies récursivement. Les littéraux sont des formules bien formées.

Notion de clause La démonstration de théorème utilise des expressions logiques mises sous forme clausale (sous forme de clauses). Une clause est une formule bien formée consistant en des disjonctions de littéraux. Un algorithme a été proposé par Winston pour construire des clauses à partir de formules logiques :

1. Eliminer les implications
2. Déplacer les négations dans les formules atomiques
3. Eliminer les quantificateurs existentiels
4. Renommer les variables si nécessaire
5. Déplacer les quantificateurs universels vers la gauche
6. Déplacer les disjonctions dans les littéraux
7. Eliminer les conjonctions
8. Renommer les variables, si nécessaire
9. Eliminer les quantificateurs universels

Reprenons un exemple souvent cité dans les ouvrages :

Etre un terrien implique trois choses :

1. le terrien vit sur la Terre
2. il n'existe pas de Martien qui soit parent avec un Terrien
3. il n'existe pas de non Terrien qui soit connu d'un Terrien

On peut reformuler cet énoncé de la manière suivant en formules logiques :

$$\begin{aligned} [\forall x, \text{ TERRIEN}(x) \rightarrow & [[\exists y, \text{ VIVRE_SUR}(x, y) \cap \text{ TERRE}(y)] \\ & [\neg \exists y, \text{ PARENT}(x, y) \cap \text{ MARTIEN}(y)] \\ & [\forall y, \neg \text{ TERRIEN}(y) \rightarrow \neg \text{ CONNU}(x, y)]]] \end{aligned} \quad (17)$$

Elimination des implications

$$\begin{aligned} [\forall x, \neg \text{ TERRIEN}(x) \cup & [[\exists y, \text{ VIVRE_SUR}(x, y) \cap \text{ TERRE}(y)] \\ & [\neg \exists y, \text{ PARENT}(x, y) \cap \text{ MARTIEN}(y)] \\ & [\forall y, \text{ TERRIEN}(y) \cup \neg \text{ CONNU}(x, y)]]] \end{aligned} \quad (18)$$

Déplacer les négations dans les formules atomiques

$$\begin{aligned} [\forall x, \neg \text{ TERRIEN}(x) \cup & [[\exists y, \text{ VIVRE_SUR}(x, y) \cap \text{ TERRE}(y)] \\ & [\forall y, \neg \text{ PARENT}(x, y) \cup \neg \text{ MARTIEN}(y)] \\ & [\forall y, \text{ TERRIEN}(y) \cup \neg \text{ CONNU}(x, y)]]] \end{aligned} \quad (19)$$

Elimination des quantificateurs existentiels Dans cette phase, les fonctions dites de **skolem** doivent être introduites. Une fonction de skolem sert à substituer une variable quantifiée existentiellement. Elle s'exprime en fonction de variables définies par les quantificateurs universels de l'expression dans laquelle la substitution doit être faite. Ici, la

6. Well-formed formula en anglais.

fonction skolem de la variable existentielle y sera notée $PLANETE(x)$, soit :

$$\begin{aligned} & [\forall x, \neg TERRIEN(x) \cup [[\exists y, VIVRE_SUR(x, PLANETE(x) \cap TERRE(PLANETE(x))] \\ & \quad [\forall y, \neg PARENT(x, y) \cup \neg MARTIEN(y)] \\ & \quad [\forall y, TERRIEN(y) \cup \neg CONNU(x, y)]]]] \end{aligned} \quad (20)$$

Renommer les variables, si nécessaire

$$\begin{aligned} & [\forall x, \neg TERRIEN(x) \\ & \quad \cup [[VIVRE_SUR(x, PLANETE(x) \cap TERRE(PLANETE(x))] \\ & \quad [\forall y, \neg PARENT(x, y) \cup \neg MARTIEN(y)] \\ & \quad [\forall z, TERRIEN(z) \cup \neg CONNU(x, z)]]]] \end{aligned} \quad (21)$$

Déplacer les quantificateurs universels vers la gauche

$$\begin{aligned} & [\forall x, \forall y, \forall z, \neg TERRIEN(x) \\ & \quad \cup [[VIVRE_SUR(x, PLANETE(x) \cap TERRE(PLANETE(x))] \\ & \quad [\neg PARENT(x, y) \cup \neg MARTIEN(y)] \\ & \quad [TERRIEN(z) \cup \neg CONNU(x, z)]]]] \end{aligned} \quad (22)$$

Déplacer les disjonctions dans les littéraux

$$\begin{aligned} & [\forall x, \forall y, \forall z \\ & \quad [\neg TERRIEN(x) \cup VIVRE_SUR(x, PLANETE(x))] \\ & \quad \cap [\neg TERRIEN(x) \cup TERRE(PLANETE(x))] \\ & \quad [\neg TERRIEN(x) \cup \neg PARENT(x, y) \cup \neg MARTIEN(y)] \\ & \quad [\neg TERRIEN(x) \cup TERRIEN(z) \cup \neg CONNU(x, z)]] \end{aligned} \quad (23)$$

Eliminer les conjonctions

$$\begin{aligned} & [\forall x, \neg TERRIEN(x) \cup [VIVRE_SUR(x, PLANETE(x)] \\ & \quad [\forall x, \neg TERRIEN(x) \cup TERRE(PLANETE(x))] \\ & \quad [\forall x, \forall y, \neg TERRIEN(x) \cup \neg PARENT(x, y) \cup \neg MARTIEN(y)] \\ & \quad [\forall x, \forall z, \neg TERRIEN(x) \cup TERRIEN(z) \cup \neg CONNU(x, z)]] \end{aligned} \quad (24)$$

Renommer les variables si nécessaire

$$\begin{aligned} & [\forall x, \neg TERRIEN(x) \cup [VIVRE_SUR(x, PLANETE(x)] \\ & \quad [\forall u, \neg TERRIEN(u) \cup TERRE(PLANETE(u))] \\ & \quad [\forall v, \forall y, \neg TERRIEN(v) \cup \neg PARENT(v, y) \cup \neg MARTIEN(y)] \\ & \quad [\forall w, \forall z, \neg TERRIEN(w) \cup TERRIEN(z) \cup \neg CONNU(w, z)]] \end{aligned} \quad (25)$$

Eliminer les quantificateurs universels

$$\begin{aligned} & \neg TERRIEN(x) \cup [VIVRE_SUR(x, PLANETE(x) \\ & \quad \neg TERRIEN(u) \cup TERRE(PLANETE(u)) \\ & \quad \neg TERRIEN(v) \cup \neg PARENT(v, y) \cup \neg MARTIEN(y) \\ & \quad \neg TERRIEN(w) \cup TERRIEN(z) \cup \neg CONNU(w, z) \end{aligned} \quad (26)$$

Ceci permet la démonstration de théorème.

Algorithme de résolution

1. Nier le théorème à prouver et ajouter le résultat dans la liste des axiomes
2. Mettre la liste des axiomes sous la forme de clauses
3. Jusqu'à ce que la liste de clauses soit vide (NIL) soit générée ou qu'il n'y ait plus de paires de clauses résolubles, chercher les clauses résolubles, les résoudre, et les ajouter à la liste des clauses
4. Si la clause vide est générée, annoncer que le théorème est vrai, s'il n'y a pas de clause résoluble, annoncer que le théorème est faux.

2.2.3 Représentation des connaissances par objets

Imaginer comment représenter les connaissances est une gageure. La modélisation de la mémoire humaine a inspiré les psychologues, qui cherchent à reproduire les fonctions et mécanismes de raisonnement pour les étudier. Les réseaux sémantiques sont issus de ces travaux et sont devenus depuis des mécanismes récupérés par la communauté informatique pour exprimer commodément ce qui "est su" en général : "le sens commun".

Graphes et réseaux sémantiques Les entités présentes dans un réseau sémantique sont respectivement des noeuds (représentant des objets) et des arcs (représentant des relations entre ces objets). Le lecteur s'aperçoit bien maintenant des racines des différentes méthodes d'analyse et conception "objet"!

Un graphe représentant un réseau sémantique pourrait être celui de la figure 11.

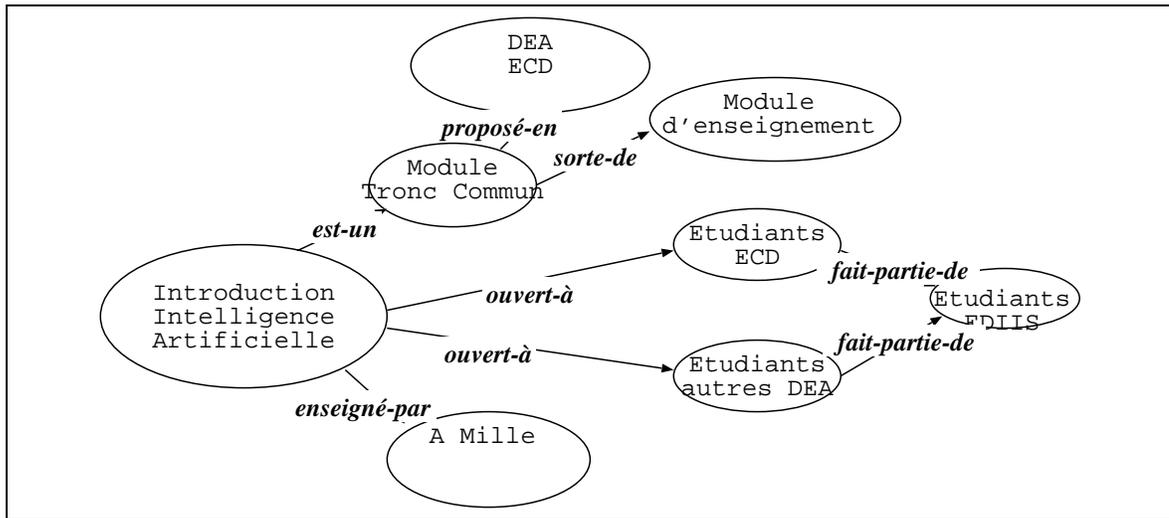


FIG. 11 – Exemple de réseau sémantique

Les réseaux sémantiques ont été utilisés pour l'explication (ou la compréhension) de phrases, comme :

Le professeur lit l'exercice

Le réseau sémantique correspondant est illustré dans la figure 12.

Cette phrase pourrait aussi s'exprimer sous la forme du prédicat :

[LIRE(PROFESSEUR,EXERCICE)]

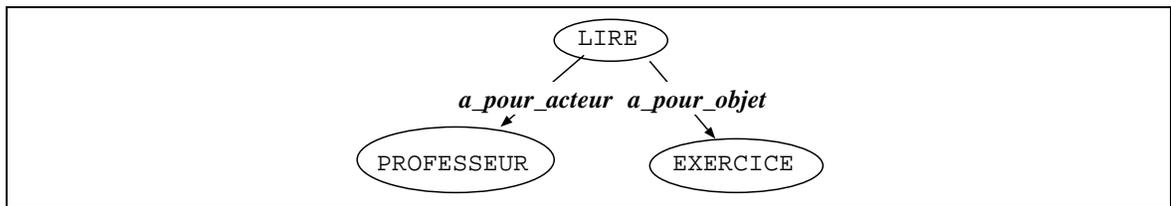


FIG. 12 – Réseau équivalent à la phrase

Plus généralement tout objet pourra être représenté avec ses relations avec les autres selon une structure de réseau sémantique particulière appelée structure d'objet.

Un objet est défini par un ensemble de propriétés. Ces propriétés sont classiquement appelées des attributs. Une classe est un ensemble d'objets qui partagent des caractères communs. L'exemple présenté ci-dessus présente le réseau sémantique correspondant à la

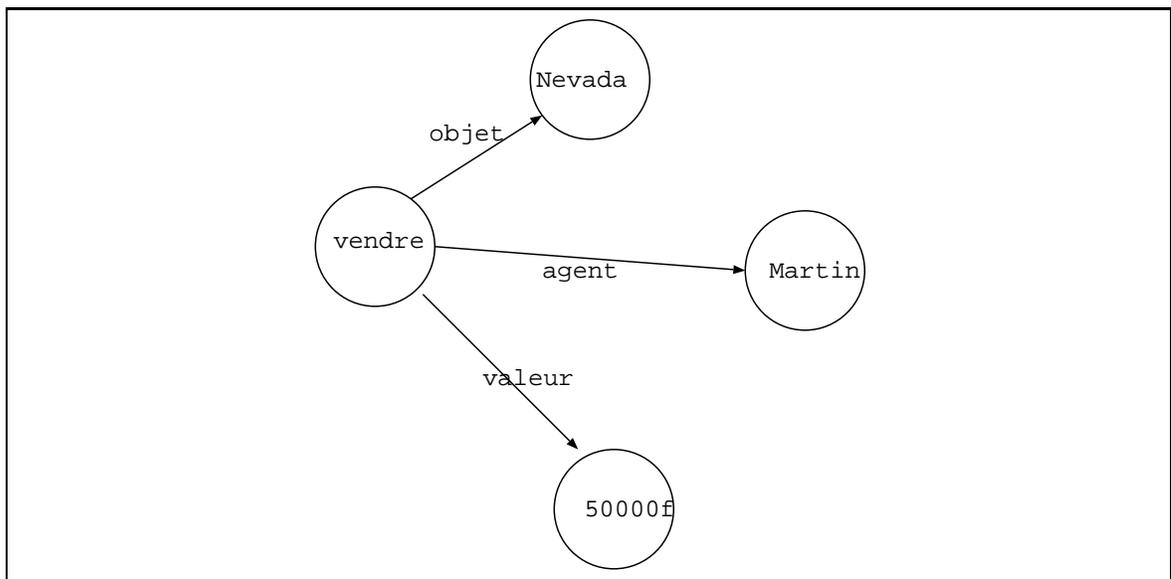


FIG. 13 – Exemple de réseau sémantique

phrase: **Martin vend une Nevada pour 50 000 francs.**

On peut faire une équivalence entre réseau sémantique et calcul de prédicat.

La représentation du réseau présenté en figure 13 peut s'exprimer sous la forme suivante :

$$Vendre(Martin, Nevada, 50000f) \quad (27)$$

Si on généralise la proposition, on l'exprime sous la forme d'un prédicat :

$$(\exists x, \exists y, \exists z)(Vendre(z, x, y) \& agent(z) \& objet(x) \& valeur(y)) \quad (28)$$

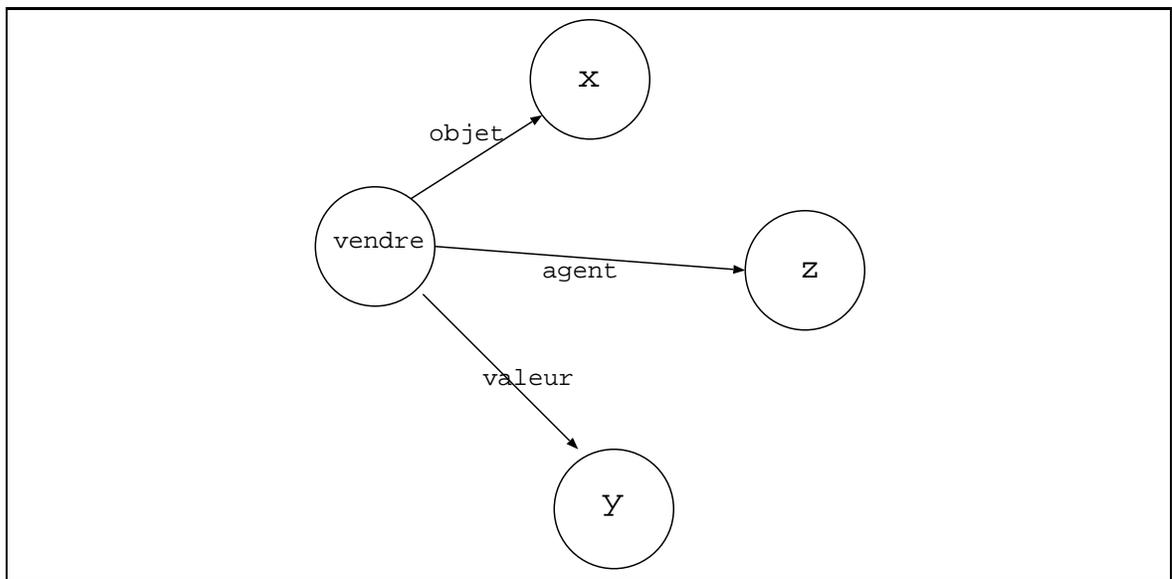


FIG. 14 - Réseau plus générale

2.3 Un système à base de connaissances

Ce paragraphe illustre de manière pratique le fonctionnement d'un moteur d'inférence exploitant la logique des propositions sur la base d'un exemple simple.

2.3.1 Un système d'aide au voyageur

alphabet	distance.<.2km distance.<.300km aller.à.pied prendre.le.train prendre.l'avion avoir.le.téléphone aller.à.l'agence téléphoner.à.l'agence acheter.un.billet durée.>.2.jours être.fonctionnaire () non (négation) \wedge (et, ou conjonction) \rightarrow (implique)
procédé de formation des expressions	expression \leftarrow symbole expression \leftarrow (expression) expression \leftarrow non expression expression \leftarrow expression1 \wedge expression2 expression \leftarrow expression1 \rightarrow expression2
axiomes	R1 : distance.<.2km \rightarrow aller.à.pied R2 : ((non distance.<.2km) \wedge distance.<.300km) \rightarrow prendre.le.train R3 : (non distance.<.300km) \rightarrow prendre.l'avion R4 : (acheter.un.billet \wedge avoir.le.téléphone) \rightarrow téléphoner.à.l'agence R5 : (acheter.un.billet \wedge (non avoir.le.téléphone)) \rightarrow aller.à.l'agence R6 : prendre.l'avion \rightarrow acheter.un.billet R7 : (durée.>.2.jours \wedge être.fonctionnaire) \rightarrow (non prendre.l'avion) F1 : (non distance.<.300km) F2 : avoir.le.téléphone
règle de dérivation	Si A est un théorème, et si A \rightarrow B est un théorème, alors B est un théorème Vous avez reconnu cette règle comme étant le "modus ponens" !!

Les R_i ont été obtenues de « l'expert ». C'est la mémoire à long terme, ou base de connaissances du système; on ne les modifiera que si on y détecte une erreur.

Les F_j représentent un problème qu'on va soumettre au système. C'est sa mémoire à court terme, ou base de faits.

Ni la BdC ni la BdF ne sont programmées : elles sont mises telles quelles dans le système. Le seul programme est le moteur d'inférence, qui réifie la règle de dérivation :

```
ça marche
tant que ça marche
  ça ne marche pas
  boucle sur les Ri
    boucle sur les Fj
      si Ri est de la forme "Fj -> Fk"
        ajouter Fk à la BdF
        ça marche
      finsi
    finboucle
  finboucle
fintant
```

2.3.2 Déroulement

Lançons le système :

```
ça ne marche pas
R1 et F1 : rien
R1 et F2 : rien
R2 et F1 : rien
R2 et F2 : rien
R3 et F1 : F3 (prendre.l'avion) et ça marche
R3 et F2 : rien
R3 et F3 : rien
...
R6 et F3 : F4 (acheter.un.billet) et ça marche
...
R7 et F4 : rien
ça ne marche pas
R1 et F1 : rien
R1 et F2 : rien
R1 et F3 : rien
R1 et F4 : rien
...
R4 et F4 et F2 : F5 (téléphoner.à.l'agence)
```

NON! C'est vrai dans l'interprétation, mais ce n'est pas un théorème du système formel! Nous n'avons pas écrit, et encore moins programmé :

règle de dérivation	Si A est un théorème, et si
	B est un théorème, alors
	A \wedge B est un théorème

La distinction entre "et" et " \wedge " n'est pas un snobisme, pas plus que celle que nous faisons entre "si...alors..." et " \rightarrow ". Corrigeons :

```
ça marche
tant que ça marche
```

```

ça ne marche pas
boucle sur les Ri
  boucle sur les Fj
    si Ri est de la forme "Fj -> Fk"
      ajouter Fk à la BdF
      ça marche
    sinon
      boucle sur les Fl
        si Ri est de la forme "Fj ^ Fl ->..."
          ajouter Fm = (Fj ^ Fl) à la BdF
          ça marche
        finsi
      finboucle
    finsi
  finboucle
finboucle
fintant

```

Relançons le système :

```

ça ne marche pas
modus ponens avec R3 et F1 : F3 (prendre.l'avion) et ça marche
modus ponens avec R6 et F3 : F4 (acheter.un.billet) et ça marche
ça ne marche pas
règle du "et" avec F4 et F2 : F5 (acheter.un.billet ^ avoir.le.téléphone)
                               et ça marche

ça ne marche pas
modus ponens avec R4 et F5 : F6 (téléphoner.à.l'agence) et ça marche
ça ne marche pas
arrêt

```

NON! C'est vrai en Logique, mais pas en Informatique! Notre programme va continuer à produire:

```

modus ponens avec R3 et F1 : F7 (prendre.l'avion) et ça marche
...

```

Notre programme ne donne pas la solution en temps fini, car il boucle. Pour éviter cela, nous marquons les F_j au fur et à mesure que nous les utilisons:

```

ça marche
tant que ça marche
  ça ne marche pas
  boucle sur les Ri
    boucle sur les Fj non marqués
      si Ri est de la forme "Fj -> Fk"
        ajouter Fk à la BdF
        marquer Fj
        ça marche
      sinon
        boucle sur les Fl

```

```

    si Ri est de la forme "Fj ^ F1 ->..."
      ajouter Fm = (Fj ^ F1) à la BdF
      marquer Fj
      ça marche
    finsi
  finboucle
finsi
finboucle
finboucle
fintant

```

A présent, le programme est correct, et donne la réponse en temps fini, quel que soit l'ordre dans lequel nous écrivons les R_i et les F_j . Nous avons donc réalisé notre objectif: nous pouvons exprimer nos connaissances sans les programmer.

2.3.3 Autre exemple

Reprenons le même système, en changeant un peu la base de faits :

axiomes	R1: distance.<.2km → aller.à.pied
	R2: ((non distance.<.2km) ∧ distance.<.300km) → prendre.le.train
	R3: (non distance.<.300km) → prendre.l'avion
	R4: (acheter.un.billet ∧ avoir.le.téléphone) → téléphoner.à.l'agence
	R5: (acheter.un.billet ∧ (non avoir.le.téléphone)) → aller.à.l'agence
	R6: prendre.l'avion → acheter.un.billet
	R7: (durée.>.2.jours ∧ être.fonctionnaire) → (non prendre.l'avion)
	F1: (non distance.<.300km)
	F2: avoir.le.téléphone
	F3: durée.>.2.jours
	F4: être.fonctionnaire

Lançons le système :

modus ponens avec R3 et F1: F5 (prendre.l'avion)

modus ponens avec R6 et F5: F6 (acheter.un.billet)

règle du "et" avec F3 et F4: F7 (durée.>.2.jours ∧ être.fonctionnaire)

règle du "et" avec F4 et F2: F5 (acheter.un.billet ∧ avoir.le.téléphone)

modus ponens avec R4 et F5: F6 (téléphoner.à.l'agence)

modus ponens avec R7 et F7: F8 (non prendre.l'avion)

L"expert" va s'insurger : notre système a déduit, d'une part, qu'il fallait prendre l'avion (F5), et d'autre part qu'il ne fallait pas (F8).

Alors, sous réserve d'un stockage des déductions, nous pouvons interroger le système :

Q: pourquoi F5?

R: modus ponens avec R3 et F1

Q: pourquoi F8?

R: modus ponens avec R7 et F7

Q: pourquoi F7?

R: Règle du "et" avec F3 et F4

Nous savons à présent que l'erreur provient d'une contradiction entre R3 et R7, que nous montrons à l'expert. Comment corriger? En supprimant R3, et en le remplaçant par les bons axiomes, ceux qui traitent tous les cas. Où placer les nouveaux axiomes? N'importe où.

La mise au point ne coûte pas \$4000, parce que le système s'explique, et nous permet de trouver rapidement la cause de l'erreur la correction n'a pas à tenir compte du reste de la BdC

Terminologie Dorénavant, nous appellerons "règles" les axiomes de type Ri, c'est-à-dire les axiomes qui contiennent " \rightarrow ", et qui forment la BdC. C'est regrettable, car cela entretient une confusion avec les règles de dérivation, mais c'est l'usage. Une règle a une partie gauche (LHS) et une partie droite (RHS).

La LHS est une conjonction de prémisses, c'est-à-dire que les prémisses sont reliées par des \wedge . La partie droite est une conjonction de conséquents. En Prolog, la partie droite est à gauche, et il n'y a qu'un conséquent.

2.3.4 Généralisations

Les moteurs correspondant aux logiques d'ordre supérieur.

De 0 à 0+ Reprenons notre S.F., avec une nouvelle base de faits :

F1 : distance.=.500km

F2 : avoir.le.téléphone

Il ne se passe rien. Le système ne sait pas, à partir du symbole distance.=.500km , affirmer(non distance.<.300km) . Pour que cela soit possible, il faudrait "casser l'atome" :

Terminologie:

distance est un "truc valable" (ce n'est pas une variable) = est un prédicat (une fonction qui renvoie "vrai" ou "faux");

500km est une constante;

Nous avons quitté le Calcul des Propositions, ou Logique d'ordre 0, pour une Logique que nous appellerons "0+", dans laquelle apparaissent des "trucs valables", des constantes, et des prédicats. Le prix à payer est de programmer la sémantique des prédicats, c'est-à-dire d'écrire un programme capable, par exemple, de remplacer "distance = 500km" par "non distance < 300km".

De 0+ à 1 Supposons que le système me "dise" d'aller à l'agence. Ce serait bien qu'il me dise aussi comment aller à l'agence, sachant que la distance est de 1,5km. Mais il ne peut pas, parce que "distance" n'est pas une variable: si j'écrase "distance = 500km" par "distance = 1,5km", je vais obtenir des résultats contradictoires. Solution: dupliquer le système :

axiomes	R1 : distance.destination < 2km \rightarrow aller.à.pied.destination
	R1a : distance.agence < 2km \rightarrow aller.à.pied.agence
	R2 : ((non distance.destination < 2km) \wedge distance.destination < 300km) \rightarrow prendre.le.train
	R2a : ((non distance.agence < 2km) \wedge distance.agence < 300km) \rightarrow prendre.le.train.agence
	F1 : distance.destination = 500km F2 : distance.agence = 1,5km F3 : non avoir.le.téléphone

Mais supposons maintenant que je veuille travailler sur n objets: je ne peux pas "n-pliquer" mon système. Il faut que je puisse écrire :

R1 : quelle que soit la destination, $\text{distance}(\text{destination}) < 2\text{km} \rightarrow \text{aller.à.pied}(\text{destination})$

destination devient alors une variable, et j'ai introduit d'autre part "quelle que soit", qu'on appelle le quantificateur universel, noté \forall .

Je suis à présent en Calcul des Prédicats du Premier Ordre, ou "Logique d'Ordre 1". Mon moteur d'inférence va se compliquer, puisqu'il va devoir chercher, non plus à faire coller un "F" avec la partie gauche d'un "R", mais chercher quelles valeurs des variables permettent cet appariement.

De 1 à 2 Pour définir qu'une relation R est transitive, il suffit d'écrire :

$$\forall R(\forall x\forall y\forall z(xRy \wedge (yRz) \rightarrow (xRz)) \rightarrow \text{transitive}(R))$$

Mais on n'a pas le droit! En effet, on fait porter le quantificateur universel sur "R", une variable de relation, alors qu'on n'a le droit de le faire porter que sur une variable d'individu.

On passe alors à la Logique d'Ordre 2... mais ceci est une autre histoire.

Du monotone au non monotone Supposons à présent que j'écrive :

$$\text{fen\^etre.ouverte} \wedge \text{courant.d'air} \rightarrow \text{fermer.la.fen\^etre}$$

"fermer.la.fen\^etre" est un théorème, qui doit s'entendre: "il faut fermer la fen\^etre". Mais ce qui m'intéresserait, c'est qu'un robot domotique dispose de telles connaissances, et qu'il aille effectivement fermer la fen\^etre avec ses petites mains. "fermer.la.fen\^etre" devient alors une action. Mais si mon robot ferme la fen\^etre, il y aura contradiction entre l'état du monde et sa Base de Faits, qui contiendra encore "fen\^etre.ouverte". Donc, il faut compléter :

$$\text{fen\^etre.ouverte} \wedge \text{courant.d'air} \rightarrow \text{fermer.la.fen\^etre} \wedge \text{non fen\^etre.ouverte}$$

$$\text{Ce qui revient à écrire: } A \wedge \dots \rightarrow \text{non } A \wedge \dots$$

Or toute la Logique repose sur l'interdiction d'une telle formule!

Nous créons donc (dans les années 70), les Logiques non monotones, dans lesquelles cela n'est plus interdit (non monotone, par analogie avec l'Analyse: le nombre de théorèmes n'est plus une fonction non-décroissante du temps). Nous gagnons en puissance d'expression, mais nous perdons quelque chose de très important: la certitude de terminer en temps fini. En effet, la même situation peut se reproduire (quelqu'un ré-ouvre la fen\^etre), et il faudra appliquer à nouveau la même règle aux mêmes faits et donc le système peut boucler. On remarquera que cette notion de monotonie/non-monotonie est indépendante de l'Ordre de la Logique considérée.

Du non monotone au temporel Si je dis à présent à mon robot :

$$\text{oeuf.cru} \wedge \text{eau.froide} \rightarrow \text{allumer.le.gaz}$$

Dans 3 minutes, je pourrai affirmer "non oeuf.cru". Mais pendant ces trois minutes? Donc je vais programmer mon système pour qu'il ne fasse aucune déduction à ce propos pendant 3mn. On entre dans les Logiques temporelles. Si je dis maintenant :

$\text{non en.stock(article)} \wedge \text{besoin(article)} \rightarrow \text{commander(article)}$ c'est pareil, sauf que je ne sais pas quand arrivera le livreur.

Du bi-valué au flou Jusqu'à présent, nous appuyant sur les S.F., dans lesquels une expression est, ou n'est pas, un théorème, nous ne nous sommes intéressés qu'au "vrai" et au "faux". Or, dans la vraie vie, les choses ne sont pas si simples. Il y a des choses dont

nous ne savons pas si elles sont vraies ou fausses. On peut donc souhaiter manipuler les 3 valeurs : (V ? F). Mais cela peut encore sembler trop étroit : on peut avoir besoin d'une graduation plus fine : (Vrai Probable Inconnu Improbable Faux). Et puis pourquoi ne pas continuer ? On peut considérer l'intervalle continu $[-1 +1]$. Mais c'est encore simpliste : à une notion donnée, on peut associer une certitude, comprise entre -1 et +1, et une précision, comprise entre 0 et 1. On entre alors dans le domaine des Logiques floues (fuzzy).

Le problème devient alors :

Si $A \rightarrow B$ avec une certitude CR et une précision PR, et si A est connu avec CA et PA, on peut en déduire B, mais combien valent CB et PB ?

Et après Jusqu'ici, nous sommes restés "objectifs". Si à présent nous voulons introduire "l'homme dans la boucle", nous pouvons envisager des opérateurs tels que "X croit que Y est vrai", "X sait que Y est vrai", etc. Avec des règles plus ou moins contraignantes. Par exemple, "nul n'est censé ignorer la loi" s'écrira :

$$\forall X \forall A, A \rightarrow \text{sait}(X, A).$$

2.3.5 Synthèse

Il n'y a pas une Logique, mais un grand nombre de Logiques. Lorsque vous décidez de résoudre un problème en faisant appel à des connaissances, la première chose à faire est de vous situer dans l'espace à 4 dimensions ci-dessus, ... et de regarder si le moteur d'inférence correspondant existe.

2.3.6 Programmation par contraintes

Problématique Dans chaque matière, le TD ne peut pas avoir lieu avant le cours. La salle C3 contient au maximum 36 étudiants. Le cours d'Infographie nécessite du matériel audiovisuel, qui ne se trouve que dans les salles C1 et C2. L'effectif du cours de Probas-Stats est 70. Le cours de Logique est commun aux deux options. Le cours de Systèmes d'Exploitation a comme pré-requis la première moitié du cours d'Algo. Mr B. enseigne l'Algo et la Compil. Madame S. n'est pas là le Mercredi. Monsieur C. vient de loin, il faut lui grouper ses cours sur une journée. Si on met Probas-Stats et Méthodes Numériques le même jour, les étudiants déjantent. Si ils n'ont pas cours de 10h à 16h, ils risquent de ne pas revenir à 16h... Faites moi l'emploi du temps. Je viens d'apprendre que Mr A. a eu un accident ; il est en arrêt pour trois semaines. Aménagez-moi l'emploi du temps.

Analyse Dans un tel problème, on distingue :

- des contraintes fortes : la même personne ne peut faire deux cours simultanément
- des contraintes faibles : il vaut mieux ne pas placer un cours le Vendredi de 16h à 18h,
- une fonction à optimiser : de deux solutions, on choisira celle qui fait faire le moins d'aller-retours aux enseignants qui effectuent leur recherche ailleurs que sur le campus.

Dans de tels problèmes, on distingue également :

- les problèmes sur-contraints : n'importe comment, il faudra violer certaines des contraintes faibles, car il n'y a pas de solution,
- les problèmes sous-contraints : on a l'embarras du choix, donc on va s'attacher à minimiser la fonction coût,

- les problèmes qui ont une solution et une seule: montrez m'en un!

On distingue d'autre part :

- Les problèmes booléens: chaque variable vaut soit "vrai" soit "faux". Exemple: l'Anglais habite la maison rouge, l'Allemand boit de la bière, le Français a un chien, la maison bleue n'est pas à côté de la maison de l'Italien... qui boit du jus de citron?
- Les problèmes en nombres entiers. Exemple: mon sac à dos contient au plus 45kg, une boîte de choucroute pèse 1kg et m'apporte 100cal, 15 lipides, 12 glucides, une tablette de chocolat pèse 100g et m'apporte 250cal,... comment dois-je remplir mon sac pour avoir le maximum d'énergie avec un régime équilibré?
- Les problèmes en nombres "réels". Exemple: calculer la puissance nécessaire pour le ventilateur, sachant que plus il est gros plus il évacue de chaleur, mais que plus il est gros plus son moteur chauffe.
- Les problèmes non numériques. Exemple: quelle forme donner à la voiture pour qu'elle séduise un maximum de clients? Nous n'aborderons pas ici ces problèmes.

Enfin, on peut considérer différents types de contraintes :

- les contraintes unaires: $x < 5$,
- les contraintes binaires: le fil A ne doit pas toucher le fil B,
- les contraintes ternaires: aucun obstacle ne doit se trouver sur la droite reliant la lampe à la cellule photo-électrique etc : la distance entre A et B doit être inférieure à la distance entre C et D.

Par ailleurs, en ce qui concerne la solution, tout dépend si :

- je veux la solution optimale,
- je me contenterai d'une solution approximative dans x minutes,
- il me faudra une solution, mais je ne sais pas quand ("anytime algorithms").

Formalisation Un problème de satisfaction de contraintes (CSP) est un quadruplet $[X, D, C, f]$, où

- les X_i sont les variables,
- les D_i sont les domaines possibles pour chaque variable,
- les C_j sont les contraintes, et.,
- f est une fonction à minimiser.

Dans certains cas, f n'existe pas, on serait déjà bien content de trouver une solution. Nous n'aborderons pas ici les problèmes d'optimisation multi-critères, dans lesquels il y a plusieurs fonctions à minimiser.

On appelle affectation, le fait d'instancier certaines variables X_i par des valeurs x_i (évidemment prises dans les domaines D_i). Cette affectation peut être totale (elle concerne toutes les variables du problème) ou partielle. Une affectation est consistante si elle ne viole aucune des contraintes dans lesquelles interviennent ses variables.

Exemple: 4 personnes, U D C X, doivent traverser un pont. Le pont ne supporte que deux personnes. Il faut une lampe pour traverser, et on n'a qu'une lampe. U met 1 minute, D 2, C 5, et X 10. Si deux personnes traversent ensemble, c'est à la vitesse de la plus lente.

Il est clair qu'on va faire traverser 2 personnes, en faire revenir 1, traverser 2, revenir 1, et enfin traverser 2.

On a donc 8 variables: V11, V12, V2, V31, V32, V4, V51 et V52.

Le domaine est (U D C X).

Aux contraintes du texte s'ajoute l'évidence: V_{ij} différent de V_{ik}

Je peux affecter :

	UDCX /
X à V11 et U à V12 (coût : 10)	DC / UX
U à V2 (11)	DCU / X
D à V31 et C à V 32 (16)	U / DCX
C à V4 (21)	UC / DX
C à V51 et U à V52 (26)	/ UDCX

J'ai une affectation totale, consistante. Ajoutons une contrainte: le temps total doit être inférieur à 21.

	UDCX /
X à V11 et U à V12 (coût : 10)	DC / UX
U à V2 (11)	DCU / X
D à V31 et C à V 32 (16)	U / DCX
C à V4 (21)	UC / DX

Cette affectation partielle n'est pas consistante. Je vais donc backtrack. Au lieu de faire revenir C, je fais revenir X:

	UDCX /
X à V11 et U à V12 (coût : 10)	DC / UX
U à V2 (11)	DCU / X
D à V31 et C à V 32 (16)	U / DCX
X à V4 (26)	UX / DC

Inconsistance, je backtrace.

	UDCX /
X à V11 et U à V12 (coût : 10)	DC / UX
U à V2 (11)	DCU / X
D à V31 et C à V 32 (16)	U / DCX
D à V4 (18)	UD / CX

Et je repars

	UD / CX
U à V51 et D à V52 (20)	/ UDCX

Gagné. Mais cette affectation est-elle optimale?

Pour le savoir, relançons le backtrack bien que nous ayons une solution évidemment en nous arrêtant dès que nous trouvons une solution pire qu'une solution que nous avons déjà...

Au bout d'un temps... largement supérieur à 20 minutes, nous allons trouver une solution meilleure, en 17 minutes. Au bout d'un temps encore plus grand, nous aurons épuisé tous les cas, ce qui nous permettra de prouver que la solution optimale coûte 17 minutes.

Si, au lieu de faire développer cet arbre par une machine, nous utilisons notre intelligence, nous parvenons au raisonnement suivant :

- C'est X qui coûte le plus cher, c'est à lui que je dois m'intéresser en premier (ce qui ne signifie pas que c'est lui que je dois faire passer en premier).
- Il faut qu'il passe, ce qui me coûte 10, il ne faut pas qu'il revienne, ce qui me coûterait 30.

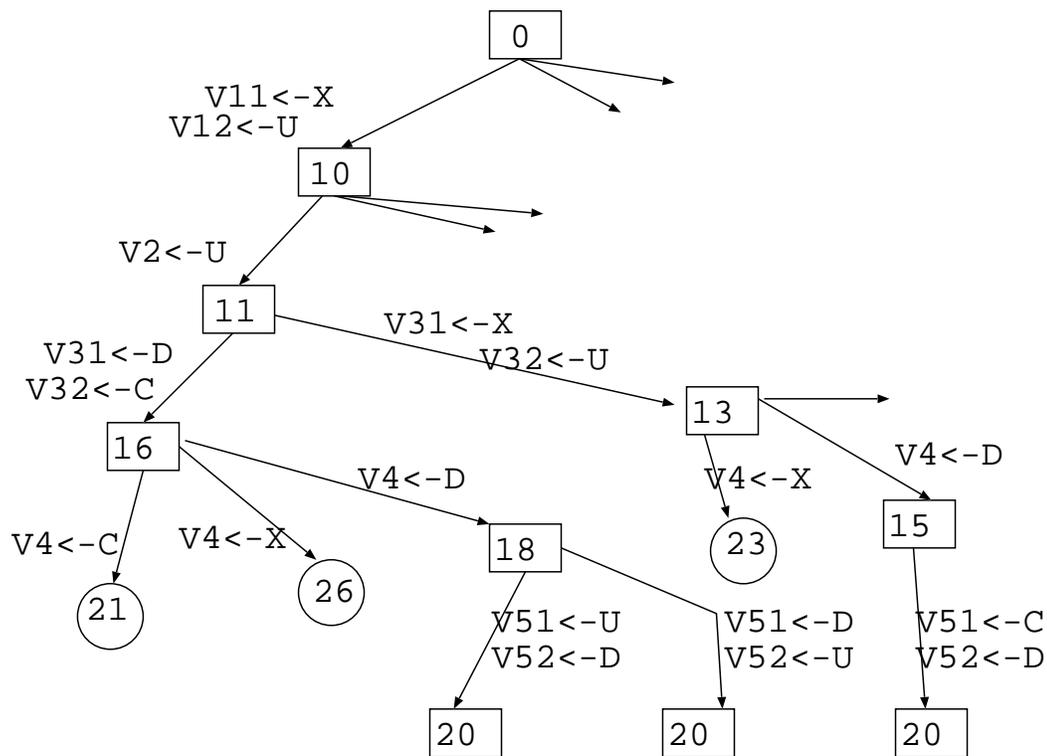


FIG. 15 – Arbre de résolution

- Ensuite, c’est C qui coûte le plus cher, mais s’il passe avec X il ne me coûte rien!
- Il faut donc qu’ils passent ensemble, et qu’ils ne reviennent pas.
- Donc ils ne peuvent pas faire partie du premier voyage (l’un des deux doit rapporter la lampe).
- Et pas non plus du dernier (l’un des deux doit avoir rapporté la lampe).
- Donc le premier voyage concerne U et D, et le dernier aussi.

Ce raisonnement est séduisant (merci), mais est-il généralisable à plusieurs milliers de variables? Non bien sûr. Toutefois, il fait apparaître deux choses :

- Il semble utile de s’intéresser d’abord aux éléments les plus contraints.
- Il n’est peut-être pas nécessaire d’envisager tous les cas.

Le premier point ci-dessus illustre la notion de (méta)-heuristique.

Le second point introduit la notion de méthode incomplète.

Une méthode incomplète ne cherche pas à envisager tous les cas, mais cherche à trouver le plus vite possible une solution acceptable : arrivé à un noeud de l’arbre,

- on élimine (définitivement?) certaines branches,
- on ordonne les branches à explorer.

Il est donc possible qu’on ne trouve pas la solution optimale; il est par ailleurs certain que l’on ne pourra pas prouver que la solution trouvée est optimale. Il est même admissible que la solution trouvée viole certaines contraintes.

Ces méthodes nécessitent l'introduction de notions plus fines que la simple consistance :

- Un problème est dit k -consistant si, quelle que soit une affectation consistante de $k-1$ variables, et quelle que soit une autre variable X_k , on peut trouver une affectation consistante pour X_k .
- Un problème est dit noeud-consistant s'il est 1-consistant, c'est-à-dire si on peut affecter n'importe quelle valeur (dans le domaine, bien sur) à n'importe quelle variable sans violer aucune contrainte. En d'autres termes, le problème est bien posé.

Ces définitions n'ont pas grand intérêt a priori. Elles en prennent lorsqu'on progresse, c'est-à-dire lorsqu'on a déjà proposé une affectation partielle et qu'on considère le sous-problème restant.

Certaines méthodes incomplètes consistent à donner une première affectation totale, probablement non consistante, puis à la perturber astucieusement pour l'améliorer. Supposez par exemple que vous vouliez faire entrer un maximum de petits pois dans une boîte : vous en mettez autant que vous pouvez, puis vous secouez la boîte, et vous pouvez alors en rajouter (attention, cette méthode est inefficace à bord d'un vaisseau spatial).

3 Connaissances : expertise et expérience

3.1 Nouvelle génération de systèmes informatiques ?

Les applications de l'informatique couvrent un domaine aux frontières sans cesse repoussées. La puissance de calcul des processeurs, l'interconnexion des systèmes en réseau, la capacité immense des moyens de stockage numérique de l'information, la disponibilité du multimédia sur le moindre des postes de travail et surtout l'émergence d'outils de développement facilitant l'exploitation de ces moyens ont favorisé le surgissement de l'outil informatique dans la panoplie de la plupart des métiers et services. Dans le même temps, le rôle joué par l'informatique dans la résolution de problèmes s'est diversifié et intervient dans des contextes qui changent de plus en plus vite et où la capacité à s'adapter se révèle une qualité de plus en plus recherchée. Après (ou plutôt à côté de) une informatique «normalisatrice», semble émerger une informatique «exploratrice» cherchant à rendre des services souples (aussi bien en calcul qu'en gestion d'informations) à des utilisateurs souhaitant échapper au maximum au cycle fermé [spécifications, développement, mise en oeuvre, maintenance]. Idéalement le système informatique est ressenti comme un auxiliaire possédant des qualités de calcul et de mémoire très importantes auquel on pourra poser des questions (ou des problèmes).

Cette confiance et ce nouvel enthousiasme dans les possibilités des systèmes informatiques permet aujourd'hui un retour au premier plan des techniques de l'intelligence artificielle qui étaient en veille depuis plusieurs années dans les milieux industriels particulièrement dubitatifs sur l'intérêt de cette approche dans ce que d'aucuns appellent «le monde réel». Comme la recherche internationale n'a pas faibli pendant ces années de relative désaffection de l'intérêt industriel, des transferts de technologie de plus en plus nombreux permettent aux techniques qui font appel à l'apprentissage et à la réutilisation de l'expérience d'être l'objet d'articles et de présentations dans les revues techniques industrielles. Nos travaux ont commencé dans la période de doute des industriels et se développent actuellement dans des conditions bien plus favorables, portés qu'ils sont par l'intérêt général pour la gestion efficace des connaissances et savoir-faire de l'entreprise. *Exploiter intelligemment les informations disponibles* est le souci commun des différentes recherches menées. Cette exploitation ne peut être *intelligente* que si elle correspond ou collabore à la tâche d'un utilisateur dans son travail. Nous utilisons ici un peu abusivement le terme coopération homme-machine pour exprimer plus simplement que le système informatique se place comme auxiliaire (puissant et inépuisable) de l'homme. La notion d'explication s'entendra donc comme argumentation d'une réponse proposée à un utilisateur.

Ce chapitre a pour objet de préciser ce que nous entendrons par expertise et expérience dans les tâches de résolution de problèmes de façon à pouvoir éclairer nos premières contributions et soutenir notre point de vue sur la coopération des connaissances issues d'expertises avec celles issues de l'expérience. Les tâches pour lesquelles nous avons proposé des assistances fondées sur des systèmes à base de connaissance relèvent de la résolution de problèmes et, dans une moindre mesure, dans cette période initiale de nos travaux de l'aide à la décision.

3.2 Tentative de définitions pour les concepts d'expertise et d'expérience

La démarche de recherche présentée et les travaux l'illustrant s'inscrivent dans une problématique générale : « *Etudier les mécanismes informatiques permettant d'assister de façon la plus coopérative possible les tâches de résolution de problèmes de l'utilisateur* ».

Associer expertise et expérience s'est imposé progressivement comme principe pour y parvenir. Il convient donc de clarifier cette démarche en précisant le sens que nous donnons aux différents termes et expressions dans cette présentation.

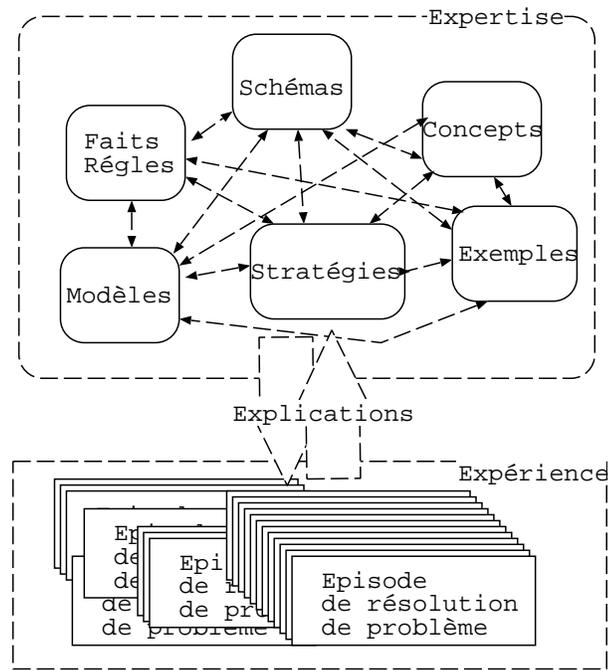
La **résolution de problèmes** est un champ d'études bien connu dans plusieurs disciplines des sciences cognitives [Houdé et al., 1998]. Les modèles « computationnels » issus de ces travaux ont largement inspirés les chercheurs du domaine de l'intelligence artificielle. Toutefois, la résolution de problèmes y est étudiée en essayant de mettre en œuvre des heuristiques qui garantissent qu'une solution pourra être trouvée et qu'en énumérant des propositions, on est capable de juger qu'il s'agit bien de solutions. « *On cherchera à mettre en œuvre une procédure complète (trouvant une solution s'il en existe une), qui termine (est complète et s'arrête nécessairement s'il n'y a pas de solution) et qui est admissible (termine et fournit une solution optimale relativement à un certain critère)* ». Cette définition idéale ne correspond que rarement à la réalité des problèmes à résoudre. Nous utilisons le terme de résolution de problèmes en relâchant explicitement et de manière la plus contrôlée possible complétude et correction, en laissant l'homme dans la boucle de résolution de problèmes. Dans la mesure où l'utilisateur est dans la boucle de résolution, la tâche principale du système à étudier relève alors de **l'aide à la décision**.

L'aide à la décision, thème majeur en intelligence artificielle [GDR-PRC-IA, 1997] suppose de définir ce qu'est un problème de décision. En effet, la tâche d'un décideur comporte souvent plusieurs facettes :

- l'identification des acteurs concernés ;
- la formulation du problème ;
- l'élaboration d'une liste d'actions possibles ;
- la collecte de l'information préférentielle.

Après cette phase de structuration de la décision à prendre, le problème de décision consiste à élaborer une recommandation, par exemple sous la forme d'une sélection des bonnes actions ou d'un classement de celles-ci. Les systèmes réalisés à l'occasion des recherches présentées dans ce mémoire s'inscrivent tous dans l'assistance à tout ou partie de ces facettes de la tâche de décision. En intelligence artificielle [Lang and Perny, 1997], comme dans la tradition de la théorie des jeux [Von Neumann and Morgenstern, 1947], on cherche surtout à fournir la décision optimale selon un certain critère et avec un risque (une chance) bien maîtrisée. Notre approche consiste plutôt à fournir le cadre le plus adapté à la décision, c'est à dire celui dans lequel une décision « satisfaisante »⁷ devrait pouvoir s'imposer. On voit bien que dans le cas limite, cela revient à la définition précédente (idéale) qui suppose que la décision puisse se calculer (se déduire) sans ambiguïté à partir du cadre dans lequel on doit la prendre. A l'opposé, dans le cas de situations complexes ou non complètement modélisées, le meilleur cadre de décision est celui qui limite le champ des décisions à prendre de manière implicite, c'est à dire sans être forcément capable d'énoncer l'ensemble des décisions ainsi guidées. L'assistance à la décision consiste alors à fournir un cadre qui

7. Satisfaisante au sens d'un critère de validité connu éventuellement *a posteriori*.


 FIG. 16 – *Expérience et expertise : complémentarité*

ait le plus «de chances» de guider l'utilisateur dans sa décision⁸. Nous verrons dans la suite de ce manuscrit des cas concrets illustrant cette façon de voir l'aide à la décision.

Le terme **expertise**⁹ sera réservé dans notre exposé pour désigner la **capacité à résoudre un problème** [Chi et al., 1988]. Cette expertise est donc potentielle, mobilisée au moment d'une résolution de problèmes. L'expertise est construite de règles, de faits, de connaissances conceptuelles, de schémas de résolution, de connaissances stratégiques [Mayer, 1992]. Ces connaissances d'expertise sont d'autant mieux interconnectées qu'elles ont été fréquemment mobilisées pour résoudre des problèmes.

Le terme **expérience**¹⁰ sera utilisé dans son sens le plus général mais en restreignant sa portée aux **épisodes de résolution de problèmes**. L'expérience serait ainsi constituée d'expériences distinctes mais situées dans un état d'expertise précis. Le schéma de la figure 16 illustre ces définitions.

Les connaissances d'expertise sont classées selon des catégories souvent évoquées [Bonnet et al., 1986], [Hoc and Nguyen-Xuan, 1986], [Richard, 1984], [Richard, 1990] ou encore [Weil-Barais et al., 1993] et ce classement est lié au contexte de la résolution coopérative de problèmes, présenté plus haut comme un type d'aide à la décision.

En dehors d'une modélisation cognitive où ces catégories ne représentent que des modèles abstraits des différentes formes de mobilisation de connaissance qui peuvent être observées (selon des protocoles expérimentaux), ces catégories correspondent bien aux

8. Cette situation suppose que l'on n'a pas pu représenter l'ensemble des actions possibles, mais que l'on est capable de modéliser les situations de décision. En théorie, rien n'empêche d'acquérir ces connaissances plus spécifiques de l'expérience.

9. Absent du «Vocabulaire de sciences cognitives [Houdé et al., 1998].

10. Selon [Houdé et al., 1998], le terme d'expérience est utilisé dans deux acceptions. Selon l'acception la plus générale, l'expérience est la rencontre entre l'esprit et la réalité...Selon l'acception plus technique du terme, une expérience signifie une expérimentation...

principales formes de représentation de connaissances¹¹[Kayser, 1997] (sans y intégrer les variantes fort nombreuses) exploitées concrètement en informatique pour réaliser des systèmes à base de connaissances. On reconnaîtra facilement :

- « – **Les règles et les faits** : la logique [Thayse, 1991] fonde les techniques de l'intelligence artificielle, comme son histoire [Crevier, 1993] le montre, et fournit le support de son étude dans la plupart des ouvrages introductifs [Haton et al., 1991], [Winston, 1988]. Le mode d'inférence privilégié est la déduction logique.
- « – **Les concepts** : la conceptualisation en informatique fournit une façon commode de gérer des objets complexes en les organisant dans un réseau de relations. Directement issue des études psycho-cognitives [Quillian, 1969], l'approche est formalisée sous forme de réseaux sémantiques [Sowa, 1991], de graphes conceptuels aux propriétés formelles intéressantes [Chein and Mugnier, 1992] et connaît un développement important avec la définition des logiques de descriptions [Napoli, 1997] dans la ligne des propositions faites par Brachman et disponibles dans KL-ONE [Brachman and Schmolze, 1985]. L'inférence principale est la classification.
- « – **Les schémas** : l'idée de départ fut proposée dans le monde de la psychologie cognitive [Bartlett, 1932], repris sous des formes diverses [Minsky, 1975], [Schank and Abelson, 1977] avant de trouver des usages informatiques de plus en plus nombreux (langages de «frames») pour exprimer des objets complexes aux attributs possédant de multiples facettes. Les langages à objets (qui ne sont pas tous des langages de représentation de connaissances) [Masini et al., 1989] sont typiques de cette approche. Des langages comme Shirka [Rechenmann, 1985], KEE [Fikes and Kehler, 1985], Yaffool [Ducourneau and Quinqueton, 1986], KRS [Van Marke, 1988] ou encore Mering [Ferber, 1989] s'inscrivent dans cette famille. Les modes d'inférence les plus souvent retrouvés sont le filtrage, l'héritage, l'appariement et la classification. Ces modes d'inférence sont partagés avec les langages implantant les logiques de descriptions, autorisant donc de considérer que ces derniers langages appartiennent également à la famille des langages à objet (par exemple C-CLASSIC [Cohen and Hirsh, 1994]).
- « – **Les modèles** : le modèle d'un système est une description de ce système effectuée, selon des règles de modélisation, pour faciliter son étude. La modélisation est un exercice habituel pour tous ceux qui conçoivent des systèmes informatiques. Les modèles fonctionnels, structurels, causaux, etc, sont autant de descriptions s'attachant à mettre en évidence les relations fonctionnelles, structurelles, causales, etc, entre différents éléments d'un système permettant d'inférer à partir de leur connaissance. En psychologie cognitive [Gentner and Stevens, 1983], le modèle d'un système peut servir comme guide pour comprendre et expliquer un autre système considéré comme analogue. En logique, la notion de modèle est très différente du sens commun : un modèle M possède trois constituants : un univers, une interprétation sur cet univers, des symboles de prédicat et de fonction et, enfin, une assignation des variables libres sur les éléments de l'univers [Kayser, 1997]. Les modèles dans le schéma de la figure 16 s'entendent donc comme des descriptions formelles exprimant les relations existantes entre les différents éléments modélisés de façon à permettre l'inférence¹².

11. La notion de formalisme de représentation de connaissances sera réservée à la syntaxe choisie, tandis que la forme désigne de manière plus abstraite quelle forme « logique » est privilégiée pour représenter les connaissances.

12. Ce qui suppose bien entendu, que les relations soient elles-mêmes décrites avec des propriétés fournissant le mécanisme d'inférence.

- « – **Les exemples** : un exemple est un **problème type**¹³ résolu. Il présente l'énoncé, la solution et (parfois) la manière de passer de l'énoncé à la solution. L'exploitation d'un exemple en résolution de problèmes a été étudié en psychologie cognitive [Ross, 1987], et a démontré que la difficulté de réutilisation résidait dans l'élucidation des similarités pertinentes (utiles dans le contexte de résolution courant). Un exemple **bien expliqué** dans le cadre d'une théorie peut ainsi servir de cadre à l'apprentissage [Ellman, 1989].
- « – **Les stratégies** : il s'agit de métaconnaissances [Pitrat, 1990] (des connaissances sur les connaissances de base), particulièrement orientées pour la résolution de problèmes. Ces connaissances restent des connaissances et peuvent s'exprimer selon les modes précédents. Nous ne garderons en général que deux niveaux de connaissances : les connaissances de base et les connaissances stratégiques.
- « – **Les épisodes de résolution de problèmes** : un épisode de résolution de problèmes possède à peu près la même structure qu'un exemple, mais correspond précisément à une instance de résolution située dans le temps et dans l'espace. Les éléments constitutifs de cet épisode proviennent aussi bien de ce qui aura été le fait du système informatique que ce qui aura été le fait de l'utilisateur. Chaque élément constitutif est «expliqué», c'est à dire mis en relation avec «l'expertise». Ce réseau d'explications peut provenir de la trace d'inférences menées ou d'hypothèses de l'utilisateur. *Un épisode de résolution de problèmes est donc le résultat tangible d'une coopération entre l'homme et la machine dans un cadre précis et (largement) explicable.*

Il faut comprendre des efforts de définition précédents qu'ils ne prétendent pas fonder universellement le sens des termes employés, mais fournissent plus modestement les limites qui seront les leurs dans l'exposé des travaux de recherche.

La dualité expertise/expérience correspond à une évolution dans la démarche de recherche. Les travaux tendant à exploiter l'expertise seule pour fournir des assistants **intelligents** dans la prise de décision en résolution de problèmes par l'homme ont été en effet les premiers réalisés. Ils ont montré l'intérêt de la démarche de résolution coopérative, en particulier dans l'étude de la modélisation des situations à risques et de leur résolution (préventive ou interactive). La recherche s'est alors orientée vers l'étude de la prise en compte effective de l'expérience concrète dans l'assistance à la résolution de problèmes.

13. problème type = problème abstrait de ses éléments factuels (contextuels de sa première résolution par exemple) n'intervenant pas dans sa résolution.

4 L'acquisition des connaissances : charnière du passage de l'expérience à l'expertise ou comment ne pas oublier l'expérience

Ce chapitre a pour objectif d'introduire de manière concrète les invariants et les différences de l'acquisition de connaissances selon que l'on vise une expertise partagée ou que l'on cherche à rendre compte de l'expérience réutilisable.

Une première section présente les approches orientées «tâches» habituellement utilisées pour les méthodes et les outils du génie cognitif.

Nous présentons dans la section suivante ce qui caractérise nos propositions sur la base de notre expérience illustrée par deux projets caractéristiques :

- « – Supervision en qualité d'une production vaccinale: système expert associé à des cartes de contrôle multivariées (Mérieux 93). La modélisation d'une expertise multiexperts (expérience non prise en compte).
- « – Programmation de fours à micro-ondes guidées pour la minéralisation: système d'aide à la programmation strictement basée sur la réutilisation de l'expérience (Prolabo 96). La modélisation de l'expérience des utilisateurs praticiens chevronnés (faible expertise explicite).

La conclusion tente de montrer à quel point l'expérience gagne à être baignée dans une théorie qui l'éclaire, tandis que l'expertise encapsulée tire profit de la présence de l'expérience accumulée pour illustrer ses modèles.

4.1 Les méthodes, techniques et outils de la littérature: une approche pratiquement toujours tournée vers l'expertise

Le rapide tour d'horizon que nous proposons sur les méthodes, techniques et outils de l'acquisition de connaissances n'a pas valeur d'état de l'art. Nous n'avons pas en effet effectué une recherche méthodique dans ce domaine. Par contre, nous en avons une pratique importante autour de projets qui ont mobilisé ce que l'on a appelé des «cogniticiens»¹⁴ et des experts¹⁵.

C'est probablement Newell[Newell, 1982] qui en 1982 a officialisé l'autonomie de l'ingénierie des connaissances en proposant un niveau spécifique de représentation, «le niveau des connaissances», différent du «niveau symbolique». Une véritable communauté autour de l'ingénierie des connaissances s'est constituée avec ses manifestations¹⁶ et ses journaux¹⁷, tout en prenant une bonne place dans les manifestations et revues de l'intelligence artificielle mais aussi du génie logiciel. Cette activité de plus de 10 ans a produit des résultats importants et qui ont trouvé des aboutissements dans des méthodes et des outils structurant et guidant l'acquisition des connaissances :

- « – dès 1986, la notion de **tâches génériques** est proposée [Chandrasekaran, 1986], tandis que la communauté proposait des méthodes pour les trouver [O'hara and Shalbolt, 1993], ou pour faciliter leur réutilisation [Aben, 1993];

14. En pratique, tout analyste est confronté au problème de la mise en évidence des connaissances des experts.

15. Le terme d'expert est associé à toute personne dont les connaissances seront d'une manière ou d'une autre exploitées dans le «code» de l'application informatique.

16. Conférence annuelle INGÉNIERIE des CONNAISSANCES en France par exemple.

17. KNOWLEDGE ENGINEERING au niveau international par exemple.

- « – des méthodes «ad hoc» sont proposées dans le cadre de ce que l'on appelle encore des systèmes experts [Reitman-OLson and Rueter, 1987], pour fournir des outils pratiques aux «cogniticiens» [Firlej and Hellens, 1991], en y associant parfois le point de vue des utilisateurs [Zaff et al., 1993];
- « – des outils d'aide à la conception de systèmes à bases de connaissances sont développés sur les principes issus de la recherche [Erikson, 1994], dont les plus célèbres sont fondés sur les méthodes KADS [Schreiber et al., 1993] et COMMET [Steels, 1991]¹⁸. On pourrait citer également la contribution française de Nathalie Aussenac [Aussenac, 1989]. Enfin une offre importante se développe autour de l'accès direct des experts à des «formulaire d'expression d'expertise» en intranet ou internet [Gaines and Shaw, 1998] ou [Maurer and Dellen, 1998].

La validation des différentes méthodes proposées pour guider l'acquisition de connaissances et réutiliser les connaissances d'un système à l'autre est maintenant au centre des préoccupations de la communauté de l'ingénierie des connaissances (une session complète de 9 communications était réservée à ce thème à KAW98¹⁹).

Il est maintenant bien admis que l'acquisition de connaissances est d'abord un travail de **modélisation** plutôt que d'**extraction** [Krivine and David, 1991]. Le processus d'acquisition de connaissances est orienté par le modèle principalement partagé²⁰ [Charlet, 1992], [Chandrasekaran et al., 1998], [Seroussi and Morice, 1993] y compris pour les systèmes basés sur le raisonnement à partir de cas [Janetzko et al., 1993].

Acquisition des connaissances et apprentissage automatique se conjuguent actuellement selon plusieurs approches et notamment par :

- « – l'exploitation d'exemples pour guider la description conceptuelle [Bareiss et al., 1988], [Porter et al., 1990];
- « – l'intégration des techniques de l'apprentissage dans le processus d'acquisition de connaissances [Van De Velde and Aamodt, 1994];
- « – l'apprentissage à partir des erreurs pour adapter la théorie en fonction de l'expérience [Stroulia and Goe...

L'ingénierie des connaissances couvre un champ d'application nouveau avec les enjeux de la capitalisation des connaissances dans l'entreprise. L'organisation des savoirs de l'entreprise est au centre des préoccupations des organisations [Brunet and Ermine, 1994] et la connexion entre cette problématique et celle de la gestion des systèmes d'information est à l'ordre du jour [Alquier and Tignol, 1998], [Simon, 1998].

L'acquisition de connaissances s'inscrit donc comme un processus qui n'est envisageable qu'en organisant le système d'information autour de l'idée même de capitalisation continue des connaissances. Les efforts actuels sont orientés vers les méthodes d'évaluation et de validation des bases de connaissances constituées et la question même de la faisabilité de la validation est posée.

Notre hypothèse de travail est que la meilleure validation est l'utilisation effective des connaissances en résolution de problèmes. La trace d'utilisation de ces connaissances est une connaissance fiable sur leur valeur en situation d'utilisation. Le Raisonnement à Partir de Cas pour la résolution de problèmes se présente alors comme une approche particulièrement bien adaptée pour un tel objectif. Nous partageons avec Isabelle Bichindaritz

18. Des méthodes comme KOD [Vogel, 1988] ou MACAO [Aussenac, 1989] ont été proposées au sein de la communauté française.

19. 11th Workshop on Knowledge Acquisition, Modelling and Management, 18-23 Avril 1998, Banff, Alberta, Canada.

20. Nous voulons dire par là qu'il existe toujours une problématique majeure qui s'exprime selon une modélisation appropriée et souvent partagée par les acteurs impliqués.

[Bichindaritz et al., 1997] le point de vue que l'apprentissage est lié indissolublement à la réutilisation (donc à la nécessaire adaptation pour y parvenir).

Nous développerons cette hypothèse dans la section présentant plus particulièrement nos travaux en Raisonnement à Partir de Cas.

Notre contribution au champ d'étude de l'acquisition des connaissances se traduit par le témoignage d'une pratique de terrain, aboutissant à l'hypothèse de travail précédente. Les paragraphes suivants illustrent deux expériences de terrain particulièrement importantes.

4.2 Synthèse de notre expérience de terrain : deux projets impliquant l'acquisition de connaissances

4.2.1 Supervision en qualité d'une production vaccinale : système expert associée à des cartes de contrôle multivariées (Mérieux 93)

Rapports internes [Mille et al., 1994], [Mille and Pialot, 1994]

La maîtrise de la qualité représente un enjeu considérable en production industrielle et l'idée principale du projet est d'exploiter les informations recueillies au cours de la fabrication d'un lot de production en vue de le situer par rapport aux lots antérieurs afin de détecter au plus tôt d'éventuelles dérives de qualité. Des Cartes de Contrôle Multivariées (qui a donné son nom CCMV au projet) représentent graphiquement les résultats d'une analyse statistique multivariée. Ces cartes sont utilisées comme tableaux de bord des différentes productions connues et la fabrication en cours y est projetée pour comparaison (voir figure 17).

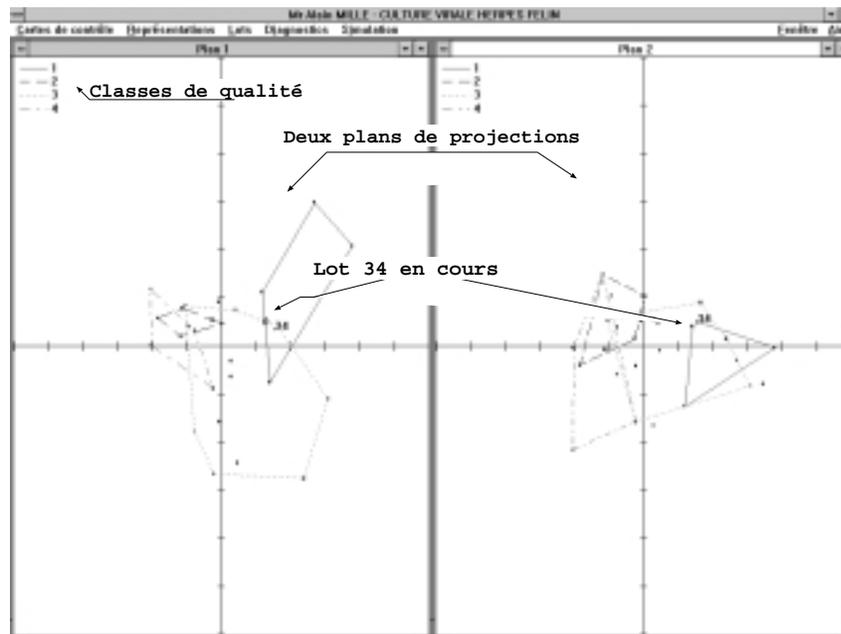


FIG. 17 – Tableau de bord fondé sur les cartes de contrôle multivariées

Si une dérive est diagnostiquée, les opérateurs doivent en comprendre les raisons et imaginer des actions correctives éventuelles. Pour aider les opérateurs dans ces tâches, il a été décidé d'expérimenter le couplage de l'approche statistique à un système «expert»

sollicité en cas de dérive et chargé de fournir les éléments d'explication et de préconisation adéquats.

Le projet s'est déroulé sur deux années comprenant l'étude de l'historique, le recueil des connaissances sur la fabrication choisie comme application pilote et la réalisation d'un prototype industriel complet intégrant toutes les fonctions (illustration dans les figures 18 et 19).

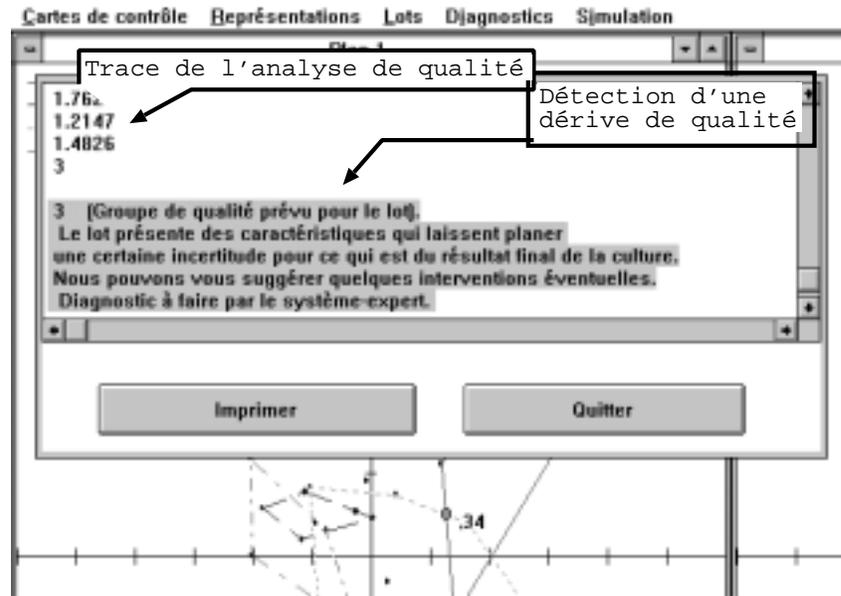


FIG. 18 – Analyse qualité, détection de dérive et suggestion d'appel du module de diagnostic

Les tâches principales assignées au système sont donc :

- « – détection d'une dérive de qualité (la qualité dépend de plusieurs critères - rendement, efficacité, aspect, etc.) ;
- « – analyse des causes possibles de la dérive et présentation de l'interprétation (cf. figure 18) ;
- « – diagnostic, explications et proposition d'actions correctives (cf. figure 19).

L'hypothèse est faite que l'expertise nécessaire aux tâches fixées existe collectivement dans l'entreprise. Cette connaissance doit être disponible pour aider l'opérateur dans les tâches correspondantes de diagnostic puis de correction.

Analyse de l'historique et élaboration de la carte de contrôle multivariée L'information contenue dans l'historique de production est segmentée et on analyse séparément l'historique de la production du principe actif brut, l'historique de la purification et celui de la mise sous forme pharmaceutique, de manière à obtenir des cartes de contrôle multivariées locales, correspondant à chaque zone. L'historique de chaque étape de la production sélectionnée comporte à la fois des informations sur le procédé (conditions de déroulement de l'étape considérée) et sur la qualité obtenue. Dans le cas général, plusieurs paramètres de qualité sont disponibles. Les relations entre ces paramètres sont analysées séparément en prenant en considération l'ensemble de l'historique, soit en utilisant une analyse en composantes principales (si tous les paramètres sont quantitatifs), soit en utilisant une analyse des correspondances multiples (si certains paramètres sont qualitatifs).

Les groupes de qualité qui serviront de référence ont été établis en se basant sur les résultats de cette analyse multivariée (4 groupes dans le cas de l'exemple prototype, soit schématiquement «très bien», «bien», «juste suffisant» et «insuffisant»). Chaque lot de l'historique est rattaché à un groupe de qualité. La partie «procédé» de l'historique est alors considérée. On cherche à mettre en relation les observations sur le procédé pour le lot et l'appartenance à un groupe de qualité, en utilisant une analyse factorielle discriminante sur variables qualitatives en raison de la présence constante de paramètres qualitatifs dans les procédés. Les plans factoriels de l'analyse discriminante fournissent le support de représentation de la carte de contrôle multivariée. Cette approche s'inspire des propositions de [Costecalde, 1984] et de [Pialot, 1990]. L'enveloppe convexe de chaque groupe de qualité est dessinée dans les plans de la carte de contrôle. En fonction des caractéristiques du procédé, tout lot réel ou simulé est représenté en projection dans les plans de la carte. Il est facile de vérifier immédiatement de quel groupe de qualité le lot apparaît proche d'après les caractéristiques du procédé et faire un pronostic de qualité en fonction des résultats déjà obtenus pour des lots proches dans le déroulement du procédé. Ainsi la carte de contrôle multivariée automatise une analyse par similitude. Il est également possible de rapprocher la position occupée par le lot, des positions occupées par les lots précédents ce qui offre l'opportunité de détecter visuellement l'apparition d'une évolution (peut-être une dérive) des conditions du procédé. A partir des caractéristiques réelles ou simulées du procédé, on calcule la distance du lot au centre de chaque groupe de qualité. Le lot est rattaché au groupe dont il apparaît le plus proche au sens de cette distance. En cas d'éloignement important de chacun des groupes, le lot peut n'être rattaché à aucun des groupes de qualité. Dès lors que le lot est rattaché au groupe non satisfaisant, on évalue la contribution de chacun des paramètres du procédé à la position du lot et on incrimine les paramètres qui y contribuent le plus (deux paramètres sont ainsi retenus dans l'exemple prototype). Ces paramètres sont cités dans le diagnostic statistique et transmis au système expert. Comme on le voit, cette approche s'apparente à certaines techniques de fouille de données fondées sur l'analyse statistique des données pour découvrir des règles de causalité en particulier.

Recueil des expertises et formalisation des connaissances La nature cloisonnée de la fabrication et le nombre de zones concernées dans le procédé de fabrication a imposé de recueillir les connaissances de nombreux experts. Les experts sont des cadres de l'entreprise, en situation opérationnelle et possédant expertise théorique et expérience pratique conséquente. Ces experts sont les responsables des zones de fabrication et des zones de contrôle. Ils collaborent tous à la fabrication et connaissent le lot chacun selon des points de vue qui peuvent être assez différents. Leur tâche principale est la mise en œuvre du procédé dans leur zone mais ils ont également comme tâche plus rare d'expliquer les problèmes de qualité quand ils surviennent. La procédure d'explication d'une dérive de qualité consiste à rassembler les acteurs de la fabrication et des contrôles et à étudier le lot en cause pour remonter aux événements de son histoire qui pourraient expliquer sa dérive. Cette procédure n'est pas décrite explicitement. Il s'agit plutôt d'une sorte d'enquête pour vérifier tous les éléments d'information concernant le lot puis de se livrer à un «brain storming» pour expliquer ce qui s'est passé. Les informations relatives à ces travaux sont confidentielles et sensibles. L'entreprise souhaite que le plan de recueil d'expertise soit décrit par avance et que le calendrier de mise en œuvre ne soit pas remis en cause.

Un groupe de pilotage est mis en place, avec :

- « – un chef de projet Industriel (représentant officiel de l'entreprise) ;

- « – un responsable industriel et un responsable scientifique pour l'organisation pratique du recueil des connaissances ;
- « – un groupe de quatre interviewers.

Huit experts sont désignés par le chef de projet de façon à ce que les différents secteurs de production et de contrôle soient représentés. Un expert généraliste complète la liste et, dans un secteur qui vient de changer d'équipe très récemment, deux experts sont associés. La méthode de travail est très classique dans son principe :

- « – exploitation des documents existants ;
- « – série d'interviews enregistrées, retranscrites et contrôlées ;
- « – réunions de synthèse.

Un «fil rouge» doit guider les entretiens selon les principes suivants :

- « – L'expert situe de manière «ouverte» sa place dans la fabrication.
- « – L'expert décrit les tâches liées à son secteur.
- « – Des études de cas sont réalisées. C'est sur ces études que l'expert est amené à réfléchir « à voix haute » .
- « – Les informations issues des cas sont généralisées en règles.

La réalisation pratique du recueil des connaissances a nécessité un peu plus de six mois.

Calendrier	Descriptif succinct et situation dans le temps	Nb/type
RP1 à T_0	Groupe de Pilotage	3 réunions sur 2 mois
RP2 à $T_0 + 2$ mois	Groupe de Pilotage et groupe d'experts	une réunion
RP3 à $T_0 + 2,5$ mois	Groupe de pilotage	une réunion
INT1 à $T_0 + 3$ mois	Une interview/expert	6 interviews sur 1 mois
RS1 à $T_0 + 4$ mois	Synthèse , groupe de pilotage	une réunion
INT3 à $T_0 + 4$ mois	Une interview/expert	6 interviews sur 1 mois
RS2 à $T_0 + 4,5$ mois	Groupe de Pilotage, et groupe d'experts	une réunion
INT4 à $T_0 + 5$ mois	1 expert Interviewé, 1 expert Interviewer, 1 Observateur-cogniticien	6 réunions sur 1 mois
RS3 à $T_0 + 6$ mois	Groupe de pilotage, Groupe d'experts, un spécialiste en communication	2 réunions sur 1 mois
RT à $T_0 + 7$ mois ⇒ $T_0 + 8$ mois	Groupe de pilotage restreint, Réunion de Transfert	3 réunions sur 1 mois

L'acquisition de connaissances fut un processus particulièrement contrôlé compte tenu de la confidentialité des connaissances formalisées et des coûts induits par la mobilisation des cadres de l'entreprise. Les méthodes de recueil d'expertise correspondent assez bien aux méthodes recommandées à l'époque avec beaucoup de reformulations, des retranscriptions systématiques, et l'intervention d'un «médiateur» pour décoder les biais d'acquisition. Les tâches d'interprétation et de décision des experts ont été bien appréhendées. Le choix (préalable) de représenter les connaissances sous la forme de règles et

métarègles a considérablement gêné le transfert de connaissances des experts (plus de 600 pages de connaissances informelles dactylographiées) vers une base de connaissances (100 règles et 20 métarègles...). Le système est opérationnel, mais le système expert ne rend probablement compte que d'une très faible partie des connaissances recueillies. L'impact de ce projet dans l'entreprise fut important dans la mesure où il a mis en lumière les rôles effectivement joués par chacun dans la qualité globale. La qualité se joue sur l'ensemble du processus de fabrication et les protocoles très strictes de passage d'une zone à l'autre cachent l'interaction des différentes phases.

Le deuxième exemple sera présenté de manière beaucoup plus courte, mais mérite d'être évoqué dans la mesure où il pourrait se situer à l'opposé dans l'approche d'acquisition de connaissances.

4.2.2 Programmation de fours à micro-ondes guidées pour la minéralisation : système d'aide à la programmation basée sur la réutilisation directe de l'expérience (Prolabo 96)

Rapports internes [Mille et al., 1995], [Mille et al., 1996]

Un minéralisateur est un appareil destiné à préparer un échantillon pour l'analyse de ses composants chimiques élémentaires. La préparation consiste à casser toutes les chaînes moléculaires existantes pour «libérer» les composants recherchés. L'appareil est constitué d'un système de chauffage rapide par micro-ondes guidées et d'un ensemble d'injecteurs de produits chimiques pilotés par des pompes. L'appareil est automatisé et la réalisation d'une minéralisation est faite selon un programme de minéralisation qui est défini comme illustré par la figure 20.

Notre intervention a consisté à :

- « – modéliser le protocole de minéralisation et les concepts qui lui sont reliés ;
- « – modéliser la tâche d'adaptation telle qu'elle est mise en œuvre par un expert chargé de concevoir un protocole de minéralisation ;
- « – réaliser une maquette de faisabilité avec des outils «universitaires»²¹ qui a été extrêmement importante pendant la phase de modélisation ;
- « – enfin réaliser un prototype complètement opérationnel qui sera rapidement présenté dans le chapitre sur les réalisations de l'équipe.

Un seul expert a travaillé pour l'élaboration du projet. La société sortait d'une tentative (malheureuse) de réalisation d'un système expert classique basé sur un générateur de système expert²². Les connaissances nécessaires à la mise en œuvre d'un système d'aide à la conception d'un programme de minéralisation se déclinent selon trois modèles :

- « – modèle de cas²³ ;
- « – modèle conceptuel du domaine²⁴ ;
- « – et un modèle d'adaptation²⁵.

La réalisation de l'ensemble du projet depuis les premiers contacts jusqu'à la mise en œuvre d'un prototype opérationnel (testé chez des clients de l'industriel) a duré une année.

21. Nous avons utilisé essentiellement le moteur CLIPS et REMIND.

22. Guru est un outil qui a connu son heure de gloire dans les années 80.

23. Un cas = une recette = une sorte de plan.

24. Appareils, fonctions, produits, dangers, etc.

25. Algorithme de modification de plan selon des écarts locaux à réduire à différents niveaux d'abstraction.

Nous retenons de cette expérience que :

- « – la réalisation d'un système à base de connaissances reposant sur le Raisonnement à Partir de Cas ne dispense pas d'une phase de modélisation des connaissances conséquentes même pour un problème bien ciblé ;
- « – la disposition de cas dès le début permet la simulation et la validation progressive (et les remises en question) de la modélisation ;
- « – les cas remplissent le rôle des connaissances à «grain fin» dans le système à base de connaissances ;
- « – les connaissances d'expert sont ciblées «naturellement» sur la résolution de problèmes du fait même de la présence de cas réels concrétisant la réussite ou l'échec des connaissances modélisées pour les résoudre.

Cette expérience industrielle²⁶ a permis de faire avancer considérablement notre compréhension et donc la formalisation des connaissances à modéliser en présence d'expertise et d'expérience conjuguées.

4.3 Apprendre : associer expertise et expérience dans un réseau d'explications mutuelles ?

Notre proposition concernant l'acquisition des connaissances permettant l'hybridation de l'expertise et de l'expérience se structure selon plusieurs idées principales :

- « – L'expérience étant (dans notre définition) représentée par des épisodes de résolution de problème, la modélisation des connaissances est orientée par les points de vue liés à ce contexte²⁷.
- « – Les connaissances ainsi modélisées peuvent (doivent) évoluer avec l'émergence de nouveaux points de vue, c'est à dire de nouveaux cas de résolution de problème. La base de connaissances explique les cas d'une classe de problèmes, et s'explique par eux selon une «projection» particulière à ce point de vue.
- « – Les techniques d'apprentissage automatique peuvent être exploitées comme des méthodes d'acquisition «continue» de connaissances. La disponibilité de l'expérience sous la forme de cas permet d'apprendre à plusieurs niveaux du cycle du RàPC : apprendre à indexer les cas (avec sa tâche duale : apprendre à retrouver les cas), apprendre à adapter, apprendre les concepts se dégageant des cas résolus.
- « – L'utilisateur d'un système à bases de connaissances est une source essentielle pour évaluer son intérêt et pour guider son évolution en fonction des tâches qui sont les siennes. L'expérience individuelle d'un utilisateur en interaction avec un système à base de connaissance doit permettre une «remise en cause» naturelle et permanente des connaissances exploitées par le système pour assister la(les) tâche(s) de l'utilisateur lui-même. Les cas de résolution sont un moyen commode pour permettre à un utilisateur de modifier indirectement une base de connaissance. Les travaux sur la modélisation de la réutilisation chez l'homme et en particulier sur les mécanismes liés à la gestion de l'échec sont une source d'inspiration pour cet axe particulier.

Cette ambition d'approfondir les rapports entre expertise et expérience se concrétise dans les nouveaux projets de recherche qui sont amorcés et présentés dans le dernier chapitre.

26. Par opposition aux travaux qui s'inscrivaient explicitement dans une démarche de recherche.

27. La notion de contexte doit s'entendre au sens général du terme.

Diagnostic à faire par le système expert.
 Numéro du lot 151
 Le : 04 /08 /94 à 20 :06 :12
 Variables pesantes (Paramètres qui pourraient être en cause) :

Paramètres	Niveau	Effet
Volume en culture virale	3	0.0740
Fin Ampli. Cell.	1	0.0490
Concentr. Cell. Cult. virale à J0	1	0.0400
pH fin Cult. Cell.	1	0.0390

Paramètres soumis au système expert :
 Volume en culture virale
 Concentr. Cell. Cult. virale à J0

Compte rendu de l'analyse du système expert :

 Pour le diagnostic concernant la variable " concentration cellulaire en culture virale à J0 " je travaille sur une valeur "saisie"

Pour le diagnostic concernant la variable " volume en culture virale " je travaille sur une valeur " saisie "
 ***** diagnostic r27

La concentration cellulaire est plutôt faible en considération des situations observées dans l'historique. Les opérations de numération et de préparation du milieu se sont-elles déroulées correctement ?

FIG. 19 – Exemple de diagnostic simple du système expert

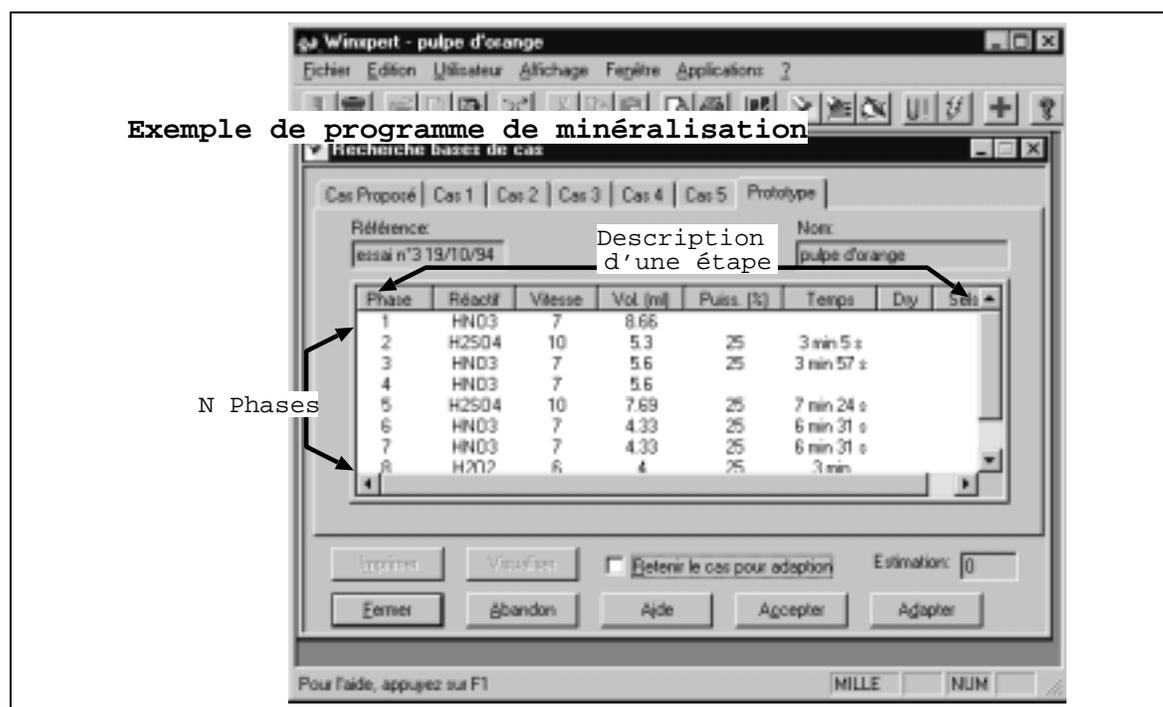


FIG. 20 – Description d'un programme de minéralisation

5 État de l'art sur les recherches en raisonnement à partir de cas

5.1 Mise en contexte de la recherche

La recherche dans le domaine du Raisonnement à Partir de Cas s'est affirmée d'abord aux États-Unis avec les célèbres conférences « DARPA » commencées en 1988 [Kolodner, 1988], avant de s'imposer en Europe avec la première conférence Européenne en 1993 à Kayserslautern [Richter et al., 1993], puis avec la première conférence internationale à Lisbonne en 1995 [Veloso and Aamodt, 1995]. Ces deux dernières conférences ont lieu en alternance chaque année. Plusieurs sites WEB proposent des bibliographies et des forums sur le sujet. Le plus actif est probablement [<http://www.surveying.salford.ac.uk/ai-cbr-mirror>] qui permet d'atteindre la plupart des autres sites du monde. Il n'existe pas encore de journal spécifique à ce domaine de recherche mais de nombreux journaux ont consacré des numéros spéciaux au raisonnement à partir de cas, et d'une manière générale les publications sur le raisonnement à partir de cas ont acquis leur place dans les journaux spécialisés de l'intelligence artificielle mais aussi dans de très nombreux journaux spécifiques aux domaines d'application principaux comme la conception par exemple.

Il semble que le Raisonnement à Partir de Cas soit maintenant plus orienté vers les sciences cognitives aux États-Unis tandis que les chercheurs européens appartiennent majoritairement à la communauté des techniques de l'intelligence artificielle. De fait, le domaine du Raisonnement à Partir de Cas constitue un pont « naturel » entre les deux communautés et plusieurs nouveaux projets de recherche mobilisent des chercheurs des deux spécialités.

Nous nous situons naturellement dans la communauté informatique des chercheurs en intelligence artificielle, mais sommes persuadés que l'exploitation de l'expérience pour aider à la tâche de l'utilisateur doit s'appuyer sur les résultats des travaux des sciences cognitives portant sur la manière dont l'homme réutilise son expérience pour la résolution de problème. Il en sera donc souvent question dans la suite du document.

Pour présenter de manière synthétique les tendances actuelles de la recherche en Raisonnement à Partir de Cas, nous présenterons d'abord les tendances concernant le paradigme RàPC en tant que sujet de recherche, puis nous nous appuierons sur le modèle maintenant bien établi par [Aamodt and Plaza, 1994] et illustré dans la figure 21 pour situer les travaux plus spécialisés se rapportant à tel ou tel aspect du RàPC.

Nous ne présenterons pas les outils du RàPC et n'évoquerons les applications (très nombreuses en conception, diagnostic et assistance à l'utilisateur) que par le biais des retombées générales pour la recherche qu'elles ont permises. Pour disposer d'un état de l'art particulièrement complet sur ces derniers points, nous engageons le lecteur à se reporter au dernier ouvrage de Watson [Watson, 1997].

Pour mettre en valeur le dynamisme propre de la communauté émergente des chercheurs en RàPC, nous séparerons le tour d'horizon international des contributions que nous avons sélectionnées parmi les contributions récentes en France. Le même plan de présentation sera gardé.

Enfin, nous concluons en situant ces tendances en regard des *Directions stratégiques en Intelligence Artificielle* telles qu'elles ont été synthétisées dans [Doyle and Dean, 1996]²⁸.

28. Ce rapport ne fait pas référence aux recherches sur le Raisonnement à Partir de Cas, ce qui n'est pas très étonnant dans le contexte américain, mais fait autorité sur les enjeux majeurs de la recherche en intelligence artificielle.

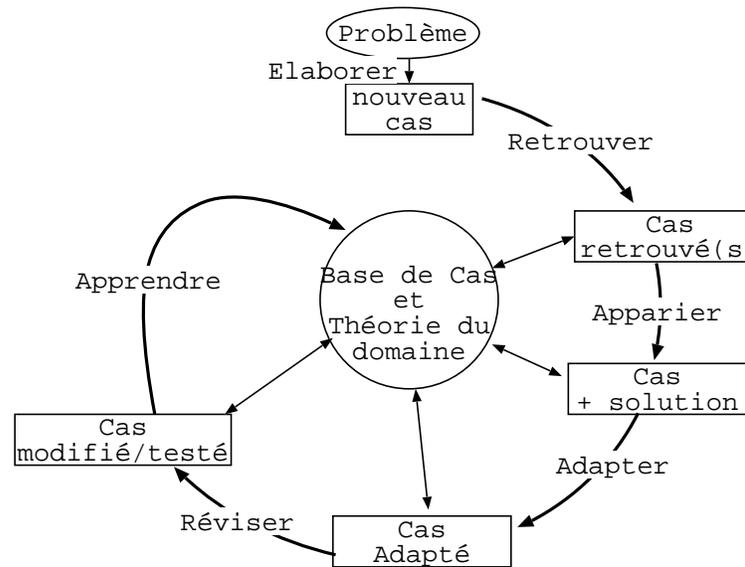


FIG. 21 – Le cycle du raisonnement à partir de cas

5.2 Le RàPC : Tendances de la recherche.

5.2.1 Principales tendances au niveau international

Efforts de modélisation et de formalisation du RàPC

La *modélisation formelle* de l'ensemble du cycle RàPC reste à faire. Il s'agit en effet d'un système d'inférences non monotones par essence²⁹. Il est habituellement classé comme une sorte de raisonnement par analogie [Haton et al., 1991], ce qui de notre point de vue occulte la différence importante qui existe entre transfert (Raisonnement analogique [Coulon et al., 1990]) et adaptation (RàPC)³⁰. Les différentes phases du cycle ont longtemps été considérées comme largement indépendantes. Il est maintenant bien admis que les connaissances exploitées pour mener à bien chaque étape ne sont pas indépendantes et que adaptation et apprentissage se situent au centre d'un réseau de dépendances. Nous retiendrons toutefois deux contributions importantes faites pour formaliser le RàPC dans le cadre de la planification et sous les conditions de connaissances suffisantes pour démontrer la justesse d'une solution, Jana Koelher [Koelher, 1996] propose une formalisation à l'aide de logiques de descriptions qui a l'avantage de rester cohérente pour l'ensemble du cycle mais ne peut que difficilement prendre en compte l'adaptation de cas en l'absence de théorie complète, tandis que Ralph Bergmann et Wolfgang Wilke [Bergmann and Wilke, 1995] proposent une approche hiérarchique de cas abstraits mais nécessitant un changement de langage de représentation³¹. Nous verrons que la communauté française a fait preuve de beaucoup d'initiative sur ce thème.

29. Dans un système d'inférences monotone: $si A \models B$ alors $(A \wedge C \models B)$, ce qui n'est évidemment pas garanti dans un raisonnement à partir de cas où l'inférence que l'on peut tirer d'un fait constaté dépend complètement de son contexte.

30. Le transfert consiste à réaliser une mise en correspondance d'un réseau d'explications à un autre réseau d'explications sur la base d'une analogie « de forme d'explications », alors que l'adaptation consiste à réutiliser des explications analogues pour substituer des éléments de solutions par d'autres éléments choisis selon l'écart d'explication à réduire [Mille, 1995].

31. L'abstraction des cas est faite dans un langage de description qui n'est pas celui des cas eux-mêmes.

La modélisation cognitive qui a mené au paradigme du RàPC est due à Schank et son équipe de recherche [Schank, 1982] qui l'a popularisé en proposant des mécanismes de base pour le mettre en œuvre [Schank and Riesbeck, 1989]. Elle connaît de nouveaux développements dans les équipes de Kolodner pour partager l'expérience sur les réseaux [Petrushin and Kolodner, 1996] et Schank dans le domaine de l'enseignement [SCHANK, 1996]. En Europe des travaux tentant de faire le pont entre modèles cognitifs et systèmes de RàPC se sont intéressés à la modélisation des tâches RàPC pour faciliter l'acquisition de connaissances [Janetzko et al., 1993]. Ce thème est actif en France également dans le domaine de la réutilisation des connaissances pour la conception.

La modélisation conceptuelle sous la forme du cycle présenté en figure 21 tire ses racines des premiers efforts de présentation globale du RàPC comme [Slade, 1991] ou [Kolodner, 1993] et est maintenant largement partagée dans la communauté.

Nous avons proposé de préciser cette modélisation conceptuelle [Mille, 1995] en nous inspirant de l'approche de modélisation du raisonnement analogique [Py, 1994] et en analysant le rôle spécifique de l'adaptation dans le cycle de raisonnement.

Une modélisation du RàPC au niveau connaissance a été proposée par [Armengol and Plaza, 1993] (première approche en termes de tâches génériques) en prenant un système particulièrement bien connu CHEF [Hammond, 1986] comme exemple. Cet effort est particulièrement important pour clarifier les types de connaissances et les types d'inférences réalisées pendant le cycle RàPC.

C'est sur la base de ces travaux que nous avons proposé une modélisation des tâches du RàPC [Fuchs, 1997].

Une modélisation du RàPC en logique floue a été proposée par Yager dès 1996 dans [Yager, 1996] et précisée dans [Yager, 1997]. Le mode d'inférence est considéré comme fondamentalement de même nature et pour cela le cycle est réduit à deux étapes principales : l'étape de comparaison et l'étape de composition d'une solution. Deux techniques sont suggérées pour réaliser la composition de solution en RàPC : la médiane pondérée quand les actions du domaine sont constituées d'une liste ordonnée d'alternatives et une variante d'apprentissage par renforcement, pour les domaines dont les actions impliquent une sorte de plan. Nous verrons que ce rapprochement entre RàPC et logique floue a été également mené quasiment simultanément en France (et en Espagne) sur des bases différentes.

Une modélisation spécifique aux cas complexes utilisés en conception a été proposée. Une série de travaux³² ont fortement influencé la manière dont les problèmes de conception sont abordés avec le RàPC. Une synthèse de ces travaux est faite dans [Maher et al., 1995] qui pose le principe de trois formes de connaissances : un modèle de décomposition de problèmes en sous-problèmes, un modèle de descriptions d'épisodes de conception qui correspondent à la résolution de sous-problèmes, et un modèle d'adaptation sous forme de règles de transformation de formes.

Pour compléter ce tour d'horizon, nous devons citer la proposition « synthétique » du *Memory Based Reasoning* faite dès 1986 [Stanfill and Waltz, 1986] qui intègre l'ensemble du cycle dans un mécanisme d'activation mémoire, étudié sous un angle plus théorique dans [Brown, 1993] qui étudie le modèle de mémoire sous-jacent sous la forme d'un réseau avec passage de jetons, et mis en pratique sur un réseau massivement parallèle avec les travaux de Kitano pour la traduction automatique [Kitano and Yasunaga, 1992], [Kitano, 1993]. Le parallélisme a d'ailleurs été étudié séparément, pour accélérer la recherche de cas et une intéressante proposition a été faite dans ce sens pour le domaine de la planification

32. Ces travaux ne sont pas toujours référencés dans la communauté du RàPC mais ont été abondamment publiés dans la communauté de la recherche en conception.

[Kettler et al., 1994] qui permet d'envisager d'exploiter une très grande base de cas dans des conditions d'efficacité garantie.

Nous avons commencé l'exploration de cette voie pour étudier ses possibilités d'auto-adaptation [Boucard, 1996].

La recherche se décline ensuite classiquement selon les différents champs plus spécialisés d'étude du RàPC et en particulier :

- « – **La représentation des cas.** Au-delà de la traditionnelle représentation des « cas vecteurs » un nombre important de travaux se sont intéressés à la représentation des cas complexes [Voß, 1996a], [Bartsch-Spörl, 1995] notamment pour l'aide à la conception.
Nous avons été amenés à développer un langage objet pour la représentation des cas complexes dans le domaine de l'aide à la décision [Fuchs, 1997].
- « – **L'élaboration du cas.** Ces travaux visent à la mise en évidence de la pertinence de cas incomplètement décrits [Bento and Costa, 1993], ou cherchent à anticiper l'adaptabilité des cas qui seront remémorés [Leake et al., 1997a]. Nous avons pour notre part étudié cette partie dans le cadre des applications industrielles pour faciliter la description « gagnante » d'un cas [Herbeaux and Mille, 1998].
- « – **L'organisation des cas.** De nombreux travaux s'intéressent à l'organisation de la mémoire des cas pour améliorer l'efficacité de la recherche soit en prenant en compte l'efficacité des remémorations précédentes [Fox and Leake, 1995] ou plus classiquement en développant des techniques de gestion d'arbres de décision de plus en plus performantes [BRESLOW and AHA, 1997]. Le problème des très grandes bases de cas a été posé [Shimazu et al., 1993] mais cette question est en train de se renverser : comment « voir » les bases de données comme des bases de cas. Cette question est liée aux travaux sur la capitalisation des connaissances dans l'entreprise [Thompson, 1997].
- « – **La remémoration des cas.** Le choix d'une mesure de similarité est déterminant pour l'efficacité du RàPC. On a assisté à un foisonnement de propositions spécifiques issues aussi bien des travaux de l'analyse de données [Diday et al., 1982], que du monde des métriques mathématiques [Critchley and Cutsem, 1992] ou [Batagelj and Bren, 1995], qu'en psychologie cognitive [Tversky, 1977], sans oublier les travaux spécifiques au domaine de l'intelligence artificielle comme [Bisson, 1994]. Une part importante de la recherche s'est donc focalisée sur la recherche de mesures de similarités les plus génériques possibles. La sémantique d'une mesure de similarité est largement liée à la tâche de résolution de problème concernée comme l'a démontré Michael Richter [Richter, 1995b]. La question de la similarité est : quelles sont les chances que ce cas ancien soit utilisable pour résoudre ce nouveau cas sachant qu'ils sont proches sur un certain nombre de points. Les approches s'inspirant de la théorie de Dempster-Shafer sont particulièrement étudiées [Richter, 1995a]. Les équipes françaises ont été particulièrement actives dans ce domaine comme nous le verrons dans la section qui leur est consacrée. Une étude théorique a été très récemment publiée dans [Rudolph, 1997] qui tente de montrer le type d'inférence que la similarité porte réellement.
Notre contribution s'est limitée à la proposition de mesure de dissimilarité conceptuelle et événementielle pour prendre en compte la dynamique de situations pour un projet de supervision [Mille et al., 1999].
- « – **La réutilisation et l'adaptation.** L'adaptation a fait l'objet de nombreuses propositions dans les premières années de la recherche sur le RàPC ([Barletta, 1989],

[Converse et al., 1989], [Gentner, 1989], [Hinrichs, 1989], [Kass, 1989], [Turner, 1989] par exemple), et une classification des différents types d'adaptation est faite par Janet Kolodner dans [Kolodner, 1993]. Ce classement est fondé sur les différences de méthode (transformationnel/dérivationnel) et sur les différences de recherche des règles d'adaptation (règles simples liées aux variables, règles de modification de structure, ou recherche de valeur à remplacer par proximité mémoire). La phase d'adaptation a fait l'objet d'efforts plus systématiques depuis quelques années comme l'a montré l'atelier sur l'adaptation organisé à l'occasion de la conférence européenne sur l'intelligence artificielle de 1996 [Voß et al., 1996]. Dans la dynamique de cette conférence, Angi Voß propose dans [Voß, 1996b] une liste de critères pour classer les types d'adaptation réalisés dans un certain nombre de systèmes RàPC réalisés dans la communauté. Les critères de classification reprennent les critères classiques de type d'opérateur d'adaptation, mais sont étendus à la prise en compte du type de cas, du nombre de cas utilisés pour l'adaptation et sur l'évaluation de l'adaptation. Appliqué par exemple au domaine de la conception architecturale [Smith et al., 1995], les propositions faites dans [Purvis and Pu, 1995] exploitent les contraintes pour guider l'adaptation. Cette approche garantit des solutions cohérentes globalement quelle que soit la partie de cas adaptée. Elle suppose toutefois que le jeu de contraintes soit complet et correct, ce qui reste une tâche d'acquisition de connaissances difficile.

L'équipe du Trinity College à Dublin, quand à elle, s'est attachée à étudier l'adaptation dans plusieurs de ses aspects comme la complexité [Smyth and Cunningham, 1993] ou les dépendances qui existent entre les tâches et la connaissances d'adaptation [Hanney et al., 1995]. Enfin plus récemment, ils proposent d'apprendre des règles d'adaptation à partir des cas eux-mêmes [Hanney and Keane, 1997].

Plus récemment encore [Wilke and Bergmann, 1998] propose une étude des connaissances et des techniques d'adaptation en différenciant deux familles distinctes : la famille des méthodes fondées sur la transformation des solutions et la famille des approches « génératives », c'est à dire capables éventuellement de trouver la solution à partir de rien³³. Ils complètent leur étude en faisant une place à l'adaptation compositionnelle [Velo, 1997], qui consiste à adapter différentes parties d'un cas à partir de différents cas et selon des stratégies qui peuvent donc varier dans le spectre précédent.

Une contribution essentielle (à notre sens) a été faite dans [Hanks and Weld, 1995] qui propose un algorithme d'adaptation de plan fondé sur les cas et indépendant du domaine (SPA comme Systematic Plan Adaptor). Bien que, par nature, il s'agisse d'un algorithme génératif, l'effort pour le rendre indépendant des opérateurs (souvent liés au domaine) le rend particulièrement intéressant comme base de travail.

La communauté française a également participé activement à ce thème de recherche comme nous le verrons plus loin.

- « – **La révision et la mémorisation des cas** : des travaux très récents [Stroulia and Goel, 1998] proposent d'utiliser l'évaluation pour guider la mémorisation qui prend alors un véritable statut d'apprentissage. Nous trouvons dans [Leake et al., 1997b] une approche plus pragmatique. Le processus d'adaptation est au centre du dispositif et trois différents types d'apprentissage sont exposés : l'apprentissage de cas comme des plans-réponses, l'apprentissage d'adaptation de cas et l'apprentissage de la similarité.

33. Cette décomposition approfondit et précise celle qui avait été faite entre les adaptations transformationnelles et dérivationnelles.

L'apprentissage de plans réponse consiste à intégrer un nouveau plan (adapté d'un ancien ou non) dans la base. L'apprentissage de l'adaptation consiste à mémoriser des cas d'adaptation et à les indexer en fonction de leur utilisation. L'apprentissage de la similarité est fondé sur l'estimation d'un coût d'adaptation. Basé sur le couple nouveau problème - problème remémoré, un cas d'adaptation est recherché qui servira de base à l'estimation du coût d'adaptation. L'association de ces différentes approches d'apprentissage est toujours meilleure que l'utilisation d'une seule. La tâche de résolution de problème est opportuniste par essence et l'évaluation des réponses permet de gérer les conséquences d'incohérences locales. Lié à l'apprentissage, une approche originale a été proposée dans [Smyth and Keane, 1995] pour trouver des indices permettant « d'oublier » les cas qui ne seraient plus utiles dans une base.

Nous insistons sur cette approche liant apprentissage et adaptation parce qu'elle est une de nos sources d'inspiration pour l'étude du processus d'apprentissage à partir d'expérience tel que nous l'envisageons dans nos nouveaux thèmes de recherche.

Pour compléter ce tour d'horizon international, il convient de présenter les efforts de la communauté française. Cette communauté est récente, se structure sans précipitation, mais est très active. Nous participons à l'animation de ce groupe français de recherche et nous avons eu l'occasion de présenter une synthèse de ses travaux [Mille and Napoli, 1997] que nous reprenons ici en la complétant.

5.2.2 La recherche sur le raisonnement à partir de cas en France

Les contributions récentes s'organisent autour de quelques thèmes majeurs :

1. Modélisation et formalisation du RàPC

Nous distinguerons différentes approches :

« – **La modélisation cognitive du raisonnement à partir de cas chez l'Homme.**

Une série de travaux ont été consacrés à l'INRIA à l'étude de la réutilisation des connaissances chez l'homme et en particulier par la réutilisation de plans de résolution de problèmes [Visser, 1994]. Il est important de noter que l'approche cognitive du RàPC fait partie intégrante des efforts de la recherche en RàPC de la même façon que les travaux d'inspiration plus « informatique »³⁴.

« – **La formalisation du raisonnement à partir de cas par les logiques de description** Après le travail inaugural de Jana KOELHER [Koehler, 1994], des travaux proches ont été présentés dans [Napoli et al., 1996], et une étude de la formalisation du RàPC dans une logique de description est décrite [Salotti and Ventos, 1997] au moyen de critères homogènes, explicites et formels³⁵. Les cas sont organisés grâce à une taxinomie de concepts index, ce qui permet d'exploiter l'opération de classification automatique de la logique de description pour rechercher des cas similaires. Une phase d'apprentissage supervisé de concepts index est facilitée par un algorithme fondé sur la recherche du plus spécifique subsumant de deux cas.

34. Willemien Visser a ainsi animé avec Brigitte Trousse un atelier commun à l'occasion d'IJCAI'93 sur la réutilisation en conception et les journées françaises rassemblent régulièrement les deux communautés

35. Une généralisation de ce travail a été présentée pour l'appliquer aux graphes conceptuels dans [FARON and GANASCIA, 1997].

- « – **Modélisation fondée sur les ensembles flous.** [Dubois et al., 1997], comme [Yager, 1996], s'appuient sur des travaux menés sur l'interpolation fondée sur des relations de similarité pour proposer des modèles fondés sur les règles floues pour le Raisonnement à Partir de Cas. Dans ces approches, un cas est vu comme un nuplet de valeurs d'attributs précises. Les attributs sont séparés en un ensemble S pour la description du problème (S comme Source) et T pour les attributs solution (T comme Target)³⁶. L'hypothèse (forte) est faite que tous les attributs sont renseignés. L'adaptation est présentée comme une sorte d'interpolation pour obtenir les valeurs de T sachant la (dis)similarité existant entre les descriptions de problèmes S . L'existence de mesures de similarité entre attributs problème \mathbf{S} et attributs solution \mathbf{T} est supposée établie. Enfin la connaissance de fonctions de dépendances entre les attributs problème et attributs solutions est supposée également établie. Dans son expression la plus proche de l'esprit du RàPC, les auteurs posent que :

*Les plus similaires sont les problèmes s_1 et s_2 ,
le plus il est possible que les solutions t_1 et t_2 soient similaires.*

La proposition peut se formuler sous la forme d'une règle de possibilité :

$$\begin{aligned} \pi_{X|Y}(v, u) &\geq A(u) \text{ si } v \in B \\ \pi_{X|Y}(v, u) &\geq 0 \text{ si } v \notin B \end{aligned}$$

où $\pi_{X|Y}(v, u)$ estime à quel point $Y = v$ est possible quand $X = u$.

On peut dire plus littéralement : $\forall u$, si $X = u$, il est possible au moins au degré $A(u)$ que Y prenne sa valeur dans B . Si les travaux sur la similarité avaient déjà exploité les variables linguistiques³⁷ pour manipuler les connaissances et donc avaient exploité les mécanismes de calcul provenant de la logique floue, cette approche formelle permet d'analyser plus précisément en quoi le RàPC est une sorte de raisonnement hypothétique dans l'incertain.

2. **La représentation des cas.** Des contributions intéressantes ont été faites pour la représentation des cas décrivant des historiques. Un projet de modélisation de propagation d'incendie pour l'aide à la décision a été l'occasion de développer un modèle de cas permettant de représenter et manipuler des données temporelles et spatiales [Rougegrez-Loriette, 1994]. Par ailleurs, et dans un esprit qui relève plus du génie logiciel, [Jaczynski and Trousse, 1997] propose un environnement de développement d'applications fondées sur le RàPC et permettant la manipulation des historiques selon plusieurs méthodes. Ce système a été utilisé avec succès en particulier sur une application d'assistance à la recherche d'informations sur le WEB pour un groupe d'utilisateurs [Jaczynski and Trousse., 1997]. [Corvaisier et al., 1998] a proposé une modélisation d'épisodes de recherche d'information sur le WEB, intégrant les notions de session de recherche, de tentatives unitaires, d'unités de recherche et de vocabulaire utile pour réutiliser l'expérience.
3. **La remémoration.** Ce thème se décline selon plusieurs sous-thèmes :

- « – *Travaux sur les similarités.* Annonceur de l'étude théorique présentée dans

36. Cette distinction Source/Target est inhabituelle. En effet, un cas source est habituellement un cas de la bibliothèque, tandis que le cas cible est le cas à résoudre. Dans le contexte de l'article, les attributs source sont ceux qui permettront de retrouver les cas source, tandis que les attributs cibles sont ceux qui doivent être évalués.

37. L'usage de symboles représentant sous la forme d'un terme « bien choisi » une valeur « floue ». Par exemple, *jeune* est une variable linguistique dès l'instant où une fonction d'appartenance lui est associée.

[Rudolph, 1997],

, [Sebag and Shoenauer, 1993] a proposé une mesure de similarité fondée sur les règles, ce qui nécessite de voir chaque cas comme une règle avec comme prémisses les faits constitutifs du problème et en conclusion les faits déduits constituant la solution. Les propriétés de classification se sont révélées convaincantes. C'est dans le cadre d'une boîte à outils pour un atelier RàPC que [Mignot, 1997] propose de tenir compte de la structure d'un cas pour construire en parallèle la définition d'une mesure de similarité. L'élaboration de cette mesure est donc guidée par une métaconnaissance sur les effets de contexte sur la pertinence des mesures. De son côté, [Rifqi, 1996] a élaboré une formalisation des *mesures de comparaison*, qui permet de comparer des objets décrits à l'aide d'attributs dont les valeurs sont floues. L'auteur a déterminé deux familles principales de mesure constituant ainsi une extension du modèle de contraste de Tversky : les mesures de similitude (satisfiabilité, inclusion et ressemblance) et les mesures de dissimilarité.

« – *Travaux sur l'organisation des cas en hiérarchie d'objets*. C'est une façon de préparer par avance le raisonnement lui-même dès l'instant où le RàPC est considéré comme une extension du raisonnement par classification dans le contexte de résolution de problème [Koelher, 1996]. [Napoli et al., 1996] formalise cette approche dans le cadre d'une représentation basée sur les objets. C'est dans ce même esprit que [Coupey and Salotti, 1996] étend les logiques de description pour formaliser le processus de remémoration en RàPC.

« – *Organisation hybride de la mémoire des cas (neuronal/cas prototypes et instances)*. [Malek, 1996] propose d'utiliser un réseau neuronal à apprentissage incrémental fondé sur des cas prototypes pour organiser la mémoire de cas. La mémoire est constituée de deux niveaux : le niveau supérieur contient les prototypes qui représentent chacun un groupe de cas, tandis que le niveau bas est utilisé comme index pendant la phase de recherche de cas. Un mécanisme d'apprentissage continu permet de grouper dynamiquement les cas sous de nouveaux prototypes.

4. **L'adaptation.** : Une contribution originale et générique est proposée dans [Lieber, 1997] sur des principes énoncés dans [Lieber and Napoli, 1996] qui consiste à anticiper l'adaptation dans le processus même de remémoration, ce qui est l'idéal théorique pour trouver une solution facilement (et éventuellement sûrement) adaptable. Pour y parvenir, un ordre partiel, noté \succsim est défini sur l'ensemble des problèmes selon la définition : $pb_1 \succsim pb_2$ si toute solution de pb_1 peut se spécialiser en une solution de pb_2 . On associe à un problème source *source* de la base de cas un problème noté $idx(source)$ appelé index de *source* plus général que *source* : $source \preccurlyeq idx(source)$ et tel que la solution $sol(source)$ est généralisable en une solution $sol(idx(source))$. L'ensemble des index est organisé en une hiérarchie notée \mathcal{H}_{idx} . Pour résoudre un problème nouveau (*cible*), une description de *cible* est d'abord classifiée dans la hiérarchie \mathcal{H}_{idx} , ce qui permet d'identifier un ensemble d'index IDX composé d'éléments $idx(source)$ vérifiant : $idx(source) \succsim cible$. Deux cas sont alors possibles :

« – soit IDX est non vide et il existe un *chemin de similarité* tel que $source \preccurlyeq idx(source) \succsim cible$ et l'adaptation se réalise en deux étapes :

« – *généralisation* de $sol(source)$ en $sol(idx(source))$, en suivant la relation $source \preccurlyeq idx(source)$,

« – *instanciation* de $sol(idx(source))$ en $sol(cible)$, en suivant la relation $idx(source) \succ cible$

« – soit IDX est vide, alors on essaie de revenir au cas précédent en effectuant des modifications sur certains index de la hiérarchie $\mathcal{H}_{idx}(\psi(idx(source)))$ et sur $cible$ ($\varphi(cible)$) par des règles de réécriture de façon à arriver à $source \preceq idx(source) \rightarrow \psi(idx(source)) \succ \varphi(cible) \leftarrow cible$, c'est à dire à un chemin de similarité connu, donc permettant l'adaptation.

Cette approche a été combinée avec des propositions de notre équipe et sera présentée dans le cadre de la conférence [Fuchs et al., 1999].

5. **L'apprentissage** Une contribution à l'intégration de l'apprentissage dans le cycle RàPC est proposée dans [Bichindaritz, 1994] dont le système MNAOMIA permet l'apprentissage incrémental de concepts, pendant la réalisation des tâches comme le diagnostic, la planification d'un traitement ou le suivi d'un malade. Plusieurs hiérarchies de concepts différentes sont définies selon les différentes tâches cognitives. Cette association étroite de la représentation des concepts à la tâche est naturelle dans les systèmes de raisonnement à partir de cas.

5.3 La recherche dans le domaine RàPC par rapport aux directions stratégiques de la recherche en Intelligence Artificielle

Il serait particulièrement vain de vouloir énoncer des directions stratégiques de la recherche qui soient partagées par l'ensemble de la communauté internationale, mais il est important de choisir une référence pour situer nos recherches et leur orientation. Nous avons choisi de prendre comme grille d'analyse le rapport de Jon Doyle et Thomas Dean [Doyle and Dean, 1996] qui s'efforcent de tracer les lignes principales se dégageant des recherches menées, des réalisations réussies et tentent une formulation des enjeux de la décennie. La catégorisation proposée de la recherche en IA³⁸ rassemble des domaines d'application et des branches thématiques théoriques :

1. Articulation³⁹ et représentation des connaissances⁴⁰.
2. Apprentissage et adaptation qui étend les techniques de découverte statistique, analytique et scientifique et les mécanismes hypothétiques neurophysiologiques aux procédures qui extraient un large spectre de tendances générales, de faits et techniques à partir de règles, d'expérience et de données collectées.
3. Échanges de vues, planification et actions qui concernent les méthode d'aide à la décision, de construction de plans, de conception pour atteindre des buts spécifiés et le contrôle, l'interprétation, le diagnostic et la modification de plans comme la mise en œuvre des conceptions.
4. Parole et traitement du langage.
5. Interprétation d'image et synthèse d'image.
6. Manipulation articulée et déplacement autonome qui couvrent le champ de reproduction des degrés de liberté de l'homme dans l'usage de ses mains et des ses jambes pour l'accomplissement de ses tâches.
7. Agents autonomes et robots.

38. Nous garderons cet acronyme IA pour Intelligence Artificielle dans la suite du document.

39. Articulation s'entend comme mécanismes de mise en relation des différents types de connaissances.

40. Il s'agit de créer et compiler des catalogues explicites, formels et multifacettes de connaissances « ontologiques ». La nature ontologique des connaissances reste une question ouverte.

8. Systèmes multiagents pour étudier et mettre en place des comportements de résolution de problème en groupe.
9. Modélisation cognitive qui se focalise sur la simulation des traits comportementaux de la cognition chez l'homme.
10. Fondements mathématiques, qui s'emparent des techniques et concepts des autres branches comme autant de sujets de formalisation, d'analyse et de reconceptualisation.

Il nous semble que les sciences cognitives modernes fondées sur l'étude directe du cerveau par l'image en résonance magnétique auraient leur place dans cette catégorisation, et que la réutilisation des connaissances comme sujet d'étude devrait apparaître également.

Néanmoins des directions de recherche sont proposées qui s'appuient sur les résultats énoncés plus haut et qui visent des objectifs à la fois plus rationnels et plus ambitieux :

1. Poursuite de l'intégration des différents mécanismes de l'intelligence artificielle pour réaliser des systèmes multifacettes et en particulier :
 - « – modifier radicalement la nature du processus de programmation de telle façon que le programmeur puisse travailler avec des assistants intelligents facilitant toutes les tâches de réalisation d'un système informatique ;
 - « – supprimer les différences existantes entre les bases de données et les bases de connaissances ;
 - « – supprimer les différences de traitement entre les logiques de description et les systèmes à objet et entre la programmation logique et les systèmes de programmation traditionnelle ⁴¹ ;
 - « – prendre en compte le temps réel et contraint dans les solutions en favorisant des évolutions continues du raisonnement ⁴² ;
 - « – Rendre les ordinateurs plus faciles à utiliser, en particulier en facilitant leur personnalisation.

Cette dernière direction est naturelle pour le RàPC qui par la nature même de son cycle emprunte à plusieurs domaines de l'intelligence artificielle.

2. Construction de robots (physiques et simulés). Ce point est évidemment en rapport avec le précédent, puisqu'il ne s'agit pas d'un domaine particulier mais d'une intégration la plus forte possible de très nombreuses disciplines pour fournir au robot-logiciel ou au robot-matériel des capacités d'adaptation toujours plus grande dans la résolution de problème. Nous considérons que la notion d'adaptation et de remémoration sont étroitement liées et qu'en conséquence le paradigme du RàPC pourrait tout à la fois se nourrir de cette direction et y contribuer en retour.
3. Modéliser la rationalité. Il s'agit de donner un sens « économique » à la rationalité, c'est à dire la comprendre comme tout ce qui contribue à améliorer la cohérence d'un système en termes d'explications, de justifications et de vérifications, à augmenter sa compétence en termes de performances, et à faciliter sa construction en termes de facilités de conception et de développement. C'est bien ce type de rationalité « pragmatique », c'est à dire mesurée à l'aune de la réussite de la tâche, que le RàPC exploite pour résoudre des problèmes. Adopter ce type de rationalité ouvre bien entendu les portes des applications du domaine de l'économie.

41. Ce qui ne veut pas dire qu'il n'y a pas de différences, mais que les environnements de développements doivent présenter les mécanismes typique de l'IA exactement de la même façon que n'importe quel autre mécanisme de programmation.

42. Algorithmes « anytime ».

4. Assister la collaboration. Pour démontrer leur utilité comme assistants, les systèmes d'IA doivent interpréter les « mots » et les « besoins » des utilisateurs de façon à modéliser les souhaits, les intentions et les possibilités de ces personnes. Réaliser ces interprétations signifie souvent se fonder sur des propriétés statistiques de comportements et choisir comment coopérer signifie souvent évaluer et négocier les préférences des différents acteurs. C'est probablement dans cette direction que l'intégration du RàPC peut se révéler un facteur de réussite considérable par sa double capacité à estimer statistiquement (similarité) les comportements et à proposer des adaptations directement liées au contexte. Nos travaux en cours et prévus sont tendus vers l'assistance efficace et en contexte aux tâches difficiles [Prié et al., 1998], [Prié et al., 1999], [Prié et al., 1999] .
5. Améliorer la communication. Un certain nombre de priorités sont dégagées dans cette direction et en particulier :
 - « – introduire les contrôles gestuels et parlés dans les appareils de tous les jours ;
 - « – automatiser la formalisation de la connaissance à partir des livres et autres textes ;
 - « – procurer la traduction simultanée entre plusieurs langues et lecture parlée de documents écrits ou électroniques ;
 - « – autoriser l'utilisation du langage parlé ou du langage écrit pour interroger des « bases de données-connaissances » de grande taille (annuaires, bibliothèques, catalogues, etc.).

Même si le « Memory Based Reasoning » connaît un grand succès pour les applications de traduction [Kitano and Higuchi, 1993], [Kitano, 1993], il n'existe pas à notre connaissance de travaux spécifiquement RàPC permettant des avancées significatives dans ce domaine. Nous pensons que le RàPC intervient dans la qualité de la communication plutôt pour l'assister que pour intervenir directement sur le médium lui-même.

6. Obtenir des connaissances. Il reste là une volonté de connaissances universelles, qui donne à cette direction une ambition considérable. Les auteurs énoncent des priorités :
 - « – construire des encyclopédies de connaissances, d'abord générales, puis capables de couvrir toutes les méthodes et connaissances de l'homme⁴³ ;
 - « – déterminer les moyens les plus efficaces de représenter l'information pour différentes utilisations, et tout en même temps être capable de faire coopérer différents types de connaissances ;
 - « – développer des systèmes d'enseignement automatisés (exploitant éventuellement les encyclopédies construites) qui utilisent les questions et les observations de l'étudiant pour modéliser ses connaissances, ses qualités et son style d'apprentissage et pour adapter la fabrication des leçons et exercices aux besoins de l'étudiant ;
 - « – concevoir des organisations et des entreprises qui s'améliorent en exploitant leur expérience pour apprendre leurs forces et faiblesses cachées ;
 - « – automatiser les domaines les plus routiniers et générateurs massifs de données de la recherche scientifique, statistique, industrielle et commerciale.

43. Nous considérons qu'il s'agit plus d'un moteur pour la recherche qu'une fin en soi.

Cette direction est la plus riche en retombées pour le RàPC, en particulier pour tout ce qui touche à l'exploitation du retour d'expérience.

7. Approfondir les bases théoriques de l'intelligence artificielle. Il s'agit d'asseoir encore mieux les fondements scientifiques formels et en particulier les mathématiques du domaine. Nous avons ici un champ particulièrement ouvert en RàPC pour aller plus loin que la modélisation conceptuelle, vers la modélisation formelle. Nous considérons toutefois que cette recherche nécessitera de gros efforts transdisciplinaires pour réussir.

5.4 Synthèse concernant les axes à développer en raisonnement à partir de cas

L'exercice consistant à se projeter dans l'avenir est toujours périlleux, et situer les axes de recherche selon la grille précédente sera sûrement critiqué. Toutefois, l'expérience de notre équipe au travers de ses différentes contributions, nous incite à penser que l'avenir de la recherche en raisonnement à partir de cas est à chercher dans deux directions principales :

- « – L'exploitation des cas de résolution de problème comme support de validation de l'apprentissage des connaissances dans l'entreprise. L'interconnexion explicite et organisée des cas de résolution et des connaissances partagées dans l'entreprise nous semble un axe de recherche très important pour voir déboucher de nouvelles organisations des systèmes d'information dans l'entreprise.
- « – L'utilisation massive des ordinateurs pour l'accès à des informations et des services, indépendamment de tâches spécifiques préétablies, impose de se poser la question de la plasticité des systèmes informatiques fournisseurs de service aux innombrables façons de les utiliser en fonction de tâches de plus en plus personnalisées. La capitalisation de l'expérience concrète (traces directes d'utilisation), et la mise au point progressive de modèles des tâches (découverts, c'est-à-dire appris semi-automatiquement) associées à cette utilisation concrète est un enjeu considérable pour la recherche de nouveaux types de coopération entre l'homme et la machine.

Références

- [Aamodt and Plaza, 1994] Aamodt, A. and Plaza, E. (1994). Case based reasoning: Foundational issues, methodological variations, and system approaches. *AICOM*, 1:39–59.
- [Aben, 1993] Aben, M. (1993). Formally specifying reusable knowledge model components. *Knowledge Acquisition vol 5*, pages 119–141.
- [Alquier and Tignol, 1998] Alquier, A.-M. and Tignol, M.-H. (1998). Management des connaissances et management par projet. In *Proceedings de IC98, Conférence Ingénierie des Connaissances, Pont-à-Mousson, 1998*, pages 101–119. LORIA-INRIA Lorraine.
- [Armengol and Plaza, 1993] Armengol, E. and Plaza, E. (1993). Case-based reasoning at the knowledge level: An analysis of chef. In *First European Workshop on Case-Based Reasoning, EWCBR'93*, pages 290–295, University of Kaiserslautern, Germany. Lecture Notes in Artificial Intelligence, volume 837, Springer Verlag, Berlin.
- [Aussenac, 1989] Aussenac, N. (1989). *Conception d'une méthodologie et d'un outil pour l'acquisition des connaissances expertes*. Thèse d'université, Université Paul Sabatier, Toulouse.
- [Bareiss et al., 1988] Bareiss, R., Porter, B., and Wier, C. (1988). Protos, an exemplar based learning apprentice. *International Journal of Man-Machines Studies, Vol 29*, pages 549–561.
- [Barletta, 1989] Barletta, R. (1989). Case adaptation in autoclave layout design. In *Workshop on case-based Reasoning, DARPA 89*, pages 203–207. Morgan-Kaufmann, San Mateo.
- [Bartlett, 1932] Bartlett, F. C. (1932). *Remembering: a study in experimental and social psychology*. Cambridge University Press.
- [Bartsch-Spörl, 1995] Bartsch-Spörl, B. (1995). Towards the integration of case-based schema-based and model-based reasoning for supporting complex design tasks. In Veloso, M. and Aamodt, A., editors, *First International Conference on Case-Based Reasoning, ICCBR'95*, LNAI 1010, pages 144–156, Sesimbra. Springer Verlag.
- [Batagelj and Bren, 1995] Batagelj, V. and Bren, M. (1995). Comparing resemblance measures. *Journal of Classification*, 12:73–90.
- [Bento and Costa, 1993] Bento, C. and Costa, E. (1993). A similarity metric for retrieval of cases imperfectly described and explained. In *First European Workshop on Case-Based Reasoning, EWCBR'93*, pages 8–13, University of Kaiserslautern, Germany. Lecture Notes in Artificial Intelligence, volume 837, Springer Verlag, Berlin.
- [Bergmann and Wilke, 1995] Bergmann, R. and Wilke, W. (1995). Building and refining abstract planning cases by change of representation language. *Journal of Artificial Intelligence Research*, 3:53–118.
- [Bichindaritz, 1994] Bichindaritz, I. (1994). *Apprentissage de Concepts dans une mémoire dynamique: raisonnement à partir de cas adaptable à la tâche cognitive*. Thèse d'informatique, Université René Descartes Paris V, France.
- [Bichindaritz et al., 1997] Bichindaritz, I., Bradshaw, J., and Sullivan, K. (1997). Case-based learning: Learning as reuse. prépublication, Fred Hutchinson Cancer Research Center, MSI Université de Limoges.
- [Bisson, 1994] Bisson, G. (1994). Définition de la similarité dans les modèles à objet. In *Actes de LMO94, Grenoble*, pages 70–86.
- [Bonnet et al., 1986] Bonnet, C., Hoc, J., and Tiberghien, G. (1986). *Intelligence Artificielle, Automatique et Psychologie*. Edition Mardaga.

- [Boucard, 1996] Boucard, F. (1996). Etude et implantation d'une mémoire de cas distribuée sur un réseau d'ordinateurs. Rapport de DEA, DEA d'Informatique de Lyon, UCB/ENS/INSA, Lyon.
- [Brachman and Schmolze, 1985] Brachman, R. J. and Schmolze, J. G. (1985). An overview of the KL-ONE representation system. *Cognitive Science*, 9:171–216.
- [BRESLOW and AHA, 1997] BRESLOW, L. and AHA, D. (1997). Simplifying decision trees: a survey. *Knowledge Engineering Review*, 12(1):1–40.
- [Brooks, 1991] Brooks, R. (1991). Intelligence without representation. *Artificial Intelligence*, (47):139–159.
- [Brown, 1993] Brown, M. G. (1993). An under-lying memory model to support case retrieval. In *First European Workshop on Case-Based Reasoning, EWCBR'93*, pages 20–25, University of Kaiserslautern, Germany. Lecture Notes in Artificial Intelligence, volume 837, Springer Verlag, Berlin.
- [Brunet and Ermine, 1994] Brunet, E. and Ermine, J. L. (1994). Problématique de la gestion des connaissances des organisations. *Ingénierie de Systèmes d'Information*, 2(3):263–291.
- [Chandrasekaran, 1986] Chandrasekaran, B. (1986). Generic tasks in knowledge based reasoning: High level building blocks for expert system design. *IEEE Expert*, 1(3):23–29.
- [Chandrasekaran et al., 1998] Chandrasekaran, B., Josephson, J., and Richard Benjamins, V. (1998). The ontology of tasks and methods. In <http://cpsc.ucalgary.ca/KAW/KAW98>, editor, *Proceedings of the 11th Workshop on Knowledge Acquisition, Modelling and Management*.
- [Charlet, 1992] Charlet, J. (1992). ACTE: Acquisition des connaissances par interprétation d'un modale causal. *Revue d'intelligence artificielle Vol6 N1-2*, 6(1-2):99–129.
- [Chein and Mugnier, 1992] Chein, M. and Mugnier, M.-L. (1992). Conceptual graphs: Fundamental notions. *Revue d'Intelligence Artificielle*, 6(4):365–406.
- [Chi et al., 1988] Chi, M., Glaser, R., and Far, M., editors (1988). *The nature of Expertise*. Lawrence Erlbaum Associates.
- [Cohen and Hirsh, 1994] Cohen, W. and Hirsh, H. (1994). Learning the classic description logic: Theoretical and experimental results. In *Proceedings of the International Conference on Knowledge Representation and Reasoning, 1994*, pages 124–133.
- [Converse et al., 1989] Converse, T., Hammond, K., and Marks, M. (1989). Learning modification rules from expectation failure. In *Workshop on case-based Reasoning, DARPA 89*, pages 110–114. Morgan-Kaufmann, San Mateo.
- [Corvaisier et al., 1998] Corvaisier, F., Mille, A., and Pinon, J.-M. (1998). Radix 2, assistance à la recherche d'information documentaire sur le web. In *IC'98, Ingénierie des Connaissances, Pont-à-Mousson, France*, pages 153–163. INRIA-LORIA, Nancy.
- [Costecalde, 1984] Costecalde, A. (1984). Suivi de qualité de fabrication par analyse des données. *Revue AFCEQ*, XX(1):31–36.
- [Coulon et al., 1990] Coulon, D., Boisvieux, J.-F., Bourelly, L., Bruneau, L., Chouraqui, E., David, J.-M., Lu, C.-R., Py, M., Savelli, J., Seroussi, B., and Vrain, C. (1990). Le raisonnement par analogie en intelligence artificielle, formalisation, applications. In *Actes des troisièmes journées nationales du PRC-GDR-IA*, pages 45–88. Hermès.
- [Coupey and Salotti, 1996] Coupey, P. and Salotti, S. (1996). Using a description logic with default and exception to formalize the retrieval process of cbr. In *International*

IPMU'96 Congress, Information Processing and Management of Uncertainty, Granada, Spain.

- [Crevier, 1993] Crevier, D. (1993). *AI, The tumultuous history of the search for Artificial Intelligence*. BasicBooks, Harper-Collins.
- [Critchley and Cutsem, 1992] Critchley, F. and Cutsem, B. V. (1992). Representations of dissimilarities: A survey and some new results. In *Distancia 92, Rennes France, 22-26 Juin 1992*, pages 41–46.
- [Diday et al., 1982] Diday, E., Lemaire, J., Pouget, J., and Testu, F. (1982). *Éléments d'analyse de données*. Dunod Editeur.
- [Doyle and Dean, 1996] Doyle, J. and Dean, T. (1996). Strategic directions in artificial intelligence. *ACM Computing Surveys*, 28(4).
- [Dubois et al., 1997] Dubois, D., Esteva, F., Garcia, P., Godo, L., López De Mantaras, R., and Prade, H. (1997). Fuzzy set-based models in case-based reasoning. In Leake, D. and Plaza, E., editors, *2nd International Conference on Case-Based Reasoning (ICCBR'97)*, number 1266 in Lecture Notes in Artificial Intelligence, pages 599–610, Providence, Rhode Island, USA. Springer-Verlag, Berlin, Germany.
- [Ducourneau and Quinqueton, 1986] Ducourneau, R. and Quinqueton, J. (1986). YA-FOOL : Yet Another Frame-Based Object Oriented Language. Technical report, INRIA. Rapport Technique No 72.
- [Ellman, 1989] Ellman, T. (1989). Explanation based learning: A survey of programs and perspectives. *ACM Computing Survey*, 21(2):163–221.
- [Erikson, 1994] Erikson, H. (1994). Models for knowledge acquisition tool design. *Knowledge Acquisition*, 6:47–74.
- [FARON and GANASCIA, 1997] FARON, C. and GANASCIA, J.-G. (1997). Representation of defaults and exceptions in conceptual graphs formalism. In Lukose, D., Delugach, H., Keeler, M., Searle, L., and Sowa, J., editors, *Conceptual Structures: Fulfilling Peirce's Dream. Fifth International Conference on Conceptual Structures, ICCS'97, 3-8 Aug. 1997, Seattle, WA, USA*, pages 153–167, Springer-Verlag Berlin, Germany.
- [Ferber, 1989] Ferber, J. (1989). *Objets et agents : une étude des structures de représentation et de communication en intelligence artificielle*. Thèse de Doctorat d'Etat, Université Pierre et Marie Curie, Paris 6.
- [Fikes and Kehler, 1985] Fikes, R. and Kehler, T. (1985). The Role of Frame-Based Representation in Reasoning. *Communications of ACM*, 28(9):904–920.
- [Firlej and Hellens, 1991] Firlej, M. and Hellens, D. (1991). *Knowledge Elicitation, A practical Handbook*. Prentice Hall International.
- [Fox and Leake, 1995] Fox, S. and Leake, D. B. (1995). Using introspective reasoning to refine indexing. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 391–397. Morgan Kaufmann.
- [Fuchs, 1997] Fuchs, B. (1997). *Raisonnement à partir de cas : Représentation des connaissances et du raisonnement basé sur l'expérience. Application à la supervision industrielle*. Thèse en informatique, CPE-Lyon et Université Jean Monnet, Saint-Etienne, France.
- [Fuchs et al., 1999] Fuchs, B., Lieber, J., Mille, A., and Napoli, A. (1999). Vers une théorie unifiée de l'adaptation en raisonnement à partir de cas. In *Actes de IC'99, Palaiseau, Juin 99*, pages 199–207.
- [Gaines and Shaw, 1998] Gaines, B. R. and Shaw, M. L. (1998). Developing for WEB integration in Sisyphus-IV: Webgrid experience. In <http://cpsc.ucalgary.ca/KAW/KAW98>,

- editor, *Proceedings of the 11th Workshop on Knowledge Acquisition, Modelling and Management, 1998*.
- [GDR-PRC-IA, 1997] GDR-PRC-IA (1997). Article de synthèse: Décision et représentation des connaissances. In Pesty, S. and Siegel, P., editors, *Actes des journées nationales du PRC-GDR Intelligence Artificielle*, pages 211–242, Grenoble. Hermès, Paris.
- [Gentner, 1989] Gentner, D. (1989). Finding the needle: Accessing and reasoning from prior cases. In *Workshop on case-based Reasoning, DARPA 89*, pages 137–142. Morgan-Kaufmann, San Mateo.
- [Gentner and Stevens, 1983] Gentner, D. and Stevens, A., editors (1983). *Mental Models*. Erlbaum, Hillsdale, N.J.
- [Hammond, 1986] Hammond, K. (1986). Chef: a model of case based planning. In *AAAI-86, Morgan Kaufmann*, pages 267–271.
- [Hanks and Weld, 1995] Hanks, S. and Weld, D. S. (1995). A domain independant algorithm for plan adaptation. *Journal of Artificial Intelligence Research*, (2):319–360.
- [Hanney et al., 1995] Hanney, K., Keane, M., Smyth, B., and Cunningham, P. (1995). Systems, tasks and adaptation knowledge: revealing some revealing dependencies. In Veloso, M. and Aamodt, A., editors, *First International Conference on Case-Based Reasoning, ICCBR'95*, LNAI 1010, pages 461–470, Sesimbra, Portugal. Springer Verlag, Berlin.
- [Hanney and Keane, 1997] Hanney, K. and Keane, M. T. (1997). The adaptation knowledge bottleneck: how to ease it by learning from cases. In Leake, D. and Plaza, E., editors, *Second International Conference on Case-Based Reasoning, ICCBR'97*, pages 359–370. Springer-Verlag Berlin, Germany.
- [Haton et al., 1991] Haton, J. P., Bouzid, N., Charpillet, F., Haton, M.-C., Laasri, B., Laasri, H., Marquis, P., Mondot, T., and Napoli, A. (1991). *Le raisonnement en intelligence artificielle, Modèles, techniques et architectures pour les systèmes à base de connaissance*. Interditions Paris, série iia.
- [Herbeaux and Mille, 1998] Herbeaux, O. and Mille, A. (1998). Accelere: a case-based design assistant for closed cell rubber industry. In Milne, R., Macintosh, A. M., and Bramer, M., editors, *Proceedings of ES98, the Eighteenth Annual International Conference of the British Computer Society, Specialist Group on Expert Systems*, pages 69–82. BCS Conference Series, Springer-Verlag, London.
- [Hinrichs, 1989] Hinrichs, T. R. (1989). Strategies for adaptation and recovery in design problem solver. In *Workshop on case-based Reasoning, DARPA 89*, pages 115–118. Morgan-Kaufmann, San Mateo.
- [Hoc and Nguyen-Xuan, 1986] Hoc, J. M. and Nguyen-Xuan, A. (1986). Les modèles informatiques de la résolution de problème. In *PSYCHOLOGIE, dirigée par Jean Piaget, Pierre Mounoud et Jean Paul Bronckart*, pages 1712–1756. Encyclopédie de la Pléiade.
- [Houdé et al., 1998] Houdé, O., Kayser, D., Koenig, O., Proust, J., and Rastier, F. (1998). *Vocabulaire de sciences cognitives*. Psychologie et sciences de la pensée. PUF Paris.
- [Jaczynski and Trousse, 1997] Jaczynski, M. and Trousse, B. (1997). A framework for the management of past experiences with time-extended situations. In *Proceedings of the 6th International Conference on Information and Knowledge Management (CIKM'97)*, pages 32–39, Las Vegas, Nevada, USA.
- [Jaczynski and Trousse., 1997] Jaczynski, M. and Trousse., B. (1997). A world wide web browsing advisor reusing past navigations from a group of users. In *Proceedings of the third UK Case-Based Reasoning Workshop (UKCBR'93)*, Manchester, UK.

- [Janetzko et al., 1993] Janetzko, D., Berner, K., Coulon, C.-H., and Hovestadt, L. (1993). Toward a task oriented methodology in knowledge acquisition and system design in CBR, 1993. In *First European Workshop on Case-Based Reasoning, EWCBR'93*, pages 360–365, University of Kaiserslautern, Germany. Lecture Notes in Artificial Intelligence, volume 837, Springer Verlag, Berlin.
- [Kass, 1989] Kass, A. (1989). Strategies for adapting explanations. In *Workshop on case-based Reasoning, DARPA 89*, pages 119–123. Morgan-Kaufmann, San Mateo.
- [Kayser, 1997] Kayser, D. (1997). *La représentation des connaissances*. Hermes, Paris.
- [Kettler et al., 1994] Kettler, B. P., Handler, J. A., Anderson, W. A., and Evett, M. P. (1994). Massively parallel support for case-based planning. *IEEE Expert*, 2:8–14.
- [Kirsh, 1991] Kirsh, D. (1991). Foundations of AI: the big issues. *Artificial Intelligence*, (47):3–30.
- [Kitano, 1993] Kitano, H. (1993). Challenges of massive parallelism. Invited paper. In *IJCAI93, Chambéry, France*, pages 813–834.
- [Kitano and Higuchi, 1993] Kitano, H. and Higuchi, T. (1993). High performance memory based translation on ixm2 massively parallel associative memory processor. Technical report, Thinking Machines Inc.
- [Kitano and Yasunaga, 1992] Kitano, H. and Yasunaga, M. (1992). Wafer scale integration for massively parallel memory based reasoning. In *AAAI92*, pages 851–856.
- [Koehler, 1994] Koehler, J. (1994). An application of terminological logics to case-based reasoning. In Doyle, J., Sandewall, E., and Torasso, P., editors, *Proceedings of the 4th International Conference on Principles of Knowledge and Reasoning*. Morgan Kaufmann, San Francisco.
- [Koelher, 1996] Koelher, J. (1996). Planning from second principles. *Artificial Intelligence*, (87):145–186.
- [Kolodner, 1993] Kolodner, J. (1993). *Case Based Reasoning*. Morgan Kaufmann Publishers.
- [Kolodner, 1988] Kolodner, J. L., editor (1988). *Workshop on case-based Reasoning, DARPA 88*, Clearwater, Florida. Morgan-Kaufmann, San Mateo.
- [Krivine and David, 1991] Krivine, J. and David, J. (1991). L'acquisition des connaissances vue comme un processus de modélisation: méthodes et outils. *Intellectica*, 2(12):101–137.
- [Lang and Perny, 1997] Lang, J. and Perny, P. (1997). Dossier décision et intelligence artificielle. Bulletin de l'AFIA.
- [Leake et al., 1997a] Leake, D. B., Kinley, A., and Wilson, D. (1997a). Case-based similarity assessment: Estimating adaptability from experience. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*. AAAI Press, Menlo Park, CA.
- [Leake et al., 1997b] Leake, D. B., Kinley, A., and Wilson, D. (1997b). Learning to integrate multiple knowledge sources for case-based reasoning. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*. Morgan Kaufmann.
- [Lieber, 1997] Lieber, J. (1997). *Raisonnement à partir de cas et classification hiérarchique. Application à la planification de synthèse en chimie organique*. Thèse d'université, Université Henri Poincaré Nancy 1.
- [Lieber and Napoli, 1996] Lieber, J. and Napoli, A. (1996). Adaptation of synthesis plans in organic chemistry. In Voß, A., Bergmann, R., and Bartsch-Spörl, B., editors, *Working*

- Notes of the ECAI-96 Workshop on Adaptation in Case-Based Reasoning*, pages 132–136, Budapest, Hungary.
- [Maher et al., 1995] Maher, M. L., Balachandran, M. B., and Zhang, D. M. (1995). *Case-based reasoning in design*. Lawrence Erlbaum Associates.
- [Malek, 1996] Malek, M. (1996). *Un modèle hybride de mémoire pour le raisonnement à partir de cas*. Thèse d'université, Université Joseph Fourier, Grenoble.
- [Masini et al., 1989] Masini, G., Napoli, A., Colnet, D., Leonard, D., and Tombre, K. (1989). *Les langages à objet*, chapter 8, pages 275–320. InterEditions.
- [Maurer and Dellen, 1998] Maurer, F. and Dellen, B. (1998). A concept for an internet-based process-oriented knowledge management environment. In <http://cpsc.ucalgary.ca/KAW/KAW98>, editor, *Proceedings of the 11th Workshop on Knowledge Acquisition, Modelling and Management, 1998*.
- [Mayer, 1992] Mayer, R. E. (1992). *Thinking, Problem solving, Cognition*. Psychologie. W.H. Freeman and Co, New-York.
- [Mignot, 1997] Mignot, H. (1997). *Sensibilité au contexte lors de l'évaluation de la similarité dans une architecture ouverte pour le raisonnement à partir de cas*. Thèse en informatique, Université Paris-Sud, Paris 11, Orsay.
- [Mille, 1995] Mille, A. (1995). Modèle conceptuel du raisonnement basé sur les cas. In Bichindaritz, I., editor, *4ème Séminaire Français sur le Raisonnement à Partir de Cas, Paris*, pages 40–52, Université René Descartes (Paris V).
- [Mille et al., 1995] Mille, A., Di-Martino, J.-L., and Michel, A. (1995). Adaptation: the key-point in case based reasoning. A case study: Digester programming helping. présenté au «Workshop on practical developments strategies for industrial strength Case Based Reasoning applications, 16th International Conference on Artificial Intelligence, IJCAI'95, Montreal, Canada».
- [Mille et al., 1999] Mille, A., Fuchs, B., and Chiron, B. (1999). Raisonnement fondé sur l'expérience: un nouveau paradigme en supervision industrielle. *Revue d'intelligence artificielle*, 13:97–128.
- [Mille et al., 1996] Mille, A., Fuchs, B., and Herbeaux, O. (1996). A unifying framework for adaptation in case based reasoning. In Voß, A., Bergmann, R., and Bartsch-Spörl, B., editors, *Workshop on Adaptation in Case-Based Reasoning, 12th European Conference on Artificial Intelligence*, pages 22–28, Budapest, Hungary. Springer Verlag, Berlin.
- [Mille and Napoli, 1997] Mille, A. and Napoli, A. (1997). Aspects du raisonnement à partir de cas. In *6èmes Journées Nationales du PRC-GDR Intelligence Artificielle*, pages 261–287. MENESR and CNRS, Hermès, Paris.
- [Mille and Pialot, 1994] Mille, A. and Pialot, D. (1994). Un système expert pour le contrôle de qualité en production industrielle. Rapport interne 94 1, CPE-Lyon.
- [Mille et al., 1994] Mille, A., Pialot, D., and Michel, A. (1994). An expert system for predictive quality control in industrial production: focusing on the knowledge acquisition process with several experts. Rapport interne 94 2, CPE-Lyon.
- [Minsky, 1975] Minsky, M. (1975). A framework for representing knowledge. In *Readings in Knowledge Representation*, pages 245–262. Morgan Kaufmann Publishers (1985).
- [Napoli, 1997] Napoli, A. (1997). Une introduction aux logiques de descriptions. Edité par INRIA, BP105, 78153 Le Chesnay, France, <http://www.inria.fr>.
- [Napoli et al., 1996] Napoli, A., Lieber, J., and Curien, R. (1996). A formal analysis of case-based reasoning in an object-based representation context. In Faltings, B. and

- Smith, I., editors, *Third European Workshop on Case-Based Reasoning, EWCBR'96*, volume Lecture Notes in Artificial Intelligence 1168, pages 295–308, Lausanne,. Springer-Verlag, Berlin.
- [Newell, 1982] Newell, A. (1982). The knowledge level. *Artificial Intelligence*, (18):87–127.
- [Newell, 1990] Newell, A. (1990). *Unified Theories of Cognition*. Harvard University Press, Cambridge, MA.
- [O'hara and Shalbolt, 1993] O'hara, K. and Shalbolt, N. (1993). Locating generic tasks. *Knowledge Acquisition*, 5:449–481.
- [Petrushin and Kolodner, 1996] Petrushin, V. and Kolodner, J. (1996). Put your experience on the web: a tool for creating and browsing in case libraries. In Maurer, H., editor, *WebNet96 - World Conference of the Web Society. Proceedings, San Francisco, CA, USA*, pages 566–573. Assoc. Adv. Comput. Educ Charlottesville, VA, USA.
- [Pialot, 1990] Pialot, D. (1990). Analyse multidimensionnelle résultats d'essais de cuisson d'un gateau. Propositions pour la mise en oeuvre de cartes de contrôle multiparamétriques. In *Actes Du Colloque « Méthodes Statistiques Et Agro-Industrie »*, Angers.
- [Pitrat, 1990] Pitrat, J. (1990). *Métaconnaissance, Futur De L'intelligence Artificielle*. HERMÈS.
- [Porter et al., 1990] Porter, B. W., Bareis, R., and Holte, R. C. (1990). Concept learning and heuristic classification in weak-theory domains. *Artificial Intelligence*, 45(3):229–263.
- [Prié et al., 1998] Prié, Y., Mille, A., and Pinon, J.-M. (1998). A user-centered model for content-based description and retrieval of audiovisual sequences. In *Amcp'98, First Int. Advanced Multimedia Content Processing Conf.*, pages 59–65, Osaka, Japon.
- [Prié et al., 1999] Prié, Y., Mille, A., and Pinon, J.-M. (1999). A context-based audiovisual representation model for audiovisual information systems. In *International and Interdisciplinary Conference on Modeling and using Context*, volume 1688 of *LNAI*, Springer-Verlag., pages 296–309, Trento.
- [Prié et al., 1999] Prié, Y., Mille, A., and Pinon, J.-M. (1999). Modèle d'utilisation et modèles de tâches pour l'assistance à l'utilisateur fondée sur l'expérience: le cas d'un système d'information audiovisuelle. In *Actes de IC'99, Palaiseau, Juin 99*, pages 21–29.
- [Purvis and Pu, 1995] Purvis, L. and Pu, P. (1995). Adaptation Using Constraint Satisfaction Techniques. In Veloso, M. and Aamodt, A., editors, *Case-Based Reasoning Research And Development. PROceedings Of The FIrst INternational CONference On CAse-Based REasoning - Iccbr-95*, pages 289–300, Sesimbra, Portugal. LECTure NOTes In ARTificial INTelligence, Volume 1010, SPRinger VERlag, Berlin.
- [Py, 1994] Py, M. (1994). Un modèle conceptuel de raisonnement par analogie. *Revue D'intelligence Artificielle*, 8(1):63–99.
- [Quillian, 1969] Quillian, M. (1969). The teachable language comprehender. *Communications Of The Acm*, 12:459–476.
- [Rechenmann, 1985] Rechenmann, F. (1985). Shirka: Mécanismes d'inférence sur une base de connaissances centrée objet. In *Actes De La 5ème Conférence Reconnaissance De Formes Et Intelligence Artificielle*, pages 1243–1254, Grenoble, France.
- [Reitman-OLson and Rueter, 1987] Reitman-OLson, J. and Rueter, H. H. (1987). Extracting expertise from experts: Methods for knowledge acquisition. *Expert Systems*, 4(3):152–168.
- [Richard, 1984] Richard, J. F. (1984). *Résoudre Des Problèmes Au Laboratoire, à L'école, Au Travail*. Série Psychologie Française, Armand Colin Editeur, Paris.

- [Richard, 1990] Richard, J. F. (1990). *Les Activités Mentales. Comprendre, Raisonner, Trouver Des Solutions*. Série Psychologie Française, Armand Colin Editeur, Paris.
- [Richter, 1995a] Richter, M. M. (1995a). Knowledge contained in similarity measures, invited paper. In Veloso, M. and Aamodt, A., editors, *First International Conference On Case-Based Reasoning, Icbr'95*, Lnai 1010, Sesimbra. Springer Verlag.
- [Richter, 1995b] Richter, M. M. (1995b). On the notion of similarity in case-based reasoning. In Della Riccia, editor, *Proceedings Of The Issek 94 Conference On Mathematical And Statistical Methods In Artificial Intelligence*, pages 171–184, Wien, Austria.
- [Richter et al., 1993] Richter, M. M., Wess, S., Althoff, K.-D., and Maurer, F., editors (1993). *First European Workshop on Case-based Reasoning*, University of Kaiserslautern, Germany. Lecture Notes in Artificial Intelligence, volume 837, Springer Verlag, Berlin.
- [Rifqi, 1996] Rifqi, M. (1996). *Mesures De Comparaison, Typicalité Et Classifications D'objets Flous : Théorie Et Pratique*. Thèse en informatique, Université Pierre Et Marie Curie, Paris 6.
- [Ross, 1987] Ross, B. H. (1987). This is like that: The use of earlier problems and the separation of similarity effects. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, (13):629–639.
- [Rougegrez-Loriette, 1994] Rougegrez-Loriette, S. (1994). *Prédiction De Processus à Partir De Cas Observés: Le Système Rebecas*. Thèse en informatique, Université De Paris Vi, France.
- [Rudolph, 1997] Rudolph, S. (1997). On the foundations and applications of similarity theory to case-based reasoning. In Adey, R. and Rzevski, G., editors, *Proceedings Of The 12th International Conference For Applications Of Artificial Intelligence In Engineering Aieng'97*, Capri, Italy. Ashurst Lodge, Southampton, Uk.
- [Salotti and Ventos, 1997] Salotti, S. and Ventos, V. (1997). Etude et formalisation d'un système de raisonnement à partir de cas dans une logique de description. In *Journées INgénierie Des CONnaissances Et APprentissage AUTomatique - Jicaa'97*, pages 629–642, ROscoff, France.
- [Schank, 1982] Schank, R. C. (1982). *Dynamic Memory: A Theory Of Reminding And Learning In Computers And People*. Cambridge University Press.
- [SCHANK, 1996] SCHANK, R. C. (1996). A goal-based scenario for high school students. *CACM*, (4):28–29.
- [Schank and Abelson, 1977] Schank, R. C. and Abelson, R. P. (1977). *Scripts, plans, goals and understanding*. Erlbaum, Hillsdale, NJ.
- [Schank and Riesbeck, 1989] Schank, R. C. and Riesbeck, C. K. (1989). *Inside Case Based Reasoning*. Lawrence Erlbaum Associates Publishers, Hillsdale, New Jersey, Usa.
- [Schreiber et al., 1993] Schreiber, A., Wielinga, B., and Breuker, J., editors (1993). *Kads: A Principled Approach To Knowledge-Based System Development*, volume 11 of *Knowledge-Based Systems*. Academic Press, London.
- [Sebag and Shoemaker, 1993] Sebag, M. and Shoemaker, M. (1993). A rule based similarity measure. In *First European Workshop On Case-Based Reasoning, Ewbr'93*, pages 65–70, University Of Kaiserslautern, Germany. LECTURE NOTES IN ARTIFICIAL INTELLIGENCE, Volume 837, SPringer VERlag, Berlin.
- [Seroussi and Morice, 1993] Seroussi, B. and Morice, V. (1993). Problem solving expertise acquisition: A computational model. *Revue D'intelligence Artificielle*, 7(2):223–273.

- [Shimazu et al., 1993] Shimazu, H., Kitano, H., and Shibata, A. (1993). Retrieving cases from relational data-bases: Another stride towards corporate wide case-base systems. In *13th International Joint Conference On Artificial Intelligence, Chambéry, France 28-08/03-09 1993*, pages 909–914.
- [Simon, 1998] Simon, G. (1998). Aide à la conception et à la validation de mémoire d'entreprise. In *Proceedings De Ic98, Conférence Ingénierie Des Connaissances, Pont-à-Mousson*, pages 267–275. Loria-Inria Nancy.
- [Slade, 1991] Slade, S. (1991). Case based reasoning: A research paradigm. *Ai Magazine, Spring 1991*, pages 42–55.
- [Smith et al., 1995] Smith, I., Lottaz, C., and Faltings, B. (1995). Spatial composition using case: Idiom. In Veloso, M. and Aamodt, A., editors, *Case-Based Reasoning Research And Development, Iccbr'95*, pages 88–97, Sesimbra (Portugal).
- [Smyth and Cunningham, 1993] Smyth, B. and Cunningham, P. (1993). Complexity of adaptation in real world case-based reasoning systems. In *The Sixth Irish Conference On Artificial Intelligence And Cognitive Science*, pages 228–240.
- [Smyth and Keane, 1995] Smyth, B. and Keane, M. T. (1995). Remembering to forget. a competence-preserving case deletion policy for case-based reasoning systems. In Mellish, C., editor, *Ijcai-95. Proceedings Of The Fourteenth International Joint Conference On Artificial Intelligence*, volume 1, pages 377–382. Morgan Kaufmann Publishers San Mateo, Ca, Usa.
- [Sowa, 1991] Sowa, J. F., editor (1991). *Principles Of Semantic Networks : Explorations In The Representation Of Knowledge*. Morgan Kaufmann, San Mateo, California.
- [Stanfill and Waltz, 1986] Stanfill, C. and Waltz, D. (1986). Toward memory based reasoning. *Communications Of The Acm, Vol 29, No 12*, pages 1213–1228.
- [Steels, 1991] Steels, L. (1991). Commet: A component methodology for knowledge engineering. Technical report, Work Package 2/3/4 Du Projet Esprit Construct.
- [Stroulia and Goel, 1998] Stroulia, E. and Goel, A. (1998). Evaluation of a Psm specification language in adaptive design: The autognostic experiments. In [Http://Cpsc.Ucalgary.Ca/Kaw/Kaw98](http://Cpsc.Ucalgary.Ca/Kaw/Kaw98), editor, *Proceedings Of The 11th Workshop On Knowledge Acquisition, Modelling And Management*.
- [Thayse, 1991] Thayse, A. (1989–1991). *Approche Logique De L'intelligence Artificielle.*, volume 1–4. Dunod Informatique, Paris.
- [Thompson, 1997] Thompson, V. (1997). Corporate memories. *Byte, International Edition*, 32is:7–12.
- [Turner, 1989] Turner, R. M. (1989). Case-based and schema-based reasoning for problem solving. In *Workshop on case-based Reasoning, DARPA 89*, pages 341–344. Morgan-Kaufmann, San Mateo.
- [Tversky, 1977] Tversky, A. (1977). Features of similarity. *Psychological Review* 84(4), pages 327–352.
- [Van De Velde and Aamodt, 1994] Van De Velde, W. and Aamodt, A. (1994). Machine learning issues in commonkads. Technical report, Document Esprit: Project P5248, KADS II/TH.4.3/TR/VUB/002/3.0.
- [Van Marke, 1988] Van Marke, K. (1988). KRS: An Object Oriented Representation Language. *Revue d'Intelligence Artificielle*, 1(4):43–68.
- [Veloso, 1997] Veloso, M. (1997). Merge strategies for multiple case plan replay. In Leake, D. and Plaza, E., editors, *Second International Conference on Case-Based Reasoning, ICCBR '97*, pages 413–424. Springer-Verlag, Berlin, Germany.

- [Velo and Aamodt, 1995] Veloso, M. and Aamodt, A., editors (1995). *First International Conference on Case-Based Reasoning, ICCBR'95*, volume 1010 of *Lecture Notes in Artificial Intelligence*, Sesimbra, Portugal. Springer-Verlag, Berlin, Germany.
- [Visser, 1994] Visser, W. (1994). Organisation of design activities: opportunistic, with hierarchical episodes. *Interacting with Computers*, 6(3):239–274.
- [Vogel, 1988] Vogel, C. (1988). *Génie Cognitif*. Masson, Paris.
- [Von Neumann and Morgenstern, 1947] Von Neumann, J. and Morgenstern, O. (1947). *Theory of games and economic behaviour*. Princeton University Press.
- [Voß, 1996a] Voß, A. (1996a). How to solve complex problems with cases. *Engineering applications of artificial intelligence*, 9(4):377–384.
- [Voß, 1996b] Voß, A. (1996b). Structural Adaptation with TOPO. In Voß, A., Bergmann, R., and Bartsch-Spörl, B., editors, *Workshop on Adaptation in Case-Based Reasoning, ECAI-96*, Budapest, Hungary.
- [Voß et al., 1996] Voß, A., Bergmann, R., and Bartsch-Spörl, B., editors (1996). *Workshop on Adaptation in Case-Based Reasoning, ECAI'96*, Budapest, Hungary. John Wiley & Sons Ltd.
- [Watson, 1997] Watson, I. (1997). *Applying Case-Based Reasoning: Techniques for Enterprise Systems*. Morgan Kaufmann Publishers.
- [Weil-Barais et al., 1993] Weil-Barais, A., Dubois, D., Lecocq, P., Pedinielli, J. L., and Streri, A. (1993). *L'homme cognitif*. Collection Premier cycle, PUF, Paris.
- [Wilke and Bergmann, 1998] Wilke, W. and Bergmann, R. (1998). Techniques and knowledge used for adaptation during case-based problem solving. In ??, editor, *à paraître dans les actes de IEA '98*, page????
- [Winston, 1988] Winston, P. H. (1988). *Intelligence Artificielle*. Addison Wesley Europe, InterEditions, Paris.
- [Yager, 1996] Yager, R. (1996). *Fuzzy logic foundations and industrial applications*, chapter A unified view of case-based reasoning and fuzzy modeling, pages 5–26. Kluwer Academic Press, Norwell, MA, USA.
- [Yager, 1997] Yager, R. (1997). Case based reasoning, fuzzy systems modeling and solution composition. In Leake, D. and Plaza, E., editors, *Case-Based Reasoning Research and Development. Second International Conference on Case-Based Reasoning, ICCBR-97, Providence, RI, USA, 25-27 July 1997*, pages 633–642. Springer-Verlag Berlin, Germany.
- [Zaff et al., 1993] Zaff, B. S., Mcneese, M. D., and Snyder, D. E. (1993). Capturing multiple perspectives: a user-centered approach to knowledge and design acquisition. *Knowledge Acquisition*, 5:79–116.