



National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

**Revision d'une théorie du domaine pour des cas
marginaux en employant une analyse de la preuve
et des exemples prototypiques**

par

Boris Plante

Thèse déposée à
l'École des Études Supérieures et de la Recherche
en vue de l'obtention de la
maîtrise ès sciences appliquées en informatique

Université d'Ottawa
Ottawa, Ontario, Canada
Janvier 1991



National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-68056-3

Canada



UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

Je déclare être le seul auteur de cette thèse.

J'autorise l'Université d'Ottawa à prêter cette thèse à d'autres institutions ou individus, dans le but de recherches académiques.

Boris Plante

De plus, j'autorise l'Université d'Ottawa à reproduire cette thèse par photocopies ou autrement, en sa totalité ou en parties, à la demande d'autres institutions ou individus dans le but de recherches académiques.

Boris Plante

Remerciements

Stan Matwin, mon directeur de thèse, m'a supporté, aidé et bien compris tant au niveau professionnel, intellectuel et humain. Bertrand Pelletier a lu cette thèse et est responsable de nombreuses améliorations au niveau du style et de la présentation. Robert Holte m'a permis de clarifier mes idées lors de nombreuses discussions.

Merci à tous.

Boris Plante

Note

Une implémentation en Prolog couvrant presque la totalité de la théorie contenue dans cette thèse fut présentée aux deux arbitres lors de la défense de la thèse.

Table des matières

INTRODUCTION.....	1
1 PRÉSENTATION DE LA RECHERCHE.....	4
1.1 But des méthodes de base en apprentissage automatique.....	4
1.2 Le but de notre recherche.....	10
1.3 La pertinence de la recherche	11
1.4 Survol du système et limitations.....	13
1.5 Discussion des résultats	14
1.6 Contribution de la thèse	14
1.6.1 Classification des prédicats.....	14
1.6.2 Utilisation des prototypes comme guide à l'innovation (pas inductif)...	15
1.6.3 Une méthode adaptative.....	15
1.6.4 Une méthode indépendante de la théorie.....	15
2 APPRENTISSAGE AUTOMATIQUE: un survol	16
2.1 Limitations des deux méthodes principales en apprentissage	16
2.2 Révision et/ou reformulation de la théorie du domaine	18
2.3 Apprentissage mixte	18
2.3.1 Hybridation de EBL et SBL.....	18
2.3.2 But des méthodes mixtes	19
2.4 Problèmes en apprentissage.....	20

3 LES MARGINAUX	22
3.1 Définition.....	22
3.2 Où sont-ils?	22
3.2.1 Programme d'apprentissage inductif (inductive learner).....	22
3.2.2 Programme d'apprentissage déductif (deductive learner).....	23
3.3 Représentation des connaissances à deux paliers.....	24
3.3.1 Caractéristiques d'une représentation à deux paliers	25
3.3.2 Genre de règles dans le deuxième palier	26
3.4 Classification des exemples pendant l'apprentissage	26
3.5 Marginaux et les deux modèles en apprentissage.....	27
3.6 Heuristiques générales pour traiter les marginaux.....	30
3.7 LAP et les marginaux.....	30
3.7.1 Une méthode mixte pour aider un expert à réviser une théorie.....	31
3.7.2 Un exemple simple pour expliquer notre méthode: la tasse.....	32
3.7.3 Le système LAP	35
4 PRÉDICATS CLASSIFIÉS SELON LA PREUVE: PCP	37
4.1 Nomenclature.....	37
4.2 Prédicats catégorisés par la preuve (PCP).....	38
4.2.1 Définitions.....	38
4.2.2 Relations entre les ensembles lors d'une réussite de la preuve.....	39
4.2.3 Relations entre les ensembles lors d'un échec de la preuve.....	40

4.2.4 Observations	41
4.3 Analyse lors de la déduction	44
4.3.1 Analyse lors d'un échec de la preuve.....	44
4.3.2 Analyse lors d'une réussite de la preuve.....	52
4.3.3 Un exemple simple: deux tasses marginales.....	53
4.4 Technicalités concernant le module de déduction.....	56
4.4.1 Les faits.....	56
4.4.2 Les conjonctions.....	56
4.4.3 Une règle	57
4.4.4 Une disjonction de règles.....	57
5 PARTIE INDUCTIVE ET LA TABLE DES PROTOTYPES	59
5.1 Prédicats membres de la table des prototypes.....	59
5.2 Marginaux versus la table des prototypes	64
5.2.1 Marginaux positifs (classe II) et ensemble PP	66
5.2.2 Marginaux positifs et ensemble MP	67
5.2.3 Marginaux négatifs (classe IV) et ensemble PP.....	67
5.2.4 Marginaux négatifs et ensemble MP.....	68
5.3 Marginaux négatifs et localisation de la restriction	68
6 PARTIE INDUCTIVE: ANALYSE DES PRÉDICATS AYANT FAILLI.....	70
6.1 Notion de prototypes dans une taxonomie	70
6.2 L'ensemble FP et les prototypes	72

6.3 Nos animaux marginaux	73
6.4 Nos résultats.....	76
6.4.1 Les mammifères.	76
6.4.2 Les oiseaux.....	78
6.4.3 Les reptiles.	80
6.5 Discussion	80
7 TRAVAUX RELIÉS.....	81
7.1 IOU.....	81
7.1.1 Résumé	81
7.1.2 Un exemple.....	82
7.1.3 Comparaison avec LAP.....	84
7.2 LISE.....	85
7.2.1 Un exemple.....	86
7.2.2 Comparaison avec LAP.....	88
7.3 POSTHOC.....	89
7.3.1 POSTHOC: un modèle informatique.....	89
7.3.2 POSTHOC: un système à règles de production.....	91
7.3.3 POSTHOC: des résultats	93
7.3.4 Comparaison avec LAP.....	94
8 TRAVAUX FUTURS.....	95
9 CONCLUSION	95
10 RÉFÉRENCES	97

Liste des figures

Figure 1. Arbre de décision produit par ID3	6
Figure 2. Règles représentant le concept positif produit par ID3	6
Figure 3. Théorie du domaine pour $\text{safe_to_stack}(X,Y)$	7
Figure 4. Description d' un exemple pour $\text{safe_to_stack}(X,Y)$	7
Figure 5. Description du concept opérationnel pour $\text{safe_to_stack}(X,Y)$	8
Figure 6. Description des deux modèles d'apprentissage.....	9
Figure 7. Comparaison des méthodes EBL et SBL	17
Figure 8. Représentation des connaissances à deux paliers.....	25
Figure 9. Les marginaux versus les deux modèles d'apprentissage.....	29
Figure 10. Théorie du domaine pour l'exemple de la tasse.....	32
Figure 11. Tasse en polystyrène, un marginal positif.....	33
Figure 12. Petit seau, un marginal négatif.....	33
Figure 13. Règle pour inclure le marginal "styrofoam cup".....	33
Figure 14. Règle du premier palier pour exclure le marginal "petit seau".....	34
Figure 15. Règle du second palier pour exclure le marginal "petit seau"	34
Figure 16. Hiérarchie de type utile pour l'induction (BK).....	35
Figure 17. Système LAP	36
Figure 18. Prédicats de bas-niveau et leur catégorisation par la preuve (PCP).....	41
Figure 19. Théorie modifiée pour accepter un marginal positif.....	43
Figure 20. Extension pour restreindre la théorie dans le cas d'un marginal négatif	44
Figure 21. Deux règles disjointes permettant de prouver notre but.....	44

Figure 22. Deux règles arrivant directement aux faits.....	45
Figure 23. Arbre de preuve pour un échec de la preuve et PCP	46
Figure 24. Réparation de la théorie pour accepter un marginal positif (classe II).....	49
Figure 25a. Règle réparant la théorie au premier palier	50
Figure 25b. Règles, créées en cascade, réparant la théorie au premier palier.....	51
Figure 26. Arbre de preuve pour le marginal "tasse en polystyrène".....	54
Figure 27. BK pouvant justifier une nouvelle règle	54
Figure 28. Nouvelle(s) règle(s) justifiées par une BK	55
Figure 29. Ensemble NP pour le seau métallique.....	55
Figure 30. Règle restreignante non élaguée	55
Figure 31. Ensemble PTN et l'ensemble S de "Version Space"	62
Figure 32. Ensemble PTN et l'ensemble G de "Version Space".....	63
Figure 33. Règle produite avec une méthode faible pour le premier palier	66
Figure 34. Prédicats catégorisés selon la preuve pour la tasse "styrofoam".....	66
Figure 35. Prédicats catégorisés par notre méthode faible pour l'exemple du seau.....	68
Figure 36. Règle produite pour le premier palier concernant le seau	69
Figure 37. Règle produite pour le second palier concernant le seau.....	69
Figure 38. Taxonomie réduite pour le monde animal	71
Figure 39. Les tables PTI et PTN pour les mammifères.....	75
Figure 40. PCP et ses ensembles pour la baleine bleue.....	75
Figure 41. Règle suggérée pour la baleine.....	76

Figure 42. Les tables PTI et PTN pour les mammifères.....	76
Figure 43. PCP et ses ensembles pour le kangourou	77
Figure 44. Règle suggérée pour le kangourou	77
Figure 45. PCP et ses ensembles pour la chauve-souris.....	77
Figure 46. Règle suggérée pour la chauve-souris	77
Figure 47. PCP et ses ensembles pour l'ornithorynque	78
Figure 48. Règle suggérée pour l'ornithorynque	78
Figure 49. Les tables PTI et PTN pour les oiseaux	78
Figure 50. PCP et ses ensembles pour le manchot.....	79
Figure 51. Règle suggérée pour le manchot.....	79
Figure 52. PCP et ses ensembles pour l'autruche	79
Figure 53. Règle suggérée pour l'autruche.....	79
Figure 54. Les tables PTI et PTN pour les reptiles.....	80
Figure 55. PCP et ses ensembles pour la vipère d'Europe.....	80
Figure 56. Règle suggérée pour la vipère d'Europe.....	80
Figure 57. Le système IOU	82
Figure 58. Théorie pour 'drinking-vessel' utilisée par IOU.....	82
Figure 59. Description de la composante Cn du concept tasse par IOU	84
Figure 60. Le système LISE (deux premières étapes).....	86
Figure 61 Transactions de LISE	86
Figure 62. Actions de LISE	87

Figure 63. Exemple pour 'borrow' de LISE	87
Figure 64. Explication partielle de 'borrow' utilisant 'withdraw'	87
Figure 65. Nouvelles transactions créées par LISE	88
Figure 66. Théorie d'influence pour POSTHOC	90
Figure 67. Un exemple de POSTHOC pour une tâche de prédiction.....	91
Figure 68. Un exemple de POSTHOC pour une tâche de classification	91
Figure 69. Règles de production pour l'INITIALISATION par POSTHOC.....	92
Figure 70. Une hypothèse de POSTHOC, employant la théorie d'influence	92
Figure 71. Une hypothèse de POSTHOC, sans employer de théorie d'influence	92
Figure 72. Système à règles de production pour les ERREURS D'OMISSION	93

Liste des tables

Table 1. Ensemble d'exemples d'entraînement pour ID3	5
Table 2. Classification des exemples par LAP.....	27
Table 3. Cas typiques pour les mammifères, oiseaux et reptiles	72
Table 4. Marginaux pour les mammifères.....	74
Table 5. Marginaux pour les oiseaux et reptiles	74
Table 6. Exemples d'entraînement pour le concept de TASSE (IOU)	83
Table 7. Exemples d'entraînement réduits pour concept de TASSE (IOU)	83

INTRODUCTION

Continuellement durant sa vie, un être humain se sert de son jugement pour déduire si un exemple appartient ou n'appartient pas à un concept. Inconsciemment ou consciemment pour porter un jugement, il se sert de connaissances propres au domaine de ce concept. Il a emmagasiné ces connaissances au cours des années dépendamment de son milieu et de son vécu.

Malheureusement assez tôt dans son développement, il lui semble très difficile de remettre en question ces connaissances. *"Curieuse époque où il est plus facile de briser un atome que de briser un préjugé"*. Einstein résume bien le problème par cette phrase. Jeune, l'homme emmagasine et filtre peu la justesse de l'immensité des connaissances dans lequel il baigne. Très tôt, par souci de sécurité et économie de temps et d'énergie, il hésite à remettre en cause les connaissances acquises. Einstein croyait que, pour toute idée innovatrice et marginale, il fallait attendre une génération pour sentir que cette idée fasse partie du vécu et soit bien acceptée par la majorité. La relativité restreinte, énoncée en 1905, fut mal accueillie par les physiciens en place. La jeune génération qui suivit, ayant été en contact très tôt avec cette idée, la trouva normale et pleine de *"bon sens"*. Einstein lui-même, quoique ayant réfléchi à ce problème que le *"bon sens"* est mauvais conseiller, ne put accepter les idées innovatrices émanant de la mécanique quantique: *"Dieu ne joue pas aux dés"*. Le hasard ne pouvait pas jouer selon lui un rôle si important en sciences, cela allait contre tout *"bon sens"*.

L'humanité en général est régie par ce principe. Un être humain, jugé normal et bon citoyen, considère son pays comme le meilleur, sa religion comme la seule vraie et sa langue comme la plus belle. Il juge tout autour de lui en fonction des ses connaissances et, s'il doit reconnaître un *"marginal"*, il éprouve des difficultés. Il résiste au changement: inertie de l'esprit. Pourtant, la façon dont un être humain peut adapter et transformer les connaissances emmagasinées peut servir à différencier les gens caractérisés étroits d'esprit de ceux jugés souples et larges d'esprit. Racisme, guerre de religion et nationalisme sont des conséquences du manque de souplesse de l'esprit humain. Pour tous, il est très difficile de changer ou d'adapter des connaissances profondément emmagasinées en soi. Encore aujourd'hui aux États-Unis, certains États refusent le darwinisme.

Ce problème très complexe concernant l'apprentissage chez l'homme se transpose à un degré moindre dans un système informatisé. Un *système expert* peut représenter les

connaissances du domaine concernant un concept. Il va de soi que ce système normalement ne peut muter de lui-même ou ne peut s'adapter pour accepter des objets marginaux appartenant au concept mais non prévus au départ. L'expert humain est supposé lui avoir transféré toutes les connaissances pertinentes dès le départ. Un système expert représente un humain figé dans le temps et étroit d'esprit par rapport au changement. N'est-il pas seulement un expert?

Une grande différence entre l'homme et la machine est le libre arbitre. Heureusement et malheureusement, on ne peut imposer à un homme le fait qu'il doive accepter tel marginal comme membre à part du concept. La machine peut se faire imposer *a priori* le fait que le marginal doive faire partie du concept. A ce moment, il ne lui reste qu'à mettre sa puissance de calcul, sa facilité de déduction et son *parti-pris inductif* (inductive bias) au service du module d'apprentissage. L'*induction* est un mécanisme d'inférence dangereux car il peut générer la fausseté, mais il peut être responsable de créativité si l'heuristique concernant le choix de son parti-pris inductif est pertinent. L'humanité a acquis son expertise grâce à l'induction et, par déduction, elle se sert de cette expertise. Devant une série de mesures, Newton a induit les lois de base de la physique classique. Se servant de ces lois, les ingénieurs de la NASA déduisent les trajectoires des satellites. Pourtant, les lois induites par Newton ne sont qu'une approximation des lois plus justes induites en relativité. Depuis des siècles et encore aujourd'hui, l'homme se fie à un résultat en partie faux de l'induction et il le fait avec une grande créativité.

Une théorie n'est aucunement figée dans le temps et l'espace. Un système se doit de s'adapter à une extension du langage pouvant permettre de le transformer en vue d'accepter ou de réfuter de nouveaux exemples non prévus au départ.

Cette thèse se veut une contribution à ce problème. Elle scrute la possibilité d'avoir un module d'apprentissage permettant à une *théorie du domaine*, en présence des marginaux, de s'adapter et de briser la rigidité de ses règles en proposant de nouvelles règles à l'expert: l'inverse du mécanisme normal de l'acquisition des connaissances. Le fait que le système aide l'expert, allège le problème du *goulot d'étranglement* (knowledge bottleneck) si commun lors de l'acquisition de connaissances.

Le chapitre 1 présente les buts poursuivis en apprentissage automatique et souligne la contribution de cette thèse à ce champ de recherche. Le chapitre 2 contient un survol des méthodes de base employées en apprentissage automatique. Le chapitre 3 introduit la notion de marginal (instances mal classées) et analyse diverses possibilités permettant de

Thèse - Boris Plante

mieux les classer. Ce chapitre introduit la solution de l'auteur. Le chapitre 4 présente la partie déductive du système utilisé. Les chapitres 5 et 6 traitent de la partie inductive du système. L'induction d'une nouvelle règle peut se faire de plusieurs façons. Le chapitre 6 montre comment on peut utiliser les prédicats coupables de l'échec. Le chapitre 5 présente le dernier pas inductif essayé par le système pour cerner le marginal. Le chapitre 7 résume quelques travaux reliés à cette recherche. Plusieurs suggestions de travaux futurs se retrouvent énumérés au chapitre 8. Enfin, le chapitre 9 conclue ce travail.

1 PRÉSENTATION DE LA RECHERCHE

1.1 But des méthodes de base en apprentissage automatique

En *apprentissage automatique* (machine learning), deux grandes méthodes se côtoient: l'apprentissage inductif ou empirique et l'apprentissage déductif ou analytique. Ces deux méthodes sont utilisées lors de l'apprentissage de concepts. Ces méthodes, quoique très différentes, tentent à produire des résultats similaires. Souvent le concept déduit ou induit sera représenté par une *disjonction* de règles. Chaque règle se compose de la *conjonction* de prédicats de bas-niveau. Le concept formé est un concept opérationnel (ex. structural), pouvant s'apparier directement aux prédicats de l'exemple, d'où le nom de prédicats de bas-niveau. Ces deux méthodes sont brièvement décrites ci-dessous.

L'*apprentissage par similarité* (SBL), pour définir son concept, utilise normalement aucune théorie du domaine. Cet apprentissage produira une définition opérationnelle utilisant les prédicats se retrouvant dans la multitude d'exemples qu'il consulte. Son outil principal est l'induction. Souvent une *connaissance d'arrière-plan* (background knowledge), généralement sous forme de réseau sémantique ou de structure hiérarchique, aide à généraliser la définition produite. Dans ce cas, la définition du concept pourra contenir des prédicats non contenus dans la description des exemples. Cette méthode situe son apprentissage au *niveau des connaissances* (knowledge level) [Dietterich 1986]. En effet, une nouvelle connaissance non présente implicitement est acquise. Il est important de remarquer que la connaissance acquise peut être représentée sous différentes formes, par exemple, des arbres de décision (ID3) [Quinlan 1986], des règles de production [Michalski 1983].

Il va de soi que la qualité du concept appris dépend du parti-pris inductif employé, de l'algorithme inductif et de la disponibilité d'exemples représentant bien le concept.

Un exemple simple et connu dans la littérature peut illustrer ceci [Quinlan 1986]. On veut trouver une façon de classer les "samedis matin" en fonction de certains critères concernant la météorologie, comme la température et l'humidité.

Pour ne pas détourner les lecteurs familiers en apprentissage automatique du but premier des exemples, ceux-ci seront conservés dans leur langue originale, l'anglais. L'ensemble des exemples utilisés se trouve énuméré à la table 1.

No	Outlook	Temperature	Humidity	Windy	Class
1	sunny	hot	high	false	negative
2	sunny	hot	high	true	negative
3	overcast	hot	high	false	positive
4	rain	mild	high	false	positive
5	rain	cool	normal	false	positive
6	rain	cool	normal	true	negative
7	overcast	cool	normal	true	positive
8	sunny	mild	high	false	negative
9	sunny	cool	normal	false	positive
10	rain	mild	normal	false	positive
11	sunny	mild	normal	true	positive
12	overcast	mild	high	true	positive
13	overcast	hot	normal	false	positive

Table 1. Ensemble d'exemples d'entraînement pour ID3

Nous remarquons que chaque exemple est décrit par une collection d'attributs auxquels on donne des valeurs. Pour chaque exemple, on fournit sa classification: bon exemple ou mauvais exemple. Par exemple, l'attribut *outlook* peut avoir trois valeurs: *sunny*, *overcast* et *rain*. Chaque exemple appartient à un ensemble de classes mutuellement exclusives. Ici il y a deux classes: les exemples positifs et les exemples négatifs du concept qu'on veut apprendre. Cette représentation est certes une des plus simples, elle utilise des *paires attribut-valeur*. Le concept cherché répond à un besoin qui peut nous être inconnu concernant les samedis matin. Quelle signification pouvons-nous donner à ces règles? Imaginons que ce sont des préférences concernant le "jogging" du samedi matin.

Étant donné les exemples et leur classification, la tâche est de produire des règles qui pourront classer tous les exemples: ceux connus et fournis pour l'apprentissage ainsi que les nouveaux exemples éventuels.

L'arbre de décision obtenu par l'algorithme ID3 optimise la quantité d'information à chaque niveau. La figure 1 représente cet arbre optimal. L'arbre de décision servira à classer de nouveaux exemples en décidant si oui ou non ce dernier appartient au concept. L'algorithme veut trouver l'arbre pouvant nous donner la meilleure information à chaque

noeud. Le choix de l'attribut pour chaque nouveau noeud se base sur la théorie d'information: l'attribut choisi doit subdiviser les exemples de la façon la plus efficace. Quinlan nous présente d'autres arbres pouvant classer adéquatement les mêmes exemples mais leur complexité les rend inadéquats pour donner une *explication* des exemples d'entraînement.

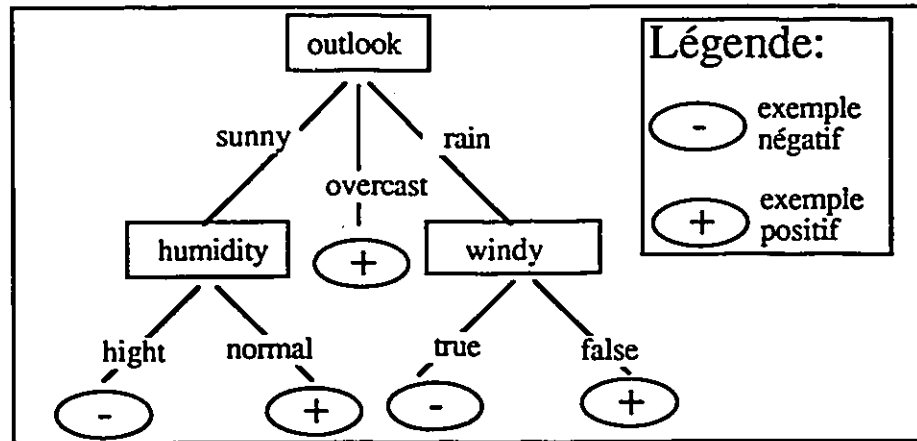


Figure 1. Arbre de décision produit par ID3

Quelques règles pouvant représenter le concept découlant des exemples positifs se retrouvent groupées à la figure 2. Les règles s'obtiennent en partant de la racine de l'arbre de décision et en se dirigeant vers une feuille positive, chaque noeud rencontré lors de ce trajet fera partie de la règle pour classer cette feuille. Ici, les exemples positifs peuvent être représentés par trois feuilles et on peut en déduire trois règles. Les négatifs conduisent à deux règles.

concept_+(X)	<--	outlook(X,sunny) and humidity(X,normal)
concept_+(X)	<--	outlook(X,overcast)
concept_+(X)	<--	outlook(X,rain) and windy(X,false)

Figure 2. Règles représentant le concept positif produit par ID3

Les règles pour le concept positif (i.e. le temps souhaité pour le jogging) représentent donc les journées couvertes, les jours de pluie sans vent et les jours ensoleillés peu humides.

Voici un résumé des conséquences et résultats de cet apprentissage (SBL):

- les règles produites ne contiennent aucune structure ou hiérarchie.
- souvent le symbole représentant le concept sert de *conséquent* à la règle produite et les *antécédents* sont formés de prédicats opérationnels, i.e. des prédicats de bas-niveau se retrouvant directement dans les exemples utilisés.

L'apprentissage analytique (EBL), pour définir son concept, utilise une *théorie du domaine* (domain theory). Cet apprentissage produira une définition opérationnelle en se servant des règles de haut-niveau (souvent non-opérationnelles) pour aboutir aux prédicats de bas-niveau appartenant à l'exemple utilisé. Ayant constaté que l'exemple appartient bien à la théorie, la méthode généralisera une nouvelle définition du concept à l'aide d'un critère d'opérationnalité. Cette méthode n'augmente en rien la connaissance du système, elle change la connaissance actuelle en une forme qui se veut plus utile et plus efficace [Mitchell, Keller et al. 1986]. Cette forme d'apprentissage est qualifiée de "*speed-up learning*" [Dieterich 1986].

Un exemple simple et largement employé dans la littérature peut illustrer nos dires. La théorie du domaine explique comment on peut avoir une définition opérationnelle de "safe_to_stack(X,Y)" [Kodratoff 1988]. Le but est de placer un objet Y sur un objet X sans le briser.

La figure 3 résume la théorie du domaine utilisée. Cette théorie nous indique de façon générale quand il est sécuritaire de placer un objet au-dessus d'un autre.

safe_to_stack(X,Y)	<--	not fragile(X)
safe_to_stack(X,Y)	<--	lighter(X,Y)
lighter(X,Y)	<--	weight(X,V1) and weight(Y,V2) and less(V1,V2)
weight(X,W)	<--	volume(X,V) and density(X,D) and W is V * D
weight(X,50)	<--	isa(X,endtable_1)
less(X,Y)	<--	X < Y.

Figure 3. Théorie du domaine pour safe_to_stack(X,Y)

Un exemple pouvant être satisfait par cette théorie nous est donné à la figure 4.

on(box_1,endtable_1)	color(box_1,red)
color(endtable_1,blue)	volume(box_1,10)
density(box_1,1)	owner(box_1,bonnie)
fragile(endtable_1)	owner(endtable_1.clyde)

Figure 4. Description d'un exemple pour safe_to_stack(X,Y)

Le résultat obtenu dépend du critère d'opérationnalité. À la figure 5, on retrouve le concept obtenu en considérant les feuilles de la théorie comme étant les prédicats jugés

opérationnels, ainsi que le prédicat "*less*". De plus, on a généralisé la définition opérationnelle du concept pour permettre l'appariement avec des cas similaires.

<code>safe_to_stack(O1,O2)</code>	<code><--</code>	<code>volume(O1,V1) and density(O1,D1) and less(V1*D1,50) and isa(O2,endtable_1)</code>
-----------------------------------	---------------------	---

Figure 5. Description du concept opérationnel pour `safe_to_stack(X,Y)`

Voici un résumé des conséquences et résultats de cet apprentissage (EBL):

- la théorie est structurée sous forme d'arbres. Les noeuds représentent des prédicats non-opérationnels et les feuilles représentent les prédicats opérationnels se retrouvant dans les exemples.
- comme dans le cas de l'apprentissage basé sur les similitudes, la définition du concept formée se compose d'un conséquent qui représente le concept et d'antécédents qui, pour la plupart du temps, sont des prédicats opérationnels.

Il est intéressant de voir que les deux méthodes tentent à produire une description opérationnelle du concept. Donc la description doit être simple et peu profonde en ce qui concerne la structure représentant cette définition.

Maintenant que nous avons bien vu les buts des deux méthodes d'apprentissage, nous allons analyser très rapidement leur fonctionnement. La figure 6 montre une représentation graphique de ces deux méthodes [Krawchuk and Witten 1988].

L'apprentissage basé sur les similitudes (SBL) utilise plusieurs exemples et contre-exemples. Il utilise un algorithme ayant un parti-pris inductif pour généraliser une définition du concept. Ce mode d'apprentissage repose sur certaines hypothèses contraignantes [Hall 1988]. D'abord les régularités observées dans les exemples doivent refléter les lois causales associées au concept. De plus, on suppose que le concept peut être décrit dans un langage limité ou biaisé lors du choix de la représentation des connaissances [Utgoff 1986].

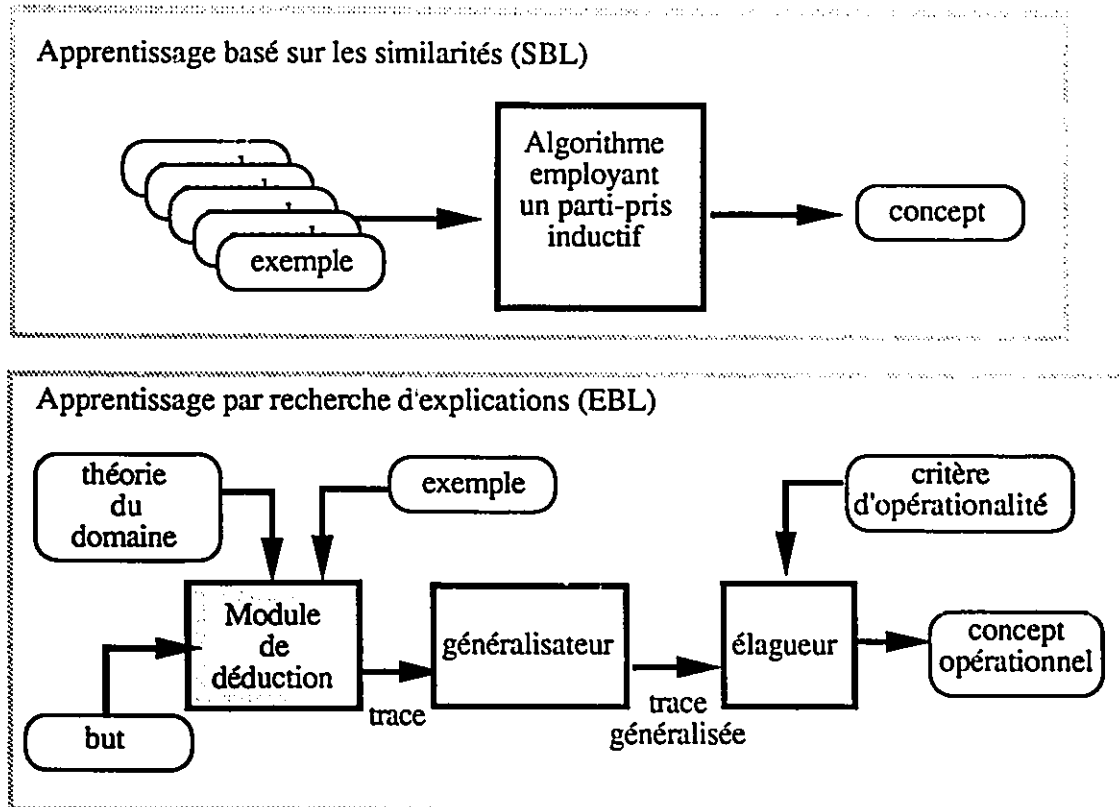


Figure 6. Description des deux modèles d'apprentissage

L'*apprentissage par recherche d'explication* (EBL) part d'un exemple et essaie de déduire une explication à l'aide d'une théorie du domaine. Cette explication n'utilise que les parties utiles (relevant) de l'exemple. Cette explication est généralisée pour pouvoir servir dans le cas d'autres exemples similaires. De plus, on ne garde de cette explication que la partie opérationnelle. En d'autres mots, EBL essaie par le biais d'une théorie du domaine de remplacer les préconditions de sa théorie par des *préconditions plus faibles*. Les limitations de ce système sont assez évidentes. D'abord la théorie se doit d'être complète. Si nous ne pouvons pas expliquer les antécédents, EBL faillit. De plus les règles produites par EBL sont toujours correctes en relation avec la théorie du domaine peu importe le contexte. Donc la théorie doit être également correcte (consistant). Comme pour le cas de SBL, EBL est limité par le choix du langage. L'ensemble des constructions permises par ce langage peut en réalité être un sous-ensemble insuffisant pour décrire adéquatement la théorie ou le concept.

Les deux modèles sont donc limités par le choix du langage: c'est un parti-pris obligatoire. Notre recherche adresse ce problème pour permettre au système de s'adapter à un changement ultérieur et non prévu du langage.

1.2 Le but de notre recherche

Nous nous attaquons à une autre dimension de l'apprentissage. Notre but n'est pas de donner une définition opérationnelle du concept étudié. Notre but est de raffiner la théorie utilisée pour qu'elle puisse accepter certaines *instances positives* non couvertes par la théorie actuelle ou refuser certaines *instances négatives* couvertes par la dite théorie. Nous voulons suggérer de nouvelles règles servant à mieux cerner la théorie du domaine. Ces règles pourraient se greffer un peu partout dans l'arbre représentant la théorie: elles pourraient donc être des sous-arbres. Ces nouvelles règles pourraient aussi se placer dans un deuxième palier de connaissances qui serait utilisé pour traiter les exceptions présentes dans toutes théories. Au chapitre trois, nous détaillerons ces deux approches à la marginalité.

Une particularité de ces règles se retrouve dans la syntaxe pouvant la définir. Le conséquent est soit le concept ou un prédicat non-opérationnel. Les antécédents sont soit des prédicats non-opérationnels ou des prédicats opérationnels. Nous voulons donc *réviser* la théorie, la changer et l'améliorer. La définition du concept est secondaire pour nous. Notre but ultime est de suggérer des règles pouvant changer l'arbre représentant la théorie du domaine lorsqu'elle est *incomplète* et *incorrecte* ou encore imposer des restrictions à une théorie du domaine trop générale.

Les problèmes inhérents à cette méthode se résument ainsi:

- Où réparer dans l'arbre? En effet, pour un exemple qu'on veut maintenant couvrir, à quelle hauteur devons-nous placer la réparation suggérée? Quel noeud choisir? Sur quel noeud agira la restriction?
- Pour un noeud qu'on veut ajouter, pouvons-nous garder d'anciennes parties de l'arbre appartenant à d'autres noeuds similaires (disjonction)?
- Où devons-nous prendre les parties nouvelles de ce sous-arbre?
- Comment devons-nous rapidement focaliser sur les bons prédicats parmi une gamme de sujets prometteurs pouvant faire partie de la nouvelle règle (sous-arbre) ou de la restriction?

Cette thèse suggère des méthodes pour répondre à ces questions. L'emphasis est mis sur l'usage de méthodes faibles. Il va de soi que l'on soulignera la possibilité d'utiliser des méthodes fortes employant une connaissance d'arrière-plan.

Les méthodes faibles sont très importantes. Elles permettent d'oser, d'essayer. Avec peu d'information accessible, il est difficile d'élaborer des méthodes aspirant trouver une similarité ou tirer des conclusions. Le mérite des méthodes faibles consiste à faire le plus de choses possibles avec peu de moyens. Il va de soi que les résultats seront généralement moins flamboyants que ceux obtenus par des méthodes utilisant beaucoup de connaissances. Les méthodes faibles forment un point de départ excellent pour tout système. D'autres systèmes d'apprentissage voulant traiter les marginaux pourraient greffer à notre méthode ses propres connaissances afin de former une méthode beaucoup plus forte.

1.3 La pertinence de la recherche

L'utilité de cette recherche se situe à deux niveaux très importants.

A un premier niveau, la modification par elle-même d'une théorie du domaine, pour qu'elle puisse tenir compte d'instances marginales, est un tour de force. En effet, le fait de pouvoir ajuster une théorie du domaine en se servant de l'exemple qui est mal classé peut être très utile. Pour un marginal, on proposera une méthode permettant d'induire des conclusions avec un exemple de ce type de marginal et là réside la difficulté.

A un deuxième niveau, notre méthode veut utiliser des heuristiques pouvant mimer certains modèles utilisés par les humains lorsque ceux-ci rencontrent des marginaux. Comment pouvons-nous juger un marginal en opposition à la majorité des éléments appartenant à une classe de prototypes. En quoi un marginal est-il marginal? Un marginal et la majorité peuvent faire partie d'un même groupe, mais comment devons-nous le rationaliser?

Imaginez le scénario suivant. On s'aperçoit qu'un système expert doit être modifié pour s'ajuster à une situation nouvelle et changeante. Les règles qu'il contient ne représentent plus la réalité. Des cas nouveaux non prévus originellement ne peuvent pas être bien classés: la théorie est incomplète et/ou incorrecte. Le langage décrivant les exemples doit être étendu. Notre méthode se veut une méthode automatique pour modifier les règles existantes ou pour en ajouter de nouvelles pouvant contenir des prédicats appartenant soit à

l'ancien langage ou au nouveau. Si le système est pourvu d'un module d'apprentissage, il pourrait se placer en mode ' *apprentissage* ' afin de suggérer de nouvelles règles aux experts.

Cette recherche veut étudier un modèle pouvant répondre à ces besoins. Le modèle proposé est une *méthode mixte d'apprentissage* qui utilise la déduction et l'induction pour essayer de former de nouvelles règles.

La déduction forme la première étape de la méthode, on essaie d'analyser le système existant pour y localiser ses faiblesses et pour catégoriser tous les prédicats de l'exemple. En effet, certains prédicats participent ou non à la preuve. Les prédicats participant à la preuve peuvent se trouver soit dans les règles coupables de l'échec ou soit dans des règles satisfaites par l'exemple. Les prédicats ignorés par la preuve peuvent aussi se retrouver chez les instances jugées prototypes ou majoritaires.

L'induction sert à focaliser notre recherche à partir des prédicats classés par la partie déductive. Le pas inductif dans cette méthode se base sur cette heuristique: '*Un marginal se doit d'être expliqué en comparaison avec la majorité ou les prototypes*'. Un élément est jugé marginal par la majorité. L'induction joue un rôle créateur. Pour que la création se fasse de façon optimale, la déduction joue un rôle de *parti-pris déclaratif* pour la partie inductive qui suivra. La déduction sert donc à extirper, de la théorie du domaine et des exemples, notre propre connaissance d'arrière-plan.

En résumé, nous développons une méthode générale et faible (weak method). En effet, seule la théorie du domaine et les exemples sont utilisés pour trouver une règle. Le but est de modifier un ensemble de règles (théorie, Système Expert) existantes pour que le système étudié puisse s'adapter à accepter certains marginaux.

De plus, les heuristiques que nous développons, peuvent servir de base à une étude d'un modèle cognitif chez l'humain. L'être humain durant sa vie recueille de l'information et se fait une idée prototypique des concepts qui l'entourent. Pour la majorité des québécois, un Africain est un individu de couleur noire. A sa grande surprise, il considérera un Africain du nord comme étant un marginal. A ce moment, il devra ajuster son image d'un Africain. Son concept sera étendu pour inclure ce marginal. Il jugera ce marginal en relation avec son image des prototypes.

1.4 Survol du système et limitations

Dans le cadre de cette thèse, on a limité la recherche de plusieurs façons. La théorie étudiée contient des disjonctions de règles et des conjonctions. Aucune limite théorique n'est imposée quant à la profondeur de l'arbre représentant la théorie du domaine. Pour le moment, la théorie ne contient pas de négations. L'implémentation jointe à cette thèse utilise des paires attribut-valeur comme celles utilisées par ID3: une seule variable est permise et représente l'identificateur de l'exemple étudié.

Notre implémentation se divise en deux passes.

La première passe traite tous les exemples. Elle emploie la déduction pour grouper les exemples en quatre classes: *prototypes*, *marginiaux positifs*, *marginiaux négatifs* et les *négatifs*. En outre, on y construit les structures de données telle la table des prototypes. Par la suite, les règles sont construites à l'aide de cette table. Ceci constitue le *parti-pris déclaratif*. Une fois cette passe faite, on connaît nos marginaux négatifs et positifs, et on connaît les vrais exemples positifs et les vrais exemples négatifs. Nos exemples marginaux sont donc étiquetés "marginiaux" après la première passe. La classification (exemple positif ou négatif) et la réaction de la théorie à cet exemple permettent de bien étiqueter nos exemples: un exemple positif refusé par la théorie sera certes un exemple marginal, car on doit ajuster la théorie.

La passe 2 en est une incrémentale où l'utilisateur choisit d'étudier tel ou tel exemple marginal. La partie inductive du système, discrimine chaque marginal en fonction des exemples considérés comme étant des prototypes. Pour cela, il emploie la table servant à concentrer l'information concernant les prototypes. Il va de soi qu'une implémentation complètement incrémentale aurait été possible et cela au dépens de structures de données temporaires. Cependant, un tel système n'aurait pas généré de meilleurs résultats. En effet, des règles suggérées, efficaces et élaguées, ne pourraient être fournies qu'à la fin de la session de travail: car les meilleures règles se basent sur l'information la plus complète possible concernant les prototypes et les négatifs. On ne peut rien déduire ou induire concernant un marginal si on n'a pas un milieu riche en information sur les normes, i.e. sur les prototypes.

Si on rattache ces dernières précisions au scénario énoncé plus haut concernant les systèmes experts, on peut imaginer que le système expert peut emmagasiner lors de son usage des informations pertinentes quant à son exécution (passe 1) et qu'on peut lui

demander en tout temps de passer en mode suggestion de règles pour les marginaux rencontrés (passe 2). Ces règles pourront être introduites dans la théorie initiale ou former une seconde couche pouvant expliquer les marginaux, nous ne prenons pas partie pour une tendance ou l'autre.

1.5 Discussion des résultats

Il est intéressant de voir que la méthode suggérée fut essayée avec deux exemples: une définition de concept et une taxonomie. Dans les deux cas, elle produit des règles pour étendre la théorie en se fiant aux exemples réussis et en essayant de démarquer le marginal. La théorie est modifiée en incorporant un second palier. Ce second palier contient les extensions et restrictions permettant de traiter les marginaux.

Les règles suggérées furent très pertinentes et suggèrent que notre méthode pourrait être employée pour traiter les cas d'exceptions dans de nombreux systèmes diversifiés par leurs tâches. Notre méthode, étant indépendante du contenu de la théorie, peut être greffée à tout système qui voudrait réagir à son environnement.

De plus, on peut facilement modifier la méthode pour permettre l'incorporation de nombreuses connaissances d'arrière-plan qui pourrait rendre le système plus autonome et plus fiable.

1.6 Contribution de la thèse

Le contenu de cette thèse se veut innovateur sur plusieurs points. Pour trouver une règle pour démarquer un marginal, le système n'utilise pas de connaissance pré-établie. On procède à l'élaboration de cette connaissance en se fiant à la théorie et aux cas typiques: ce sont ses guides. La thèse s'attaque donc à un des problèmes fondamentaux des systèmes basés sur des connaissances: celui du manque d'adaptation (*brittleness*). La thèse propose une solution à un des aspects du problème: celui de l'adaptation d'un système à des cas marginaux grâce à une technique d'apprentissage automatique.

1.6.1 Classification des prédicats

Ayant une théorie, la déduction est utilisée pour classer chaque prédicat de l'exemple en connaissant son rôle joué dans la preuve. En fonction de son rôle, chaque prédicat est inclus dans un ensemble approprié. Le système se base sur une approche formelle et permet d'introduire une catégorisation utile des prédicats employés dans une preuve de style EBG.

On part d'une jungle de prédicats et on classe ces prédicats selon leur utilité et leur rôle. Le système présenté, LAP (Learning by Analysing the Proof), utilise la puissance de la théorie pour aller chercher une signification de l'information (regroupe l'information et la raffine) à partir de données brutes déjà présentes.

1.6.2 Utilisation des prototypes comme guide à l'innovation (pas inductif)

La déduction sert aussi à classer l'information concernant les prototypes déjà acceptés par la théorie. En effet, LAP tend à combler un manque d'information concernant les marginaux en regroupant sous une forme utilisable l'information éparpillée et nombreuse rattachée aux prototypes.

Le pas inductif de LAP sera plus rapide car il essaiera de démarquer le marginal en utilisant l'information contenue dans la classification des prédicats du marginal et en utilisant l'information regroupée concernant les prototypes. LAP essaie de recouper l'information classée de façon formelle plutôt que de travailler dans une mer désordonnée d'information.

1.6.3 Une méthode adaptative

Notre méthode se veut une méthode mixte d'apprentissage qui permet de transformer un système rigide en un système adaptatif, i.e. capable de s'adapter à son environnement.

Notre méthode permet de donner une solution structurée et formelle au problème de la modification dynamique d'une théorie incomplète.

1.6.4 Une méthode indépendante de la théorie

De plus, notre méthode est indépendante du contenu de la théorie. LAP restructure toute la théorie utilisée par les marginaux et reformule l'information. On peut dire que LAP emploie une forte 'méthode faible'. La méthode est dite faible car elle n'utilise pas de théorie d'arrière plan, mais la méthode est jugée forte car la réorganisation de l'information par notre système lui donne une puissance accrue pour l'établissement de nouvelles règles comme si une théorie d'arrière-plan était disponible.

2 APPRENTISSAGE AUTOMATIQUE: un survol

2.1 Limitations des deux méthodes principales en apprentissage

Nous avons déjà introduit les deux méthodes de base en apprentissage automatique. La figure 6 contenue dans le chapitre 1 résume bien ces deux méthodes. Dans cette section, nous soulignerons les avantages et les inconvénients associés à chaque méthode.

De nombreux programmes semblent ne rien apprendre au niveau des connaissances. LEX [Mitchell, Utgoff et al. 1983], un programme spécialisé en calcul intégral, ne semble rien apprendre sur la façon d'intégrer. Toutes les intégrales qu'il peut faire sont déterminées au départ. Il apprend à effectuer plus rapidement des intégrales similaires au fur et à mesure de son apprentissage. Dietterich [Dietterich 1986] considère que ce programme n'apprend rien au niveau des connaissances. Dernièrement, Kokar [Kokar 1990] propose une nouvelle façon de spécifier le niveau des connaissances d'un agent d'apprentissage. Il propose un apprentissage pouvant se définir selon cinq dimensions:

- apprentissage du langage
- apprentissage d'axiomes
- apprentissage de théorèmes
- apprentissage de règles d'inférences
- apprentissage de règles d'interprétation

Puisque LEX part d'un ensemble plat de règles d'inférence et apprend à associer des préconditions à ces règles, Kokar juge que LEX apprend au niveau des règles d'inférence et que cela change sa connaissance.

La figure 7 compare les méthodes EBL et SBL en soulignant leurs forces et leurs faiblesses.

Apprentissage Empirique	Apprentissage Analytique
•Outil: Induction •But: produire une description d'un concept en généralisant à partir d'un ensemble d'exemples du concept. •Avantages: -SBL produit un concept sans utiliser une théorie du domaine. -SBL extrait la connaissance des exemples. On apprend au niveau des connaissances (knowledge level). -SBL fait un pas inductif (inductive leap). Ce pas peut causer un apprentissage contenant des faussetés: danger de la créativité. •Inconvénients: -SBL nécessite un parti-pris inductif pour généraliser le concept et est dépendant de ce choix. -SBL est dépendant de la qualité des exemples. -SBL ne peut induire à partir d'un seul exemple. - S B L traite mal les cas d'exceptions.	•Outil: Dédution •But: produire une description généralisée et opérationnelle d'un concept avec un seul exemple. •Avantages: -EBL produit un concept opérationnel avec un seul exemple. EBL ne produit pas de nouvelles connaissances, EBL réarrange les connaissances de façon plus utile, c'est donc un apprentissage de type "<i>speed-up learning</i>". -EBL élimine grâce à la théorie les caractéristiques non significatives de l'exemple. -EBL fournit une justification du concept fourni (arbre de preuve). La justesse de la théorie décide de la justesse du concept. •Inconvénients: -EBL nécessite une théorie complète et correcte. EBL dépend de cette théorie -EBL spécialise en fonction d'un seul exemple, on doit pouvoir traiter plusieurs exemples en vue d'obtenir une meilleure définition ni trop spécialisée et ni trop généralisée. - E B L traite mal les cas d'exceptions.

Figure 7. Comparaison des méthodes EBL et SBL

2.2 Révision et/ou reformulation de la théorie du domaine

La méthode EBL est employée pour *reformuler* la théorie. Elle n'est pas employée pour redéfinir la théorie ou pour la réparer. Pour obtenir une définition du concept, cette méthode emploie une théorie du domaine et reformule la théorie en la rendant opérationnelle grâce à un exemple. L'utilité de ce processus consiste à pouvoir remplacer une définition non-opérationnelle par une définition opérationnelle. L'appariement sera plus facile et plus rapide sauf s'il y a une sur-génération de ces règles opérationnelles. La théorie se doit d'être parfaite: une utopie.

Plusieurs méthodes mixtes furent présentées récemment dans le but de pouvoir produire une définition d'un concept à partir de théories imparfaites. Mais la majorité d'entre elles s'attaquent à la définition du concept (*reformulation*) en présence d'une théorie imparfaite.

On doit aussi s'attaquer au problème de *réviser* la théorie, i.e. trouver ses faiblesses et tenter de l'améliorer.

2.3 Apprentissage mixte

De nombreux systèmes utilisent les deux méthodes de base EBL et SBL pour former un système hybride d'apprentissage où les deux méthodes collaborent étroitement.

Nous pouvons approcher l'étude des méthodes mixtes de plusieurs façons. Deux approches nous intéressent particulièrement.

La première approche consiste à voir comment on combine les deux méthodes de base pour en faire un système d'apprentissage. Nous ne voulons pas étudier nécessairement toutes les méthodes mixtes dans le détail. Le chapitre 6 de la thèse de Johanne Morin est consacré à la comparaison de six méthodes mixtes et peut servir de base à ce résumé [Morin 1990].

La seconde approche est de classer les méthodes mixtes selon le but qu'elles poursuivent.

2.3.1 Hybridation de EBL et SBL

Lorsque les deux méthodes sont utilisées dans un seul système, il est très probable que l'une des deux méthodes serve de guide à la seconde ou que l'une des deux prépare le terrain (connaissances) pour la seconde.

A) EBL suivie de SBL

- La majorité des méthodes suivent ce patron. EBL est employé incrémentalement pour obtenir des explications (weakest preconditions) concernant les exemples positifs. Ces explications seront fournies à SBL qui induira des généralisations afin d'obtenir une définition correcte du concept cherché.

EBG [Mitchell, et al. 1986], mEBG [Hirsh 1988] et IOE [Dietterich and Flann 1989] utilisent cette façon de faire. IOE emploie un parti-pris moins fort, ainsi sa définition du concept peut couvrir plus d'exemples inconnus que les deux autres.

- La méthode EBL est employée incrémentalement pour obtenir des explications qui seront généralisées, SBL sera employée sur les parties non employées par EBL (unused parts) et non sur les explications. SBL sert à ajouter au concept trop général une partie inexplicable qui servira à restreindre le concept. IOU [Mooney and Ourston 1989] travaille de cette façon. LAP travaille aussi de cette façon, mais la déduction sert aussi à classer les exemples et à grouper les données inutilisées. Dans LAP, SBL est employée incrémentalement sur les marginaux à tour de rôle.

B) SBL suivie de EBL

Ce second patron étant très rare, il ne sera pas considéré dans ce travail.

C) SBL et EBL intermêlées

[Bergadano and Giordana 1988] emploie EBL simultanément sur tous les exemples. A un noeud opérationnel, si EBL ne peut couvrir tous les exemples positifs (emploi de statistiques), alors SBL est appelé pour aider à généraliser le concept à cet endroit. L'induction est guidée par la déduction et vient aider si nécessaire. De plus, SBL pourra spécialiser le concept vers la fin de la déduction pour éliminer des négatifs indésirables.

Ici, EBL conduit le processus dans lequel SBL vient raffiner la définition du concept si nécessaire, pour tous les exemples simultanément.

2.3.2 But des méthodes mixtes

Plusieurs méthodes mixtes ont comme premier but de trouver une définition opérationnelle du concept.

Les systèmes suivants poursuivent ce but: EBG, mEBG, IOU, IOE.

Plusieurs nouvelles méthodes mixtes poursuivent un but différent. Elles veulent raffiner la théorie du domaine utilisée par EBL

Les systèmes suivants poursuivent ce but: LAP, LISE [Genest, Matwin et al. 1990].

2.4 Problèmes en apprentissage

Cette section présente les problèmes inhérents à l'utilisation de connaissances lors de l'apprentissage.

La notation $A \vdash B$ signifie que B peut être prouvé à partir de A. Donc A est plus général que B. La négation de $A \vdash B$, notée $A \nvdash B$, signifie qu'on ne peut pas prouver B à partir de A. L'induction sera donc résumé ainsi: à partir de faits, on doit trouver une hypothèse H qui sera plus générale que les faits F.

$$H \vdash F$$

Souvent une connaissance B concernant les faits est disponible, on pourra redéfinir l'induction de cette façon:

$$B, H \vdash F$$

[Puget 1989], dans le cadre de l'induction, divise en trois classes les méthodes d'apprentissage basées sur les échecs:

- 1 Type 1: la connaissance B est correcte mais *incomplète*. B ne peut pas expliquer seul les faits. $B \nvdash F$. Alors, on se doit de trouver une hypothèse H pour expliquer les faits, c'est le cas classique de l'induction.

$$B, H \vdash F$$

- 2 Type 2: la connaissance B est *incorrecte* et permet d'expliquer des faits F indésirables. Alors, on doit enlever de B certaines particularités responsables de ce comportement incorrect, ce qui permettra d'exclure ces faits.

$$(B-H) \nvdash F$$

- 3 Type 3: Si F n'est pas vrai, alors on a

$$B \nvdash F$$

mais on devrait pouvoir dire que $B \vdash \text{not } F$ i.e. qu'il faut trouver une hypothèse H qui permet de nier F.

$$B, H \vdash \text{not } F$$

Nous étendons son interprétation pour couvrir les méthodes utilisant aussi la déduction. Nous sommes seulement intéressés par les deux premiers types d'échecs.

Les ensembles F^+ et F^- regroupent respectivement les faits positifs et négatifs. Dans le cadre de méthodes mixtes, T peut représenter la théorie du domaine rattachée au concept et B résume un ensemble de connaissances d'arrière-plan qui sont employées tant par la déduction que l'induction. Habituellement, des règles de production sont employées pour représenter T et des réseaux sémantiques peuvent être une représentation de B .

Les deux premiers types d'échecs peuvent maintenant se reformuler ainsi:

- 1 échec du premier type (théorie incomplète): T ne peut expliquer tous les faits F^+ . On peut additionner une hypothèse H à T telle que

$$B, T+H \vdash F^+ \text{ et } B, T+H \not\vdash F^- \quad (1)$$

Ici, on rajoutera une disjonction permettant d'*étendre* la théorie à ces faits positifs.

- 2 échec du second type (théorie incorrecte): T peut expliquer quelques faits F^- . L'approche originale dans le cadre du modèle inductif est d'identifier une partie H de la théorie qui, soustraite de la connaissance, permet de ne plus prouver ces faits. Dans notre approche, on considère que l'inexactitude peut être causée par l'incomplétude: la théorie sera spécialisée pour exclure les faits et on peut additionner une hypothèse H à T telle que

$$B, T+H \not\vdash F^- \text{ et } B, T+H \vdash F^+ \quad (2)$$

Ici, on rajoutera une conjonction dans une règle existante, pour *restreindre* la théorie.

Nos modifications veulent préserver la consistance, et le deuxième membre des équations (1) et (2) indique bien que toute hypothèse rajoutée dans la théorie pour l'étendre ou la restreindre se doit d'être limitée à ce seul but. Inclure un fait positif ne doit pas engendrer l'inclusion de faits négatifs, et, similairement, éliminer un fait négatif ne doit pas éliminer des faits positifs.

3 LES MARGINAUX

Cette section introduit la notion de marginal concernant des exemples acceptés ou refusés par une théorie du domaine ou un système expert. Comment devons-nous traiter les marginaux et comment pouvons-nous les utiliser pour reviser une théorie? Ce chapitre essaiera de répondre à ces questions.

3.1 Définition

Un marginal, dans le cadre de cette thèse, est tout exemple qui ne parvient pas à être classé correctement en utilisant la déduction ou l'induction. *Inventer* la règle qui permettra de classer un marginal nécessite un effort particulier. Les marginaux peuvent être causés aussi par le fait qu'ils n'étaient pas prévus dans le système initial: absent des exemples d'entraînements ou ignorés par l'expert. Bref, un marginal est causé par l'ignorance du système. Les règles concernant les marginaux normalement seront ajoutées au système et ne serviront que pour une infime classe d'exemples. Les marginaux sont soit positifs, soit négatifs. Un marginal positif est un exemple positif qui ne peut être accepté par le système. Une *règle étendant* la théorie du domaine doit être trouvée. Un marginal négatif est tout exemple négatif accepté mais qui en réalité devrait être exclu car il n'appartient pas au concept. Une *règle restreignant* la théorie devrait être ajoutée.

Un *marginal positif* ne doit pas être confondu avec les quasi-exemples (near-miss) de Winston. [Winston, Binford et al. 1983]. Ceux-ci sont des exemples négatifs qui ont raté de peu la théorie et Winston les utilise pour focaliser la généralisation dans l'apprentissage par les exemples. Nos marginaux ratent de peu la preuve alors qu'ils auraient dû réussir.

3.2 Où sont-ils?

Selon la méthode d'apprentissage, les marginaux peuvent être associés aux *small disjuncts* ou aux imperfections d'une théorie.

3.2.1 Programme d'apprentissage inductif (inductive learner)

Souvent dans l'apprentissage par similarité (SBL), le système a de la difficulté à classer certains exemples positifs et une nouvelle disjonction doit être recherchée pour ces cas. Un **marginal positif** dans ce cadre d'apprentissage est un événement positif non couvert par les *disjonctions typiques* (heavy disjuncts). Il faudra donc, pour que la description du concept soit complète, additionner des *disjonctions marginales* (small

disjuncts). Ces disjonctions causent un problème réel [Holte, Acker et al. 1989]. Elles ont un pouvoir de prédiction très faible et sont employées pour une minorité des cas possibles. Holte conclut qu'elles devraient être traitées en employant un parti-pris différent. En effet, pour mieux définir les disjonctions marginales, on devrait tenir compte de l'ensemble des données. Les disjonctions marginales, peu expressives, ne peuvent couvrir des cas peu différents de ceux utilisés pour leur création. Déjà, dans ce cadre de l'induction, Michalski [Bergadano, Matwin et al. 1988] a proposé une méthode pour représenter et apprendre des concepts mal définis ou imprécis. Cette méthode repose sur une représentation étendue sur deux paliers (2-tiered representation). Un premier palier représente les caractéristiques typiques du concept et un second palier manipule les exceptions, les instances non-typiques et les instances dépendantes d'un contexte. Nous étendrons cette méthode pour l'utiliser dans le contexte de la déduction.

De plus, tout ensemble de disjonctions étant imparfait et pouvant couvrir quelques exemples négatifs, alors des marginaux négatifs apparaissent. Souvent, des conjonctions sont ajoutées à la règle pour essayer d'en restreindre l'application. Notre approche, dans sa première passe, résume la connaissance de l'ensemble des données et dans sa deuxième passe utilise cette information pour créer des règles qui pourraient faire partie d'un second palier de la représentation.

3.2.2 Programme d'apprentissage déductif (deductive learner)

Une théorie du domaine peut être facilement incomplète et ne pas couvrir certains cas positifs. De plus, la même théorie, par son incomplétude ou par son imperfection, peut couvrir des exemples négatifs indésirables. Donc, on retrouve les marginaux dans la plupart des théories. Une façon de résoudre ce problème est d'augmenter le langage utilisé, par exemple, en changeant le parti-pris associé au langage. Toute théorie est imparfaite et doit être prête à évoluer. Le monde qui nous entoure, heureusement, n'est pas statique.

Dans certains cas, une théorie peut devenir très complexe si elle veut atteindre une perfection pouvant lui permettre d'expliquer les cas rares et marginaux. Dans un monde réel, il va de soi que la théorie du domaine ait certaines déficiences: la théorie du domaine peut être inadéquate de plusieurs façons. [Ellman 1989] classe les déficiences en quatre types différents:

- Complétude (completeness): une théorie est considérée incomplète si elle faillit à expliquer quelques exemples positifs.
- Consistance (consistency): une théorie qui contient des énoncés qui conduiront à des prédictions logiquement contradictoires sera jugée incohérente.
- Exactitude (correctness): toutes les prédictions de la théorie doivent être correctes. Chaque exemple est classé avec justesse. Une théorie incomplète peut être exacte. Mais souvent, l'incomplétude causera certaines inexactitudes.
- Tractabilité (tractability): une théorie même parfaite, ne serait d'aucune utilité si elle exigeait des ressources illimitées, soit du côté temps d'exécution, soit du côté espace mémoire nécessaire.

Une théorie incomplète ne pourra pas expliquer certains exemples positifs et certains exemples négatifs peuvent être couverts par la théorie incomplète causant l'inexactitude.

3.3 Représentation des connaissances à deux paliers

Nos deux modèles d'apprentissage peuvent profiter d'une représentation à deux paliers (2-tiered representation) suggérée par [Bergadano, et al. 1988]. Le premier palier de la représentation sert à expliquer les cas prototypiques. Le second palier sert à étendre la théorie ou à la restreindre en fonction des cas marginaux. Les auteurs emploient cette représentation dans le cas du modèle inductif d'apprentissage. Mais on peut facilement étendre ce modèle au modèle déductif. La terminologie introduite mentionne la couche BCR et la couche ICI. Le premier palier BCR (base concept representation) contient les propriétés typiques et idéales pour le concept. La couche ICI (inferential concept interpretation) définit les transformations possibles concernant le concept. De plus, ce palier contient des modifications causées par le contexte. Cette nomenclature fut attribuée dans le contexte d'un modèle inductif: le BCR est formé des conjonctions typiques (heavy disjunct) et ICI consiste en une série de règles pour les marginaux. En fait, l'ICI remplace les disjonctions marginales (small disjunct).

Nous étendons cette représentation au modèle déductif: l'ICI reste inchangé et le BCR devient simplement la théorie du domaine. La figure 8 résume cette extension.

Apprentissage Inductif	Apprentissage Dédectif
premier palier: une description compréhensible des disjonctions typiques (h e a v y d i s j u n c t) pouvant couvrir les cas typiques.	premier palier: c'est la théorie générale du domaine pouvant couvrir les cas typiques.
deuxième palier: règles spéciales pouvant couvrir les exceptions ou les marginaux.	deuxième palier: règles spéciales pouvant couvrir les exceptions ou les marginaux.

Figure 8. Représentation des connaissances à deux paliers

Notre travail se spécialise dans la recherche de règles pouvant faire partie du second palier. Pour un apprentissage automatique des règles du second palier, on peut profiter de l'information extraite du premier palier. De plus, une continuelle révision des règles créées pour le deuxième palier peut provoquer la promotion d'une règle du second palier vers le premier. En effet, certaines règles de la deuxième couche peuvent représenter simplement une ignorance de la première couche et non un cas exceptionnel ou marginal. De marginal au premier abord, on passe à un enrichissement des cas typiques.

3.3.1 Caractéristiques d'une représentation à deux paliers

On limite la description contenue dans la première couche à l'aspect compréhensif et enseignable. Ce palier donne les propriétés standards et s'occupe du côté normal et bien connu concernant le concept.

Cette première couche se doit d'être très efficace pour s'apparier aux exemples. Dans le modèle inductif, elle peut s'apparier directement aux exemples. Dans le modèle déductif, elle reposera sur une théorie jugée minimale et plus générale.

Le second palier donne à notre concept une faculté d'adaptation. Le concept, à ce palier, est prêt à évoluer. Il devient un concept flexible et dynamique. Principalement, la seconde couche garde l'explication du marginal. Cette couche pourra contenir des restrictions ou des extensions.

Par exemple, pour une chaise roulante, on peut garder dans la deuxième couche la raison de la marginalité en rapport avec une chaise jugée normale selon la première couche: les roues rendent les pattes inappropriées. De même, le pingouin ne peut voler à cause de son poids et de ses ailes minuscules. La seconde couche permet de diagnostiquer la différence concernant le sujet marginal.

3.3.2 Genre de règles dans le deuxième palier

Selon qu'on doit inclure ou exclure un marginal, deux sortes de règles peuvent se retrouver dans le second palier:

a) règles étendantes

Ce genre de règle permet à un exemple positif de devenir membre du concept tout en expliquant pourquoi on doit l'inclure. Cette explication doit également se servir de la connaissance du premier palier pour être pertinente.

b) règles restreignantes

Ce genre de règle permet à un exemple négatif d'être exclu du concept tout en expliquant pourquoi on doit l'exclure. Cette explication doit également se servir de la connaissance du premier palier pour être pertinente.

Il faut se rappeler que ces règles, si elles permettent à la théorie du domaine d'être mieux définie, pourraient être incluses dans la première couche. En effet, un prototype dont les propriétés sont mieux définies dans la première couche discriminera mieux certains exemples négatifs. À ce moment, on peut parler de la promotion d'une règle de second palier vers le premier palier.

3.4 Classification des exemples pendant l'apprentissage

Toutes les instances peuvent être classées selon qu'elles appartiennent ou non au concept et selon qu'elles doivent être incluses ou exclues du concept. Il y aura donc quatre classes.

CLASSE I: instances positives typiques (*PROTOTYPES*). Elles ne doivent jamais être exclues.

CLASSE II: marginaux positifs. Le concept doit être étendu à ces instances.

CLASSE III: instances négatives typiques. Elles ne doivent jamais être incluses.

CLASSE IV: marginaux négatifs. Le concept doit être modifié pour les exclure.

Voici un tableau résumant la situation des instances lors de l'apprentissage.

	sorte d'exemples	couvert par la théorie	action
classe I	+	oui	
classe II	+	non	étendre la théorie
classe III	-	non	
classe IV	-	oui	restreindre la théorie

Table 2. Classification des exemples par LAP

Pour une instance de la CLASSE II qui a échoué à une règle, on doit ajouter une règle sans introduire d'inexactitude (*incorrectness*). Cette règle doit être assez générale pour étendre la théorie, mais ne doit jamais introduire d'instance faisant partie d'un sous-ensemble de la CLASSE III ayant échoué au même bloc. L'échec à une règle peut servir d'*index* pour retracer les exemples et contre-exemples servant à l'élaboration de la règle.

Pour une instance de la CLASSE IV, on doit essayer de l'exclure sans pour cela exclure aucune instance de la CLASSE I qui fournit les contre-exemples pour ce genre d'action. On doit ajouter à une règle existante une conjonction qui permet d'exclure ces marginaux tout en maintenant les prototypes dans la théorie, ou on se doit d'imposer une restriction générale pour ce genre d'instances.

3.5 Marginaux et les deux modèles en apprentissage

La présence des règles de la théorie du domaine est la différence principale entre les deux modèles d'apprentissage. Le fait d'utiliser une théorie du domaine donne au modèle déductif un grand avantage. Sa connaissance est structurée et hiérarchisée. En effet, des règles regroupent sous le nom du conséquent plusieurs antécédents pour donner à leur conjonction une signification sémantique.

Prenons l'exemple de Winston concernant la tasse [Winston, et al. 1983]. Une tasse en polystyrène (*styrofoam*) sera un marginal car elle n'est pas bien classée par les deux

modèles. L'échec à reconnaître cette tasse conduira à une analyse plus riche si elle est faite par le modèle déductif. La figure 9 compare les deux modèles.

Dans les deux cas, on pourrait trouver une règle pouvant expliquer le marginal. Dans le cas du modèle déductif, on se sert de la structure pour forger une règle plus générale. Les deux modèles nous renseignent aussi sur la raison du remplacement ici. La poignée est remplacée par le polystyrène (styrofoam). Une couche sémantique nous permettrait de confirmer la pertinence de ce choix.

La figure 9 nous montre comment les deux modèles peuvent en arriver à une nouvelle règle pour le cas simple de la tasse de Winston.

Apprentissage Inductif	Apprentissage Dédectif
conjonction représentant les tasses typiques:	théorie du domaine pouvant reconnaître les tasses typiques:
<pre>cup(x) <- is(x,light) & part_of(x,y) & isa(y,handle) & part_of(x,z) & isa(z,bottom) & is (z,flat) & part_of(x,w) & isa(w,concavity) & is(w,up)</pre>	<pre>cup(x) <- stable(x) & liftable(x) & open_vessel(x) stable(x) <- part_of(x,z) & isa(z,bottom) & is (z,flat)</pre>
Le marginal sera décrit par un ensemble de prédicats qui sont les mêmes dans les deux modèles (ici une tasse en polystyrène): <pre>is(x,light) & made(x,styrofoam) & part_of(x,z) & isa(z,bottom) & is (z,flat) & part_of(x,w) & isa(w,concavity) & is(w,up)</pre>	<pre>liftable(x) <- is(x,light) & part_of(x,y) & isa(y,handle) open_vessel(x) <- part_of(x,w) & isa(w,concavity) & is(w,up)</pre>
Analyse de l'échec: 1.raison de la faillite: <pre>part_of(x,y) & isa(y,handle)</pre>	Analyse de l'échec: 1.raison de la faillite: liftable(x) avec ces subtilités: <pre>-- is(x,light) est prouvé dans liftable(x) -- part_of(x,y) & isa(y,handle) est la cause de l'échec de liftable(x)</pre>
2. la nouveauté (cas simple): <pre>made(x,styrofoam)</pre>	2. la nouveauté (cas simple): <pre>made(x,styrofoam)</pre>
3. règle explicative: <pre>replace(part_of(x,y) & isa(y,handle) <-- made(x,styrofoam))</pre>	3. règle explicative: <pre>alternative(liftable(x) <-- is(x,light) & made(x,styrofoam)) ou replace(part_of(x,y) & isa(y,handle) <-- made(x,styrofoam))</pre>

Figure 9. Les marginaux versus les deux modèles d'apprentissage

3.6 Heuristiques générales pour traiter les marginaux

Les principales heuristiques employées pour l'analyse conduisant à de nouvelles règles sont construites en fonction de l'apprentissage chez un humain.

La différence (la nouveauté) peut expliquer l' échec d'un marginal positif. Dans le cas de la tasse en polystyrène, si elle faillit la théorie du domaine, on cherchera dans notre marginal, une *différence* qui pourra causer l'explication de l'échec. La règle faillie peut nous servir à indexer nos marginaux positifs ensemble.

La différence entre la théorie et le marginal négatif peut nous aider à l'exclure. Pour un marginal négatif, il y a moins d'information recueillie. En effet, il n'y a pas d'échec lors de la construction de la preuve. Si on doit l'exclure, il faut se baser sur un prédicat nouveau et absent de la théorie actuelle. Notre passe déductive ne nous aide en aucun cas à trouver le point faible dans la théorie.

Avec un seul événement, on ne peut pas généraliser une bonne règle pour nos marginaux. On peut certes s'attendre à trouver l'explication chez les prédicats nouveaux appartenant à ce marginal. Le problème est de s'assurer de focaliser très rapidement sur le prédicat intéressant. Pour cela, on misera sur une discrimination des prédicats susceptibles d'être de bons choix par rapport aux prédicats appartenant aux prototypes.

En effet, il y a de fortes chances de trouver un prédicat comme *made(x,styrofoam)* seulement chez le marginal. Les cas typiques seront en plastique, en porcelaine...

3.7 LAP et les marginaux

Pour étudier les marginaux, en fonction de l'analyse que nous venons de faire, il est préférable de travailler avec une théorie du domaine car les règles produites pour les décrire seront plus riches et permettront d'avoir une description plus appropriée. En effet, cette théorie servira de parti-pris déclaratif pour créer notre règle. Notre pas inductif sera plus susceptible de fonctionner et d'avoir un meilleur pouvoir de prédiction. On veut certes traiter les marginaux, mais tout en suggérant des réparations à la théorie.

3.7.1 Une méthode mixte pour aider un expert à réviser une théorie

Deux étapes importantes et indépendantes seront utilisées pour la création des règles: la classification des prédicats (déduction) et la comparaison avec les prototypes (induction).

L'emploi d'un parti-pris **déclaratif** (théorie du domaine) permettra de mieux classer les prédicats de l'exemple utilisés ou non par la théorie. Cette classification des prédicats se fait grâce à l'information riche de la théorie du domaine et nous permet de créer une règle dans un milieu riche où l'information fut catégorisée.

De plus pour un marginal positif, l'indexation se fait sur le **conséquent** responsable de l'échec. Trouver dans la **différence** l'explication concernant le marginal limite de beaucoup la recherche.

Pourquoi un parti-pris déclaratif?

Si les exemples sont une représentation du concept, ils ont un cadre en commun et la théorie peut résumer ce cadre. La théorie représente donc les prototypes. Cette information concentrée chez les prototypes, est très utile pour traiter un marginal. En effet, un marginal se doit d'être marginal contre quelque chose. Un marginal doit lutter contre un modèle et la théorie du domaine résume ce modèle.

La déduction utilise la théorie pour répondre à la question "Est-ce un marginal ou non?" selon la classification introduite précédemment. Tout exemple positif, non couvert par la théorie, sera étiqueté marginal de *classe II* et tout exemple négatif, couvert par la théorie, sera étiqueté marginal de *classe IV*. De plus, cette déduction classe la connaissance en se servant de la théorie. Pour les prototypes, elle classe les prédicats non utilisés (à cause d'une extension ou non du langage) dans la table des prototypes. Pour les vrais négatifs, elle indexe leur échec sur le conséquent de telle ou telle règle. Pour les marginaux positifs, elle trouve quelle règle doit être réparée et prépare les prédicats pouvant contenir l'explication. Pour les marginaux négatifs, elle prépare les prédicats pouvant permettre d'exclure cet exemple.

Un second parti-pris, empirique celui-là, aidera à élaguer rapidement la règle. Si la table des prototypes est prête, on pourra discriminer instantanément en se servant de l'information accumulée dans la table pour justifier notre règle. L'induction se fait ici en

essayant de trouver en quoi l'exemple est un marginal. Le parti-pris déclaratif ayant classé l'information, le **pas inductif** est fait plus rapidement.

3.7.2 Un exemple simple pour expliquer notre méthode: la tasse

Utilisons l'exemple bien connu de Winston [Winston, et al. 1983] [Mooney and Ourston 1989] [Bergadano and Giordana 1988] pour illustrer ce que doit faire le système LAP. La théorie de LAP sera présentée en détail aux chapitres 4 et 5 et ce même exemple servira à expliquer la méthode. Pour les besoins de l'application, reformulons la théorie du domaine sous forme de prédicats représentant des paires attribut-valeur (figure 10). L'exemple est assez petit pour pouvoir très bien visualiser les difficultés liées à la revision de la théorie.

Cup(x)	<--	liftable(x) & stable(x) & open_vessel(x)
liftable(x)	<--	weight(x,light) & part_of(x,handle)
stable(x)	<--	part_of(x,bottom) & is(bottom,flat)
open_vessel(x)	<--	part_of(x,concavity) & is (concavity,up)

Figure 10. Théorie du domaine pour l'exemple de la tasse

Imaginons que nous jugeons cette théorie incomplète et que nous voulons étendre le langage de la théorie du domaine pour avoir les prédicats suivants: *made(x,?)*, *width(x,?)*, *placed(handle,?)* et *height(x,?)*. En gros, la théorie actuelle devrait muter pour recevoir de nouveaux marginaux et aussi pour donner des règles générales sur les mesures liées à la tasse, i.e. trouver des règles pour tous les exemples afin de remplir les manques de la théorie, ces manques étant causés par l'emploi d'un langage limité (biais).

Les marginaux.

Les figures 11 et 12 représentent deux marginaux mal classés par la théorie originale: une "tasse en polystyrène" ("styrofoam") et un petit seau métallique.

La théorie actuelle ne peut accepter la tasse en polystyrène: c'est notre marginal positif.

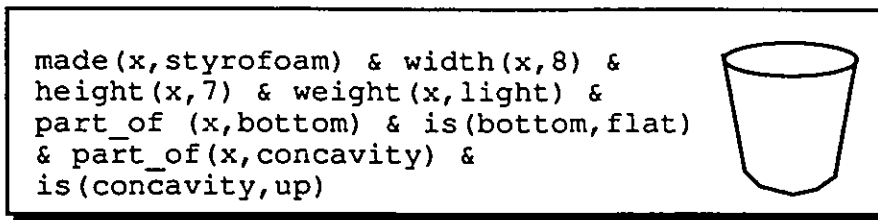


Figure 11. Tasse en polystyrène, un marginal positif

Le petit seau répond à tous les critères d'une tasse de la théorie incomplète du domaine. Cette théorie sera incorrecte en classifiant ce seau dans la classe des tasses. On a donc notre marginal négatif.

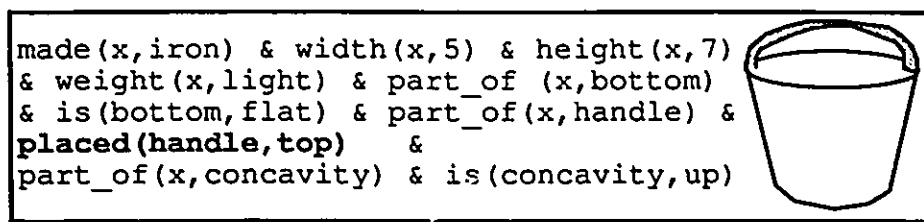


Figure 12. Petit seau, un marginal négatif

Il faut créer une règle pour le marginal positif afin de l'inclure dans le concept. Cette règle devrait expliquer comment la tasse peut être levable. De plus, cette règle pourrait être placée au premier palier de la représentation (figure 13).

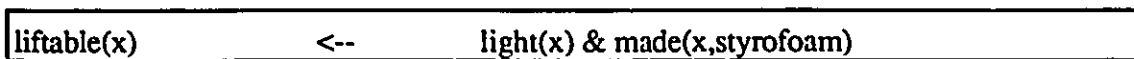


Figure 13. Règle pour inclure le marginal "styrofoam cup"

Comment allons-nous pouvoir aller chercher *light(x)* de l'ancienne théorie et choisir *made(x, styrofoam)* parmi les nouveaux prédicats associés à l'extension du langage? Le conséquent de la règle sera *liftable(x)*, comment faisons-nous pour choisir celui-là parmi tant d'autres? Une connaissance d'arrière-plan (BK) pourrait nous aider à le faire facilement, mais on veut miser sur une méthode jugée faible.

Le marginal négatif causera davantage de problèmes. Deux règles seront possibles pour le premier palier de la représentation (figure 14).

liftable(x)	<--	light(x) & part_of(x,handle) & placed(handle,side)
ou		
liftable(x)	<--	light(x) & part_of(x,handle) & not placed(handle,top)

Figure 14. Règle du premier palier pour exclure le marginal "petit seau"

Ces deux règles restreignent le concept et elles permettent de refuter le seau. Cependant la sémantique de la deuxième est fausse: on ne peut pas miser sur une anse placée sur le sommet pour renier le fait qu'un objet ne peut être levé. Une restriction pour le concept tasse, placée dans un second palier de la théorie du domaine, serait plus souple et moins compromettante (figure 15).

restriction(cup(x))	<--	part_of (x,handle) & not placed(handle,top)
ou		
restriction(cup(x))	<--	part_of (x,handle) & placed(handle,side)

Figure 15. Règle du second palier pour exclure le marginal "petit seau"

Ce genre de restriction devrait toucher seulement la tasse. D'autres objets levables peuvent avoir leur anse sur le dessus et pourrait se servir de notre théorie du domaine. La restriction se doit d'être spécialisée à la tasse. Il va de soi qu'une méthode faible et syntactique ne peut avoir ce raffinement dans les règles suggérées, mais le fait qu'elle puisse pointer du doigt le bon terme "*placed(handle,side)*" ou "*placed(handle,top)*" parmi une gamme de prédicats serait déjà un atout majeur pour une couche sémantique ou un expert (usager) qui aiderait à l'aspect sémantique de la règle.

Règles générales pour tous les exemples positifs.

Si notre méthode peut juger que la largeur et la hauteur de la tasse peuvent faire partie d'une règle commune à tous les prototypes, la théorie du domaine serait plus viable.

Si une BK est présente, rien ne nous empêcherait d'ajouter une autre règle pointant le fait que tous les matériaux employés sont fait d'une matière non poreuse. Les deux dernières règles ne seraient que le résultat normal d'un algorithme d'induction connu (Version Space [Mitchell 1982], ID3 [Quinlan 1986], Induce [Dietterich and Michalsky 1981]) utilisant les prédicats contenus dans l'extension du langage et utilisant un hiérarchie de types (figure 16).

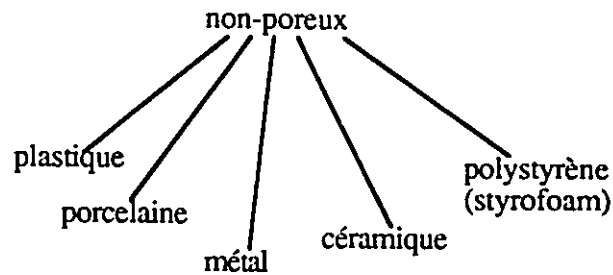


Figure 16. Hiérarchie de type utile pour l'induction (BK)

LAP montre comment une méthode faible est utile pour traiter des marginaux vis-à-vis des prototypes. Remplacer les manques dans une théorie trop générale est un problème déjà traité dans IOU [Mooney and Ourston 1989]. IOU se sert d'algorithmes inductifs pour trouver des restrictions concernant le concept élaboré à partir d'une théorie trop générale. LAP pourra trouver certaines règles générales mais il ne généralisera que pour les attributs ayant des valeurs numériques car aucune BK n'est nécessaire pour ce genre d'induction.

3.7.3 Le système LAP

La figure 17 représente le système LAP (Learning by Analysing the Proof). Le module de déduction se sert de la connaissance de la théorie du domaine pour grouper l'information. Le module d'induction, grâce à cette information génère une règle plus efficace, et cela, de façon plus rapide pour chaque marginal qu'on lui présente.

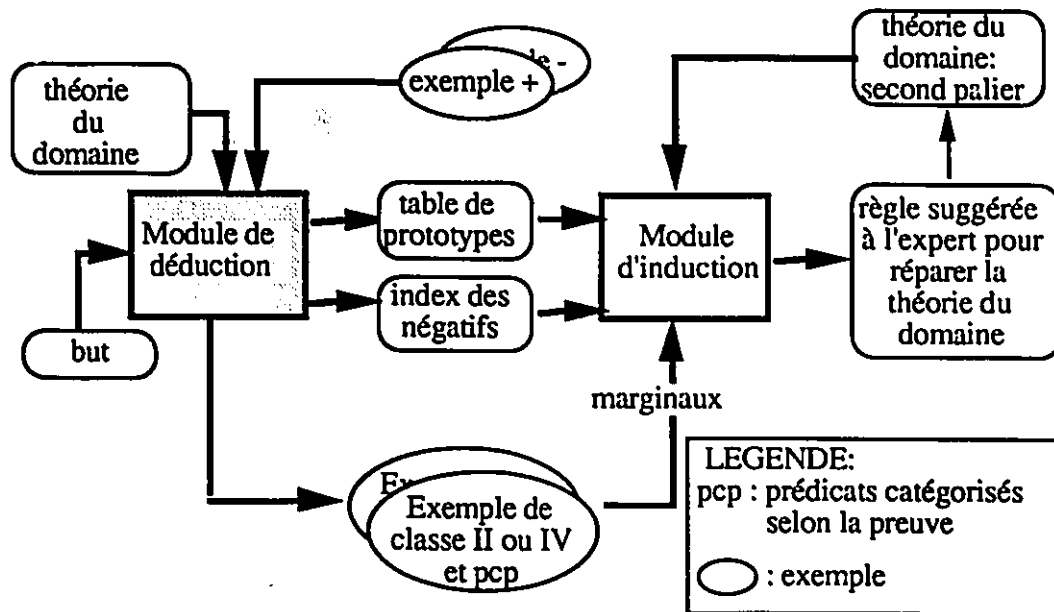


Figure 17. Système LAP

LAP créera des règles qu'il placera dans le second palier de la représentation. Il ne décidera en aucun temps de promouvoir une règle du second palier vers le premier. Il propose des règles et des restrictions, mais seul l'expert jugera de leur pertinence. L'expert pourra en tout temps réviser sa théorie du domaine après avoir consulté les suggestions de LAP; LAP est un outil de travail.

4 PRÉDICATS CLASSIFIÉS SELON LA PREUVE: PCP

Ce chapitre nous présente la partie déductive du système LAP, i.e. son analyse de la preuve lors de la réussite ou de l'échec de l'exemple traité par la théorie.

4.1 Nomenclature

Peu importe la classe de l'exemple traité, le module de déduction utilisera la théorie du domaine et ses règles pour analyser le plus en détail possible la réussite ou l'échec de la preuve démontrant que l'exemple satisfait la théorie.

Un exemple d'entraînement est décrit par une conjonction de prédicats de bas-niveaux. L'ensemble pouvant contenir tous les prédicats qui peuvent décrire les exemples se dénote P_{low} . Souvent, ces prédicats sont jugés opérationnels selon un critère d'opérationnalité. L'instance I sera représenté par un ensemble $\{p_1, p_2, \dots, p_i \dots p_n\}$ où $p_i \in P_{low}$. Ainsi les feuilles de l'arbre représentant la théorie sont des prédicats appartenant à P_{low} . Le refus d'un exemple par la théorie se base sur le fait que les prédicats de I ne peuvent apparier tous les prédicats faisant partie des feuilles de l'arbre de la preuve: certaines feuilles instantiées ne se retrouvent pas membre des prédicats de I .

Quand un prédicat de la théorie résume une conjonction de prédicats, on représente ce prédicat par p : c'est le conséquent de la règle. Normalement, à ce conséquent, se greffe une signification sémantique. La théorie du domaine est représentée par T . Cette théorie T est formée de deux sortes de prédicats. Les prédicats de haut-niveau sont des conséquents d'une règle. Les prédicats de bas-niveau sont les prédicats qui peuvent se retrouver directement dans les exemples.

Si la règle $p \leftarrow p_1 \& p_2 \& \dots p_n$ infère l'appartenance d'une instance I à la théorie T , on emploiera la notation suivante $ANTC(I, p)$ pour désigner l'ensemble $\{p_1, p_2, \dots, p_n\}$, i.e. l'ensemble des prédicats utilisés et instantiés par I . Il est à remarquer que ces prédicats peuvent être des prédicats appartenant à P_{low} ou des conséquents de règles membres de la théorie. L'ensemble $ANTC$ dépend de I , car chaque exemple spécifique I influencera les instantiations des prédicats formant l'arbre de la preuve prouvant que I appartient au concept. En conséquence, on dénotera $T(I)$ toutes les instantiations des prédicats de *bas-niveau* apparaissant dans l'inférence de I à partir de T . En somme, $T(I)$ représente les feuilles de la théorie T qui peuvent être directement appariées avec des prédicats contenus dans I .

Si $p(A)$ représente une instantiation du prédicat p , alors on dénotera par $ARGS(p(A))$ la liste de tous les arguments de p instantiés. Par exemple, $ARGS(placed(handle,top)) = \{handle,top\}$. De façon similaire, $FUNC(p(A))$ dénotera le terme p . Par exemple, $FUNC(placed(handle,top)) = placed$.

4.2 Prédicats catégorisés par la preuve (PCP)

4.2.1 Définitions

L'analyse de l'arbre de preuve nous permet d'introduire cinq catégories de prédicats. L'appartenance d'un prédicat à l'une ou l'autre de ces catégories dépend du rôle joué par ce prédicat dans l'élaboration de la preuve cherchée. Cette preuve représente soit un échec ou une réussite selon qu'on ait réussi ou non à prouver l'appartenance de l'exemple selon la théorie du domaine se rattachant à ce concept donné.

Ces catégories sont:

CP représente la catégorie des prédicats communs (*common predicates*), i.e. l'ensemble de tous les prédicats de l'exemple I employés par la preuve $T(I)$.

$$CP = \{ p_i \mid p_i \in I \text{ and } p_i \in T(I) \}$$

NP représente la catégorie des prédicats non utilisés (*non-used predicates*), i.e. l'ensemble de tous les prédicats de l'exemple non employés par la preuve $T(I)$.

$$NP = \{ p_i \mid p_i \in I \text{ and } p_i \notin T(I) \}$$

FC représente la catégorie des règles coupables (*failure consequents*), i.e. l'ensemble des règles (représentées par leurs conséquents) de l'arbre de preuve qui contiennent un prédicat de bas-niveau (feuille dans la théorie) qui ne peut être prouvé.

$$FC = \{ p_i \mid p_j \in ANTC(I, p_i) \text{ and } p_j \notin I \text{ and } p_j \in T(I) \}$$

FP représente les prédicats de bas niveaux (*failure predicates*) responsable de l'échec d'une règle, i.e. l'ensemble des prédicats appartenant aux feuilles de la théorie et qui ne peuvent être satisfaits par l'exemple.

$$FP = \{ p_i \mid p_i \in P_{low} \text{ and } p_i \notin I \text{ and } p_i \in ANTC(I, p_j) \text{ and } p_j \in FC \}$$

AP représente les prédicats appartenant à une règle coupable et qui se retrouvent prouvés par l'exemple ("*already there*" predicates), i.e. l'ensemble des prédicats appartenant à une règle de FC et qui n'est pas responsable de l'échec de la règle car ils sont satisfaits.

$$AP = \{ p_i \mid p_i \in I \text{ and } p_i \in \text{ANTC}(I, p_j) \text{ and } p_j \in FC \}$$

Tous ces ensembles contiennent les prédicats avec leurs arguments. Quelques observations découlant de ces définitions peuvent se montrer pertinentes.

Les règles non satisfaites par la preuve se regroupent en deux catégories:

1. les règles qui en appellent d'autres qui ne sont pas satisfaites elles-mêmes. Nous ne considérons pas ces règles coupables de l'échec de la preuve. En effet ces règles représentent des noeuds dans l'arbre de preuve qui échouent à cause des noeuds situés plus bas dans l'arbre.
2. les règles qui contiennent des prédicats de bas-niveau. Nous considérons ces règles coupables de l'échec de la preuve. Ces règles représentent des noeuds qui sont les parents directs d'une feuille responsable de l'échec. Ces règles seulement peuvent faire partie de l'ensemble FC.

Tous les prédicats de bas niveau faisant partie de l'instance I se retrouvent soit dans CP ou NP. En effet

$$I = CP \cup NP$$

Tous les prédicats se retrouvant dans FP sont des prédicats faisant partie de T qui furent instantiés par un appel d'une règle p_i lors de la construction de $T(I)$ et qui ne purent être appariés par I. Ces prédicats soulignent une faiblesse de l'ensemble I à prouver une règle de $T(I)$.

4.2.2 Relations entre les ensembles lors d'une réussite de la preuve

Il est à noter que si la théorie accepte l'exemple, alors $T(I)$ est un sous-ensemble de I. On peut résumer la situation grâce à cette série d'équations.

1. $CP = T(I)$
2. $I \cup T(I) = I$ et $I \cap T(I) = CP$
3. $I \setminus T(I) = NP$
4. $FC = FP = AP = \emptyset$.

La figure 18 représente les ensembles formés par les prédicats de bas-niveau pouvant être membre de l'instance I ou de la théorie. La partie gauche de la figure explique par elle-même les équations si la théorie est satisfaite.

4.2.3 Relations entre les ensembles lors d'un échec de la preuve

Si la théorie refuse l'exemple, alors l'ensemble formée par $T(I)$ n'est plus un sous-ensemble de I . Les équations suivantes résument la situation:

5. $CP \cup FP = T(I)$
6. $I \cup T(I) = NP \cup CP \cup FP$ et $I \cap T(I) = CP$
7. $I \setminus T(I) = NP$
8. $FC \neq \emptyset, FP \neq \emptyset, AP \neq \emptyset$
9. $T(I) \setminus I = FP$
10. $AP \cup CP = CP$.

La partie droite de la figure 18 résume encore mieux la situation en cas d'échec.

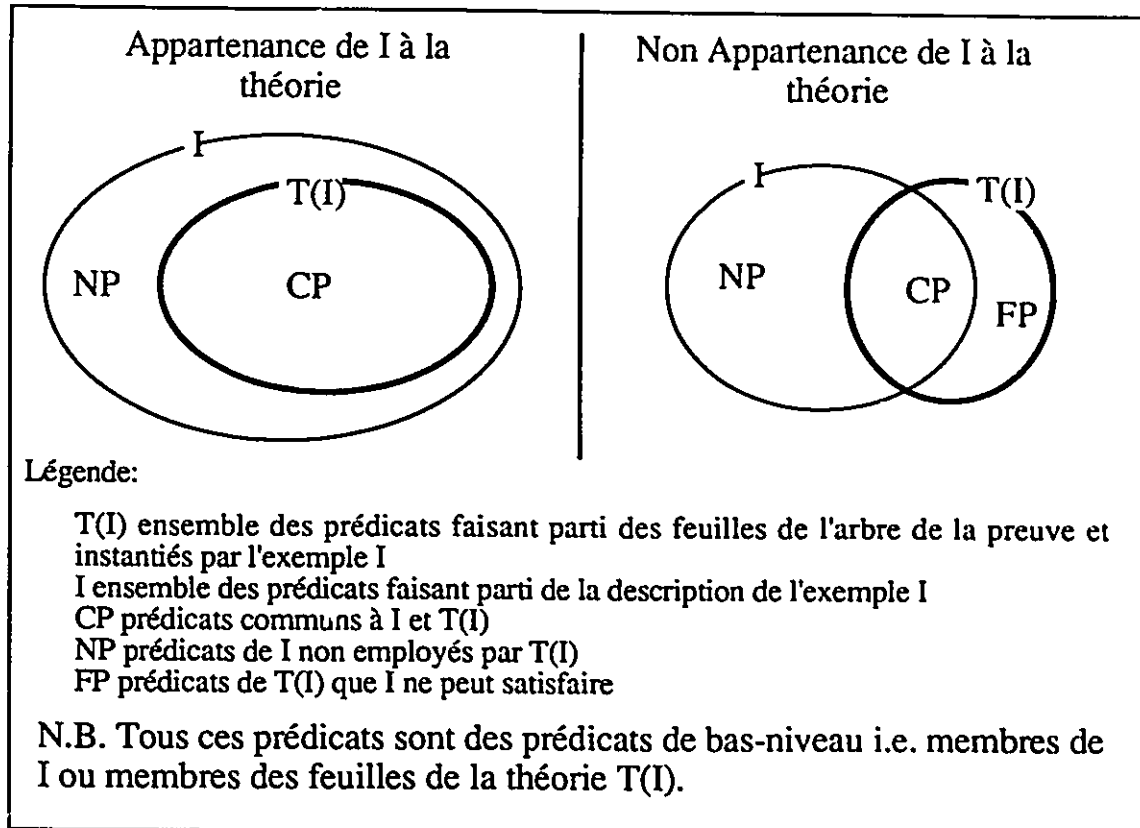


Figure 18. Prédicats de bas-niveau et leur catégorisation par la preuve (PCP)

4.2.4 Observations

Dans tous les cas, on peut noter que quelques équations sont toujours vraies, dont:

11. $NP \cup CP = I$
12. $I \cap T(I) = CP.$

Nous avons observé que l'ensemble T(I) peut être formé de deux sous-ensembles, les prédicats prouvés et ceux coupables de l'échec.

$$T(I) = CP \cup FP \quad (1)$$

et la théorie instantiée T(I) est satisfaite pour I seulement si l'ensemble FP est vide. De plus, l'exemple I répond toujours à cette équation

$$I = CP \cup NP \quad (2)$$

Ces deux équations peuvent fortement nous suggérer les actions à entreprendre pour modifier la théorie. Regardons ce qu'elles nous suggèrent pour nos deux types de marginaux.

MARGINAL POSITIF (classe II)

Rappelons d'abord que pour ce genre de marginal, la preuve échoue mais qu'elle devrait réussir. À partir des équations (1) et (2), deux solutions seulement peuvent être proposées sachant que $T(I)$ devrait devenir un sous-ensemble de I (figure 19) dans le cas d'un marginal positif (classe II). D'abord une solution triviale: les prédicats de FP devraient être présents dans I . Ici, on juge que l'exemple donné est incomplet et mal énoncé. Cette solution doit être rejetée lorsque les exemples sont supposés complets. Une solution adéquate est alors: la théorie est jugée incomplète et on se sert de l'ensemble NP pour raffiner la théorie. L'introduction d'une nouvelle règle utilisant NP fait en sorte que les prédicats de FP ne pouvant pas être satisfaits seront dorénavant inutiles pour cette preuve. En d'autres mots, "*la différence peut expliquer l'échec*". La différence est représentée par NP et l'échec est représenté à la fois par FC (conséquents de la règle fautive) et par FP (antécédents improuvables de la règle). Cette différence fait que ce positif est considéré comme un *near-miss* au sens de Winston [Winston, et al. 1983].

Il faut choisir dans NP les prédicats pouvant faire partie d'une nouvelle règle (disjonction) en vue d'étendre la théorie T_1 . La nouvelle théorie T_2 pourrait accepter ce marginal positif.

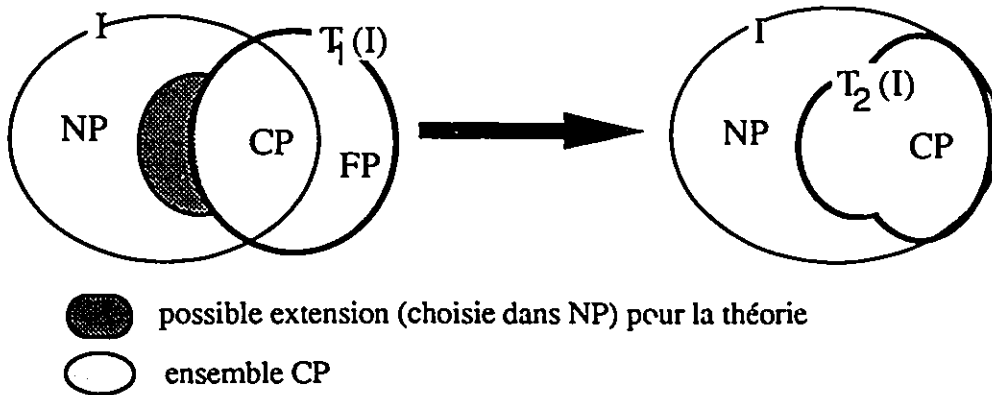


Figure 19. Théorie modifiée pour accepter un marginal positif

La figure 19 représente bien la situation: une nouvelle règle est créée pour instantier un ensemble $T_2(I)$ (différent de $T_1(I)$) qui n'emploierait pas l'ensemble FP de $T_1(I)$. A ce moment, $T_2(I)$ devient un sous-ensemble de I . Il faut remarquer que la partie de $T_1(I)$, extérieure à I , quoique non-instantiée par I , peut et est certainement utilisée par d'autres exemples. La règle introduite pour modifier la théorie, produira une nouvelle instantiation de la théorie: $T_2(I)$. Cette nouvelle règle est une disjonction permettant d'instantier $T_2(I)$ au lieu de $T_1(I)$.

MARGINAL NÉGATIF (classe IV)

Pour un marginal négatif (classe IV), la preuve réussit alors qu'elle devrait échouer. À partir des équations (1) et (2), $T(I)$ qui est un sous-ensemble de I , devrait devenir un ensemble disjoint de I . Le but étant de réparer la théorie jugée incomplète, on pourra se servir de l'ensemble NP pour raffiner la théorie et restreindre sa portée. Il suffit de trouver dans NP un prédicat qui est caractéristique de ce marginal. A priori, une solution simple serait de nier ce prédicat dans la théorie. La théorie serait modifiée pour avoir un ensemble FP qui servirait à exclure ce marginal négatif. Nous verrons que nous pouvons faire mieux que de nier le prédicat jugé pertinent. On emploiera ce prédicat dans la théorie mais avec des arguments propres aux prototypes lorsque cela est possible.

La figure 20 représente la stratégie employée. On modifie une règle de la théorie de sorte que le nouvel ensemble instantié $T_2(I)$ ait des éléments appartenant à FP. Donc, des éléments de NP seront soit niés dans la théorie ou soit instantiés dans la théorie avec des arguments propres aux prototypes.

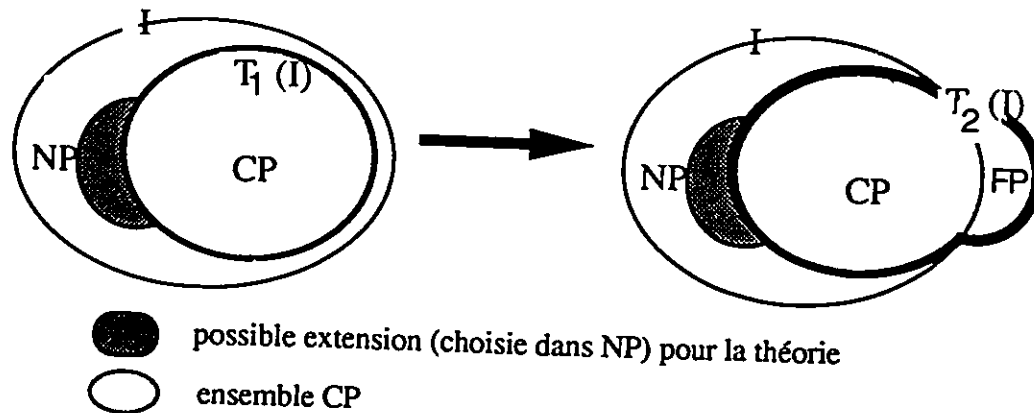


Figure 20. Extension pour restreindre la théorie dans le cas d'un marginal négatif

En bref, pour nos deux types de marginaux, nous voulons simplement passer d'un côté à l'autre de la figure 18. Un marginal positif devra éliminer l'ensemble FP en ajoutant une nouvelle règle. Un marginal négatif devra se construire un ensemble FP en restreignant une règle ou en imposant une restriction dans un deuxième palier de la représentation.

4.3 Analyse lors de la déduction

Quelle est l'utilité de connaître les prédicats (conséquents) utilisés par la théorie, et de savoir quels sont les antécédents réussis ou responsables d'un échec lors de la preuve? Le but est de ramasser de l'information pertinente sur la façon dont la théorie est construite pour, le cas échéant, construire et suggérer une règle dans ce contexte riche en information. L'induction, en employant cette information, pourra probablement nous donner une règle plus pertinente sans employer une connaissance d'arrière-plan.

4.3.1 Analyse lors d'un échec de la preuve

La figure 23 présente de façon générale l'arbre construit lors de la preuve, un but ne pouvant être prouvé pour un exemple (positif ou négatif).

Imaginons que le but peut être décomposé en un ensemble de prédicats non-opérationnelles. La figure 21 nous donne un exemple possible pour le but de la figure 23.

But	<--	p1 & p3
But	<--	p2 & p3

Figure 21. Deux règles disjointes permettant de prouver notre but

De plus, ces prédicats à leur tour peuvent en appeler d'autres. En dernier lieu, les prédicats non-opérationnels se décomposeront en prédicats opérationnels (figure 22).

p4	<--	p8 & p9
p5	<--	p10 & p11

Figure 22. Deux règles arrivant directement aux faits.

La figure 23 ne montre pas toutes les parties de l'arbre. La règle représentée par le conséquent p4 est un exemple d'une règle ayant réussi. La règle p5 est un exemple d'une règle ayant failli. La règle p2 n'est pas jugée responsable de l'échec car elle ne faillit pas à cause des faits mais à cause d'une autre règle. En effet, une réparation le plus bas possible dans l'arbre de la preuve permet de réduire le nombre de règles membres de FC, i.e. les règles à réparer. La réparation d'une de ces règles se propagera vers le haut et sera responsable de la réparation des autres règles ayant échoué à cause d'elle.

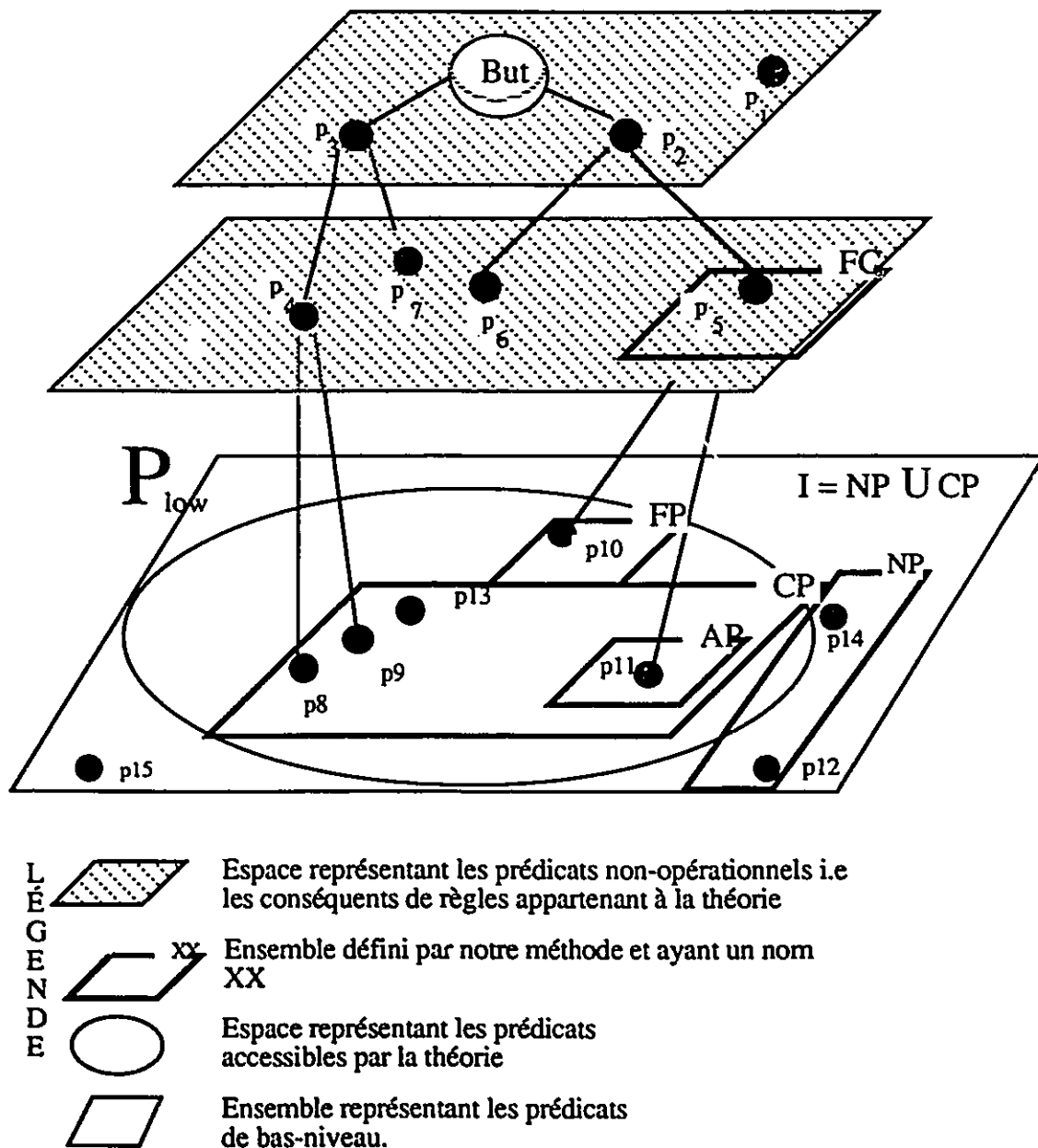


Figure 23. Arbre de preuve pour un échec de la preuve et PCP

Supposons, pour la clarté du dessin, que nous ayons seulement trois niveaux de règles et un niveau de faits pouvant être appariés de l'exemple à la théorie, et vice-versa.

La preuve du but échoue en se servant des règles p_3 et p_2 puisque p_2 faillit. Notre algorithme procède à l'étude en parallèle de toutes les possibilités et garde selon certains critères l'arbre ayant le moins de brisures. Ici, c'est l'arbre représenté. Les autres arbres examinés, s'ils existent, ont failli et seraient jugés plus difficiles à réparer. L'algorithme

mime un parallélisme et traite donc les disjonctions. Il retourne pour chaque niveau de règles le chemin le plus simple à réparer. Par exemple, il peut avoir essayé en parallèle deux règles pour p_3 et il se sert de celle qui a réussi. Notons que p_1 et p_3 auraient pu potentiellement servir pour prouver le but grâce à une autre règle concernant le but, mais cette seconde règle a failli aussi et s'est révélée plus difficile à réparer.

A la figure 23, nous avons trois niveaux de règles dans la théorie du domaine. L'ensemble FC est toujours un ensemble se situant au niveau des règles. Il représente les conséquents qui ne purent être prouvés.

L'ensemble AP peut être à n'importe quel niveau, mais un élément de AP est toujours un fils d'un élément de FC. Donc AP peut contenir un conséquent d'une autre règle et/ou peut contenir un fait. Par conséquent, les membres de AP peuvent être répartis sur plusieurs niveaux. Dans notre exemple, AP contient le fait p_{11} . Cette partie de la règle n'est pas responsable de l'échec. Si nous voulons réparer cette règle par une disjonction, cette partie pourrait être utilisée en supposant que la réparation veut introduire à cet endroit une règle remplaçant celle qui a échoué.

L'ensemble FP est toujours au niveau des faits. Il représente la raison première de l'échec d'un élément de FC: c'est le fait (l'antécédent) responsable de l'échec du conséquent. En effet, il est impossible de satisfaire cette feuille car l'instance I ne contient pas ce fait.

Le dernier niveau est celui des faits de l'instance ou des prédicats ne pouvant pas être prouvés par une règle. Donc, ces prédicats ne sont jamais des conséquents d'une règle, mais toujours des antécédents. L'ensemble CP représente tous les prédicats utilisés par la preuve et présents chez l'exemple. Certains prédicats peuvent être instantiés dans les antécédents des règles et, si absents de l'exemple, ils causent l'échec de la preuve. Pour la règle p_5 par exemple, le prédicat p_{10} nécessaire à cette preuve est absent des faits de l'exemple. Donc, p_{10} est membre de FP.

D'autres prédicats peuvent faire partie de l'exemple et ne pas être utilisés par la théorie instantiée $T(I)$. Ces prédicats normalement sont jugés hors propos car ils ne sont pas atteints par la théorie. Les prédicats p_{12} et p_{14} de la figure 23 en sont des exemples. Finalement, certains prédicats de bas niveau (p_{15}) ne sont utilisés ni par la preuve, ni par l'exemple.

Discutons maintenant de la réparation de la théorie présentée à la figure 23 pour inclure un exemple positif de classe II (figure 24).

Il faut que l'exemple positif puisse contenir des prédicats jugés utiles pour cette réparation. Supposons toujours une extension du langage. Donc en plus des prédicats hors propos, certains autres prédicats ne furent pas utilisés dans la preuve. NP est l'ensemble de ces prédicats. Il peut contenir des prédicats rajoutés à cause d'une extension du langage (p₁₄) en vue d'étendre la théorie ou des prédicats hors de propos (irrelevant) (p₁₂).

Notez bien que la théorie cible et la nouvelle théorie étendue sont représentées dans l'espace couvert par les prédicats du langage. On travaille au niveau de l'espace des hypothèses et non dans l'espace des exemples couverts ou non couverts par la théorie. En effet, certaines extensions du langage peuvent servir à éliminer certains exemples négatifs. Donc, un langage plus étendu peut aussi servir à restreindre la portée d'un concept.

Cette analyse est construite pour tous les cas, mais l'interprétation en est différente selon que l'on ait affaire à un cas positif ou à un cas négatif. En effet, l'information qu'on peut extraire dépend si on veut réparer ou non la théorie.

Échec de la preuve pour un exemple de classe II (marginal positif):

Connaissant les causes et les détails de l'échec, on pourra essayer de réparer la théorie pour ce cas particulier de marginal. On ne veut pas trouver une règle commune à tous les prototypes comme le fait IOU [Mooney and Ourston 1989]. En effet, IOU ne considère pas le cas d'exemples positifs ayant échoué la théorie du domaine. On veut récupérer ce marginal. Donc, on veut introduire une règle qui aide à couvrir ce genre de marginal, et non tous les prototypes. IOU veut restreindre une théorie trop générale qui accepte des cas négatifs et ne travaille qu'avec les prototypes.

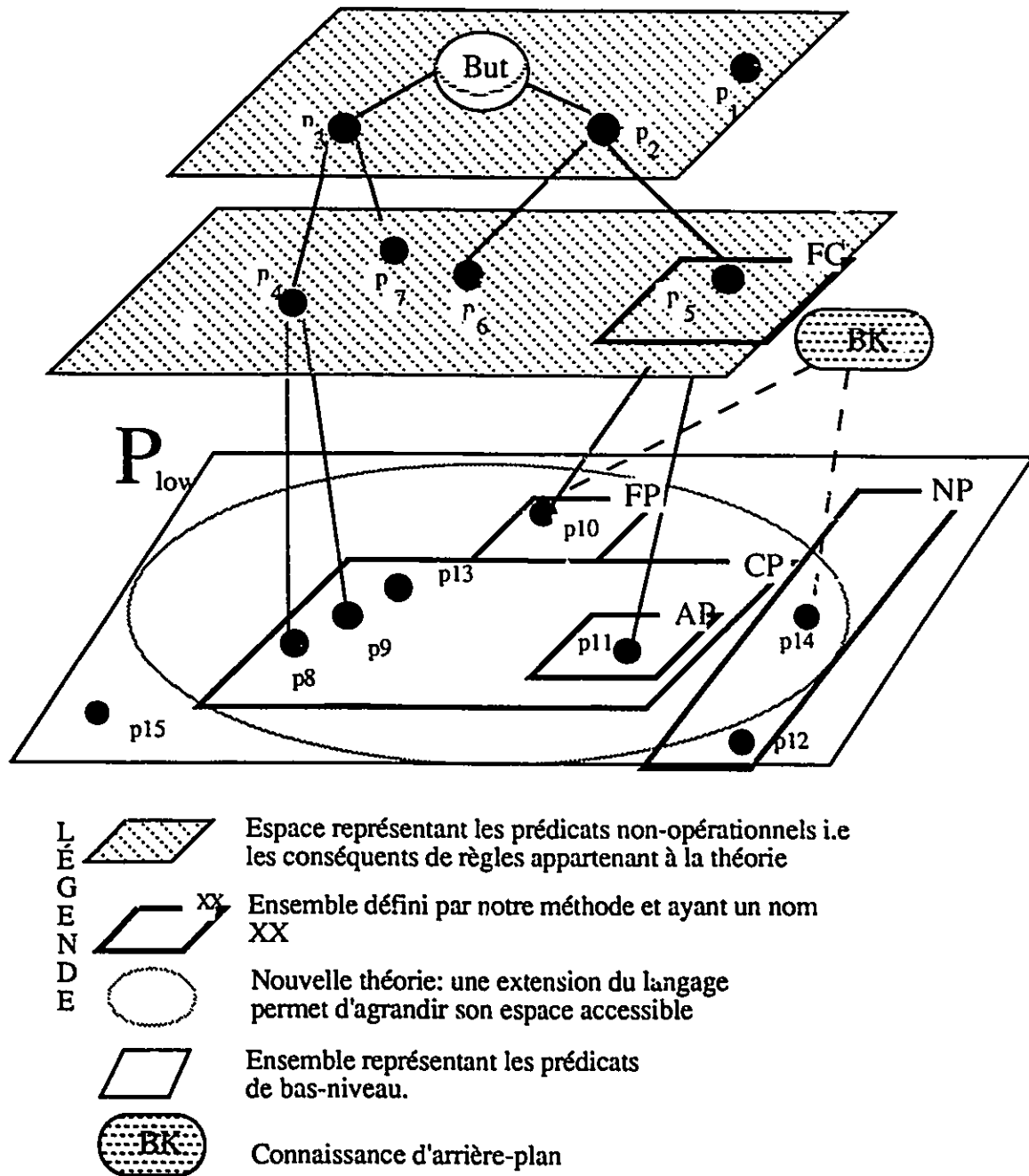


Figure 24. Réparation de la théorie pour accepter un marginal positif (classe II)

Pour réparer la théorie, on se base sur la catégorisation des prédicats (PCP). La règle suggérée sera composée des prédicats de AP et des prédicats de NP susceptibles de bien remplacer le rôle joué par ceux compris dans FP. On aurait pu suggérer une règle contenant uniquement des prédicats de NP. Notre méthode étant syntactique, produire une telle règle indiquerait un manque total de confiance en l'ancienne théorie. La règle produite

pourrait être trop générale pour être efficace et utile. C'est pourquoi nous jugeons que le rôle joué par l'ensemble AP est de première importance: il permet à la méthode syntactique de prendre ce qu'il y avait de meilleur dans la règle défectueuse. En utilisant AP, nous croyons que la règle produite sera plus efficace et qu'elle sera plus proche de la réalité. La méthode syntactique repose sur le rôle sémantique joué par AP pour augmenter la confiance en la nouvelle règle produite.

Supposez qu'une méthode non faible puisse nous renseigner sur le fait que p_{10} avait un but particulier. Un réseau sémantique par exemple, pourrait nous indiquer le fait que p_{14} peut jouer le même rôle que p_{10} . Cette connaissance d'arrière-plan nous fournirait le moyen de discriminer parmi tous les prédicats compris dans NP. Ainsi, la nouvelle règle aurait comme antécédents la réunion de ces deux prédicats: p_{11} appartenant à l'ensemble AP et p_{14} qui est jugé le meilleur de NP pour remplacer p_{10} .

Si on veut introduire la réparation au premier palier (*first-tier*) de notre représentation, deux possibilités se présentent quant au conséquent de la règle. La règle joue le même rôle sémantique que la règle à réparer. A ce moment, une disjonction est introduite dans la théorie du domaine, puisque le conséquent de la nouvelle règle est le même que le conséquent de l'ancienne règle. La réparation se propage alors à deux ou plusieurs niveaux: le niveau où se trouve FC et les niveaux des éléments de AP qui sont automatiquement placés dans cette nouvelle règle (figure 25 a).

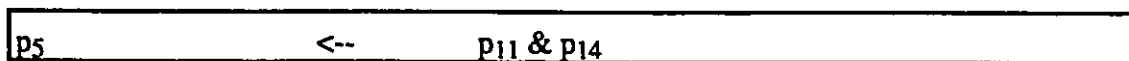


Figure 25a. Règle réparant la théorie au premier palier

Si malheureusement le rôle sémantique joué par la règle est jugé différent par l'expert, alors on introduit un nouveau conséquent (son nom peut être suggéré par l'usager en fonction de son rôle sémantique) et on duplique la règle "père" de l'élément dans FC en y remplaçant le conséquent fautif par le nouveau conséquent (figure 25 b). À ce moment, la réparation se propage à trois ou plusieurs niveaux: le niveau où se trouve FC, le niveau des éléments de AP, et le niveau "père" de FC. Il y a aussi possibilité de réparation récursive vers le haut de l'arbre de preuve.

Pnom_usager	<--	p11 & p14
p2	<--	p6 & pnom_usager

Figure 25b. Règles, créées en cascade, réparant la théorie au premier palier

Si on répare au second palier (*second-tier*), on n'a qu'à noter la raison de l'échec (FP et FC) et la façon de réparer cet échec. L'expert pourra, à partir de ces informations pertinentes, juger s'il doit promouvoir l'extension au premier palier.

La réparation de ce genre est vraiment particulière à ce marginal. Elle ne touche aucunement l'ensemble des prototypes. Le chapitre suivant vous renseignera sur l'introduction d'une méthode faible pour nous aider à choisir les prédicats dans NP sans se servir de BK (*background knowledge*). Il faudra à ce moment que le marginal n'échoue qu'une règle. La marginalité pour nous signifie une instance qui manque de peu le concept.

Si un marginal faillit plusieurs règles, c'est plutôt une exception grave. Il faudrait à ce moment une BK pour discerner les différents rôles joués par les différents prédicats dans FP. En effet, comment discerner dans NP les prédicats pouvant réparer une règle en particulier parmi plusieurs règles? Le système LISE [Genest, et al. 1990] réussit à bien réparer plusieurs règles en même temps car il peut associer le même but à la partie manquante d'une règle et à son remplaçant. Ce but lui sert de guide pour choisir le bon remplaçant parmi plusieurs.

Un autre but peut être poursuivi en étudiant ces marginaux. Rien ne nous empêche de faire de l'induction sur NP pour tous les exemples positifs comme dans IOU. On pourra alors trouver des restrictions pour notre concept. IOU ne touche pas à la théorie. LAP pourra trouver une nouvelle règle. A ce moment, on trouve qu'il y avait un manque dans la théorie et on décide de la réparer. Sans présence de BK, la règle aura un nom suggéré par l'utilisateur et conduira directement aux faits. Une BK peut même donner un nom pour cette règle (version space, ID3) et donner plusieurs niveaux de règles en se servant des noms placés dans un réseau sémantique ou dans une hiérarchie de types.

Échec de la preuve pour un exemple de classe III (négatif):

Pour un exemple négatif, il est normal d'avoir un échec. On recueillera alors l'information et indexera ces vrais négatifs en se servant des conséquents des règles échouées, pour pouvoir retrouver rapidement ces négatifs. En effet, si on ajoute ou répare une règle pour inclure un marginal positif, il faudra s'assurer que cette règle ne permet pas à certains vrais

négatifs d'être admis dans la théorie. Donc, on vérifiera si les vrais négatifs associés à chaque membre de FC peuvent satisfaire cette nouvelle règle. Si oui, il faudra discriminer davantage; sinon la règle est suggérée telle quelle.

Donc, seuls les vrais négatifs ayant failli une règle seront retracés si on essaie de réparer cette règle.

4.3.2 Analyse lors d'une réussite de la preuve

Lors du succès de la preuve, l'arbre de preuve est complet et les ensembles FC, AP et FP sont vides. La seule information ayant une certaine importance sont les ensembles NP et CP. Créer des contraintes dans la théorie du domaine sera plus difficile que d'étendre la théorie pour des marginaux positifs. En effet, le monde à exclure est vaste et moins bien structuré que le monde à inclure.

Réussite de la preuve pour un exemple de classe I (prototype):

Pour un prototype, la preuve est réussie et c'est normal. Cependant, il est possible qu'il y ait une faiblesse dans la théorie. Par exemple, il se peut que la théorie soit incomplète et que de nouvelles caractéristiques communes puissent être trouvées pour former de nouvelles règles pour tous ces cas prototypiques du genre de ceux rencontrés dans IOU. Dans ce cas, l'ensemble NP est utilisé pour construire une table des prototypes qui recueillera l'information minimale commune à tous les prototypes et non utilisée par la théorie présente. Tous les prédicats communs à chaque prototype se retrouveront dans cette table et une liste d'arguments y sera conservée. Pour des arguments ayant des valeurs entières, on gardera les valeurs extrêmes, i.e. la valeur minimale et la valeur maximale. Si une connaissance d'arrière-plan est disponible, on pourrait conserver la table des prototypes sous forme de généralisations. En effet, on pourrait remplacer par un terme plus général la liste énumérée. La table, comme un ensemble, serait alors représentée en compréhension plutôt qu'en extension. Par exemple, une hiérarchie de types aiderait à choisir un terme plus général. Ainsi la liste suivante "plastique, porcelaine, métal, céramique" pourrait par exemple avoir comme représentation le terme "non-poreux" (figure 16).

La table des prototypes jouera deux rôles:

- chercher les manques de la théorie causés par une extension du langage. C'est le même rôle que l'ensemble réduit de IOU;

- garder les valeurs typiques des instances jugées normales par la théorie du moment. En effet, lors de la réparation d'une règle pour un marginal qui faillit une règle on essaiera de discriminer contre cette table (chapitre suivant).

La table des prototypes est construite lors de la déduction, mais son usage est réservé à l'induction. Le chapitre 5 décrit plus en détail la table des prototypes et son utilisation.

Réussite de la preuve pour un exemple de classe IV (marginal négatif):

La réussite d'un exemple négatif suggère de restreindre la théorie du domaine. Pour cela, les prédicats inclus dans NP figurent comme ceux qui peuvent introduire des contraintes à respecter concernant cette théorie.

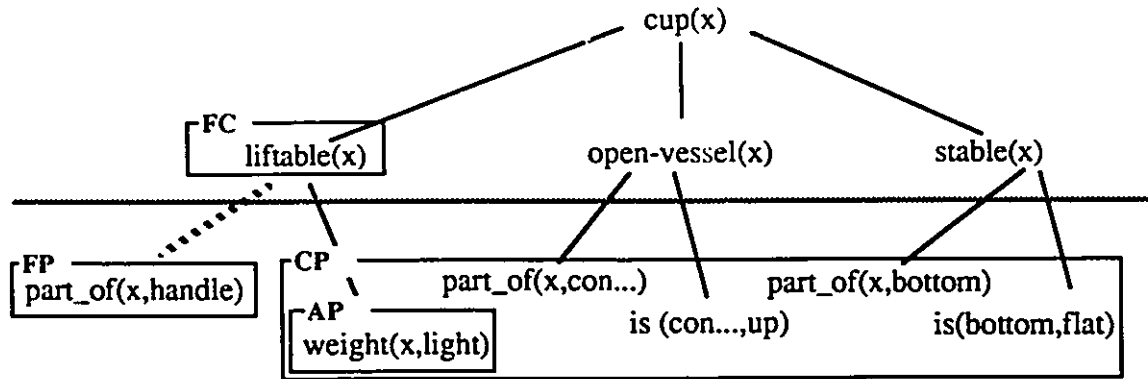
De nouveau, parmi ces prédicats, certains doivent être choisis pour justifier que cet exemple se doit d'être exclu. La table des prototypes de la classe I, aidera à trouver ces prédicats. En effet, si un prédicat se retrouve chez le marginal et dans la table des prototypes avec les mêmes arguments, alors ce prédicat ne peut être introduit pour exclure le marginal, car il excluerait le prototype. Le marginal négatif doit être exclu sans exclure de prototypes.

4.3.3 Un exemple simple: deux tasses marginales

Pour mieux expliquer la théorie, revenons à l'exemple de la tasse (section 3.7.2). Nous allons analyser les deux types de marginaux. Cette analyse est incomplète car elle ne traite que de la partie déductive du système LAP, i.e. de l'analyse de la preuve servant à classer l'information et la rendre plus facilement utilisable. On complétera cette analyse au chapitre 5.

Échec de la preuve pour un marginal positif (classeII):

La tasse en polystyrène est un exemple qui donnerait l'arbre de preuve représenté à la figure 26. Les ensembles définis précédemment y sont représentés.



FC: {lifiable(x)}
 FP: {part_of(x,handle)}
 AP: {weight(x,light)}
 NP: {made(x,styrofoam) & width(x,8) & height(x,9)}

règle possible sans élagage:
 lifiable(x) <-- weight(x,light) & made(x,styrofoam) & width(x,8)
 & height(x,9)

Figure 26. Arbre de preuve pour le marginal "tasse en polystyrène"

La théorie du domaine ne peut pas prouver que la tasse est levable. Malgré qu'elle soit légère, elle ne possède pas d'anse. La règle non élaguée de la figure 26 contient des prédicats jugés hors propos. La règle formée utilise comme antécédents les prédicats dans AP et remplace les prédicats non trouvés de FP par ceux contenus dans NP. Le conséquent pour le moment reste le prédicat de FC. Déjà, l'utilisation de l'arbre de preuve nous guide vers une règle qui peut contenir une gamme de prédicats. Malheureusement, parmi ces prédicats, plusieurs peuvent être sans rapport avec le marginal. Une connaissance d'arrière-plan du type de la figure 27 pourrait discriminer parmi ces prédicats. On constate que le responsable de l'échec a le même but ou est du même type qu'un membre de NP. En effet, le rôle d'une anse, en plus d'aider à soulever un objet est d'isoler la main du corps possiblement chaud de la tasse. À ce moment, le but d'un élément de FP pour le conséquent "lifiable" est le même que celui d'un élément de NP. Cette similarité de but justifiera le choix de cet élément parmi tous ceux présents dans NP.

isolant_thermique(x)	<--	made(x,styrofoam)
isolant_thermique(x)	<--	part_of(x,handle)

Figure 27. BK pouvant justifier une nouvelle règle

Ce prédicat de NP permettra la même conclusion que le prédicat de FP ayant échoué. A ce moment, la connaissance d'arrière-plan élaguera la règle (figure 28).

liftable(x)	<--	light(x) & isolant_thermique(x)
ou		
liftable(x)	<--	light(x) & made(x,styrofoam)

Figure 28. Nouvelle(s) règle(s) justifiées par une BK

Nous proposerons dans le chapitre suivant une méthode plus faible qui pourra élaguer la règle de la figure 26, sans justification sémantique d'une connaissance d'arrière-plan. La règle générée pourra être la même, mais sa confiance sera moindre.

Réussite de la preuve pour un marginal négatif (classe IV):

Reprenons l'exemple du seau. Ici, l'information recueillie est moins riche car la preuve est réussie et les ensembles AP, FP et FC sont vides. Il ne nous reste que NP (figure 29). La réparation risque donc d'être plus difficile.

made(x,iron) , width(x,5) , height(x,7), placed(handle,top)	NP
--	-----------

Figure 29. Ensemble NP pour le seau métallique

On se doute que la contrainte à imposer à la théorie doit être choisie parmi les prédicats de NP. Pour le moment, on ne peut pas élaguer. En effet, nous n'avons aucune idée où placer cette contrainte, car aucune règle de la théorie du domaine ne fut privilégiée dans la preuve. Voici la contrainte non élaguée:

restriction(cup(x))	<--	made(x,iron) & width(x,5) & height(x,7) & placed(handle,top)
---------------------	-----	---

Figure 30. Règle restreignante non élaguée

Une BK nous informant sur l'utilité d'avoir la concavité du dessus non obstruée pour boire permettrait d'élaguer la règle.

Au chapitre suivant, nous verrons qu'une méthode faible peut aider à choisir parmi les antécédents. Dans certains cas, on pourra même suggérer dans quelle règle placer la contrainte. Cependant, même une méthode faible permettant de raffiner la règle de la figure 30 est un atout.

4.4 Technicalités concernant le module de déduction

Le module de déduction devra trouver tous les endroits de la théorie qui sont responsables de l'échec. Pour chaque règle jugée responsable de l'échec et membre de FC, l'ensemble AP contiendra les prédicats prouvés pour chaque règle. On indexera alors ces prédicats en fonction de leur conséquent. De même, tous les prédicats jugés coupables de l'échec d'une règle seront indexés en fonction de leur conséquent. Si FC contient trois membres, alors AP et FP contiendront trois séries de prédicats indexés selon le membre correspondant de FC.

Le module de déduction doit prouver des faits et les règles et ce, en prouvant des conjonctions et des disjonctions. Toute la preuve repose sur les faits: ce sont les éléments de base pour construire la preuve. Nous emploierons un moteur d'inférence pour utiliser la théorie du domaine. La procédure de base de ce moteur sera appelée "*deduit*".

En entrée, la procédure doit avoir le prédicat à prouver et son père (conséquent) dans l'arbre de preuve. En sortie, elle retourne s'il y a échec ou non ainsi que les ensembles FC, CP, AP et FP.

La procédure *deduit* peut rencontrer trois cas spéciaux: un fait, une règle ou une conjonction. La procédure fera une analyse différente dans chaque cas.

4.4.1 Les faits

Si un fait échoue, le père de ce fait est aussitôt placé dans l'ensemble FC, et le fait est placé dans l'ensemble FP indexé par son père. On indique qu'il y a échec. Les autres ensembles sont vides.

Si le fait réussit, il est aussitôt placé dans l'ensemble AP indexé par son père. On indique la réussite. Le fait est aussi placé dans l'ensemble CP et les autres ensembles sont vides.

4.4.2 Les conjonctions

Une conjonction peut être formée de règles et/ou de faits. En se servant de la notion d'associativité à gauche, une conjonction est formée d'un élément de base et d'une autre conjonction.

L'analyse d'une conjonction possédant n membres sera formée par l'union de l'analyse du premier membre de la conjonction avec le résultat de la conjonction des analyses des $n-1$

autres membres. Ces deux composantes de la preuve auront le même père que celui de la conjonction.

La conjonction est prouvée si le premier membre est prouvé et si la conjonction restante est aussi prouvée. Les ensembles de la conjonction seront la réunion des ensembles des deux composantes de la preuve. Une conjonction prouvée donnera un ensemble FC et FP vides, et tous les membres de la conjonction seront dans le AP indexé par le père de la conjonction et dans CP.

Une conjonction faillissant un membre qui est un fait aura son père dans FC en plus des autres conséquents responsables d'échec; le fait sera dans FP, indexé par ce père. Le reste pourra être dans AP s'il a réussi.

Une conjonction faillissant seulement un membre qui est une règle aura le conséquent de cette règle dans FC. AP et FC contiendront déjà des éléments dépendant de l'analyse de la règle. Son père ne sera pas dans FC car elle n'a pas failli de faits.

4.4.3 Une règle

L'analyse du conséquent de la règle découlera de l'analyse des antécédents de la règle. Le conséquent reçoit l'analyse de ses antécédents et construit la sienne. Les ensembles CP et FC restent inchangés. Les ensembles FP et AP seront ajustés si la règle a échoué à cause d'une autre règle. A ce moment, on éliminera de ces ensembles la partie indexée par le conséquent. En effet, on ne la rend pas coupable.

4.4.4 Une disjonction de règles

On pourra avoir soit une seule règle ou soit des disjonctions de règles. Chaque règle d'une disjonction sera analysée de façon indépendante (en parallèle). L'ensemble des solutions résultant de la disjonction sera analysé ultérieurement.

Pour plusieurs règles lancées en parallèle qui ont toutes échoué, on se doit de choisir la solution la plus facile à réparer.

D'abord il faut prendre la règle ayant le moins de membres dans son FC. Une règle ayant failli à un seul endroit dans son sous-arbre sera plus facile à réparer qu'une règle ayant failli à deux endroits. En cas d'égalité de deux règles ayant échoué le même nombre de fois, plusieurs alternatives sont possibles: on peut garder la règle qui échoue le plus bas dans l'arbre, on peut garder la première règle sachant qu'elles furent toutes placées dans la

Thèse - Boris Plante

théorie en ordre d'importance, ou encore on peut juger de l'importance d'une règle en fonction du nombre de prototypes qu'elle peut expliquer.

Une méthode faible demande de choisir l'arbre le plus facile à réparer.

Une méthode forte pourrait classer en fonction de la difficulté de réparation tous les arbres engendrés par la disjonction. Ayant une BK, elle pourrait tenter de réparer chacun de ces arbres en les essayant à tour de rôle, puis elle choisirait le plus simple à réparer.

5 PARTIE INDUCTIVE ET LA TABLE DES PROTOTYPES

L'idée sous-jacente au fait de traiter les instances anormales du concept comme des marginaux nous amène à essayer de construire des règles utilisant très peu d'exemples d'entraînement. Nous serons limités du côté des exemples et il va de soi que l'induction, qui repose sur un parti-pris empirique, nécessite plusieurs exemples. Comment construire une règle inductive avec peu d'exemples? Si l'analyse de l'ensemble FP (chapitre 6) ne nous aide pas à réparer la règle en employant des prédicats déjà connus de la théorie, nous sommes intéressés à des prédicats n'appartenant pas au langage de la théorie du domaine. De plus, ces prédicats doivent indiquer une différence avec la norme, i.e. avec les prototypes. La solution est de construire une nouvelle structure basée sur les propriétés des prototypes, et de discriminer nos marginaux en utilisant l'information dense concernant les prototypes. On compense le manque d'information concernant les marginaux en se servant de l'information contenue chez les nombreux prototypes.

LAP utilise deux facettes lors de son pas inductif:

1. D'abord, on procède à une analyse des prédicats de la théorie qui ont failli (ensemble FP). Ici LAP essaiera de vérifier si ces prédicats ont failli parce que le marginal avait seulement des arguments différents de ceux des exemples de classe I. LAP donne priorité à des prédicats communs à tous les exemples et déjà utilisés par la théorie. Des arguments différents serviront à différencier prototypes et marginaux (chapitre 6);
2. Si on n'a pas réussi à réparer la théorie par l'analyse précédente, alors LAP essaiera d'utiliser des prédicats non-utilisés chez le marginal et inconnus de la théorie. Parce que la théorie ne supportait pas ces prédicats, LAP utilise cette analyse en dernier ressort. Ces prédicats doivent refléter la marginalité. La marginalité ne se trouvant plus au niveau des arguments, elle se déplace au niveau de prédicats inconnus de la théorie (ci-dessous).

5.1 Prédicats membres de la table des prototypes

La preuve construite pour une instance peut laisser inemployés plusieurs prédicats contenus dans l'instance. D'ailleurs, une des grandes forces de EBL (déduction) repose sur sa possibilité de laisser naturellement de côté les prédicats jugés inaptes (irrelevant) par la théorie du domaine. Tous ces prédicats s'accumulent lors de la construction de la preuve dans l'ensemble NP défini au chapitre précédent. Si l'instance

est décrite avec une possibilité d'extension du langage, alors en plus des prédicats jugés hors propos, nous aurons tous ces nouveaux prédicats générés par l'extension du langage. NP sera un mélange de prédicats dont quelque-uns seulement seront possiblement utiles.

Pour les instances de classe I, nous utilisons ces prédicats pour construire la *table des prototypes* PTN (Prototype Table for Non-used predicates). Cette table contient tous les prédicats qui se retrouvent chez tous les prototypes et qui n'ont pas participé à la preuve (prédicats membres de l'ensemble NP).

Pourquoi vouloir absolument que les prédicats de PTN soient tous membres de chaque prototype? Cela nous permettra de différencier entre prédicats jugés inaptes et prédicats faisant partie d'une extension du langage. En effet, si nous étendons le langage, il est à souhaiter que chaque exemple contiendra les éléments nouveaux de ce langage. Les prédicats hors propos quant à eux pourront être répartis inégalement dans les exemples; cette intersection nous permettra de les éliminer.

Tous les prédicats contenus dans la table PTN retiennent leurs arguments. La table est indexée sur le terme représentant le prédicat et pour chaque prédicat en question, on a la liste des arguments ayant appartenu à tous les prototypes instantiés lors de l'analyse de cette extension du langage. La table PTN est donc partitionnée en sous-ensembles et l'union de ces sous-ensembles redonne la totalité des prédicats (avec leurs arguments) communs à tous les prototypes.

Donnons une définition plus formelle de la table PTN:

où S représente l'union de tous les prédicats inemployés par la preuve pour tous les exemples de classe I.

$$S(I) = \{ p(X) \mid p(X) \in I \cap NP \}$$

et

$$S = \bigcup_{I \in \text{classe } I} S(I)$$

On définit la relation d'équivalence suivante sur notre ensemble S:

$$p \rho q \Leftrightarrow \text{FUNC}(p) = \text{FUNC}(q)$$

Cette relation d'équivalence partitionne notre ensemble S en plusieurs classes d'équivalence:

$$[p] = \{ q \mid q \in S \text{ \& } p \rho q \}$$

Une classe $[p]$ contient tous les prédicats ayant le même foncteur (*functor*), peu importe les arguments associés à ces prédicats.

L'ensemble F est défini comme l'ensemble de tous les foncteurs communs à tous les exemples de classe 1:

$$F = \bigcap \{ \text{FUNC}(p(X)) \mid p(X) \in S(I) \}$$

La table des prototypes PTN représente les classes d'équivalence se retrouvant dans chaque prototype. Seules les classes d'équivalence communes à tous les prototypes se retrouvent dans la table PTN. Les éléments de ces classes sont les prédicats avec leurs arguments:

$$\text{PTN} = \left\{ p(X) \mid p(X) \in S \text{ \& } \text{FUNC}(p(X)) \in F \right\}$$

La table des prototypes débute avec tous les éléments de l'ensemble NP du premier exemple de classe I. Elle évolue de deux façons bien différentes, au fur et à mesure que les autres exemples de classe I sont traités.

En terme de classes d'équivalence, elle peut retrécir ou rester stationnaire: les classes doivent être communes à tous les prototypes (voir la définition de l'ensemble F).

En terme des membres dans les classes présentes, elle ne peut qu'augmenter ou rester stationnaire: pour un même foncteur, il est possible que des exemples de classe I aient des arguments différents.

PTN et S se distinguent ainsi: S contient tous les prédicats non employés par les éléments de classe I, et PTN est un sous-ensemble de S gardant les éléments de S ayant un foncteur commun chez tous les prototypes.

La table des prototypes est amenée à jouer deux rôles différents.

Vis à vis les prototypes, PTN joue le rôle de l'*ensemble S* de "Version Spaces" [Mitchell 1982]. Elle représente un ensemble qui devient de plus en plus général avec l'acquisition d'information découlant de tout nouvel exemple de classe I. La figure 31 montre comment la table s'agrandit en présence d'un nouveau prototype ayant les mêmes classes d'équivalence¹. Le résultat est l'union de tous les prédicats de la table et des prédicats de l'exemple, car ils ont en commun les mêmes classes d'équivalence. Ici, pour la classe d'équivalence [p1], la table ajoute l'élément $p_1(Z)$ provenant de l'exemple.

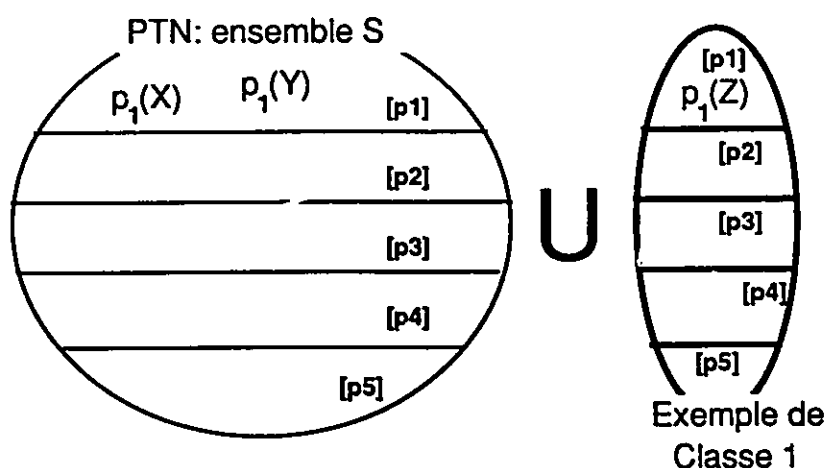


Figure 31. Ensemble PTN et l'ensemble S de "Version Space"

L'ensemble PTN peut être résumé sous forme de classes définies par la tête du prédicat, où les arguments sont les membres des classes. Un nouveau prototype permet de généraliser cet ensemble en ajoutant de nouveaux arguments à chaque classe si nécessaire.

L'utilisation d'une hiérarchie de types nous permettrait de regrouper plusieurs éléments d'un ensemble sous un seul type ou une seule appellation. Il est évident que la table pourrait être utilisée dans le but de retracer des informations concernant le concept même,

¹Dans le cas contraire, le nombre de classes d'équivalence peut diminuer advenant une classe absente chez le nouvel exemple. Le nombre total d'exemples pour les autres classes d'équivalence continue cependant à grandir et le nombre de prédicats des autres classes peut même augmenter.

et que des règles définissant mieux le concept pourraient être créées. Mais ce n'est pas le but de cette recherche.

Vis à vis les marginaux, la table PTN joue un rôle similaire à l'ensemble G de "Version Spaces". Les marginaux sont des instances qui contiennent de l'information qui diffère en un point de celle contenue dans PTN. Dans "Version Space", l'ensemble G doit être la généralisation la plus générale possible. Donc, aucun négatif ne doit être apparié par G et ces mêmes négatifs servent à spécialiser G . Les négatifs doivent contenir en eux quelque chose les excluant de G . L'ensemble G sert à restreindre le concept en éliminant les négatifs. LAP cherche pour chaque marginal une caractéristique pouvant le démarquer des prototypes. Pour un marginal de classe II, on utilise cette caractéristique pour le réintégrer dans la théorie. Pour un marginal de classe IV, on emploie cette caractéristique pour l'exclure de la théorie soit en la niant ou en affirmant celles des prototypes au niveau de la théorie. Pour nos marginaux, LAP cherche une caractéristique pouvant leur permettre de se démarquer de PTN, comme les négatifs dans "Version Space" se démarquent de G . La figure 32 montre comment la table des prototypes est utilisée pour trouver ce qui exclut un marginal.

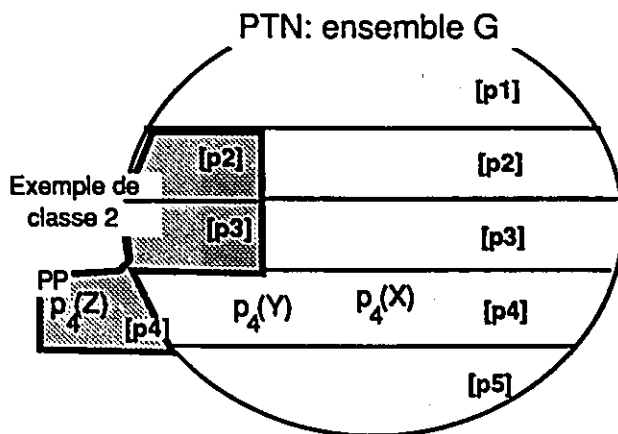


Figure 32. Ensemble PTN et l'ensemble G de "Version Space"

A la figure 32, la classe d'équivalence déterminée par la tête (functor) du prédicat p_4 est la classe qui attire notre attention, car les prototypes ne possèdent pas les mêmes arguments que le marginal. En effet, les prototypes peuvent avoir comme arguments X ou Y , et le marginal a Z comme argument.

5.2 Marginaux versus la table des prototypes

La table peut être utilisée pour trouver de nouvelles lois pour les prototypes, en comblant des manques de la théorie. Nous proposons une nouvelle façon d'utiliser cette information. Le système utilise la table pour remplacer le **manque d'information** concernant les marginaux. En effet, pour les marginaux, on est intéressé par le complément à la table des prototypes: tout prédicat appartenant par ses arguments à ce complément est jugé prometteur. Cette approche est nouvelle: on se sert de l'information des prototypes pour confirmer l'information restreinte du marginal.

Expliquons-nous. Imaginons une table PTN contenant pour le prédicat *made*, la liste suivante de matériaux: fer, porcelaine et céramique. Supposons qu'un marginal contienne dans sa description le prédicat *made*. Pour le moment, rien n'indique une marginalité. Imaginons que l'argument de *made* pour ce prédicat soit *styrofoam*. En consultant la table PTN, on constate que par son argument, ce prédicat est marginal. Ce prédicat est donc jugé intéressant en vue d'une règle future. En effet, par cet argument, qui appartient au *complément de PTN*, on suspecte une raison à la marginalité. Cette hypothèse se base sur une information riche se retrouvant chez les instances normales.

On peut utiliser la table des prototypes de deux façons pour identifier qu'un prédicat peut appartenir au complément de PTN:

1. un prédicat $p(A)$ du marginal I se retrouve dans l'ensemble NP, p se retrouve dans PTN, mais les $APGS(p(A))$ ne sont pas dans PTN. Alors, on notera que ce prédicat, par ses arguments, est marginal aux prototypes. De plus comme le prédicat fait partie de l'ensemble PTN, très sélectif de par l'intersection dans sa définition, il y a peu de chances qu'on soit en présence d'un prédicat hors propos;
2. l'instance I a un prédicat $p(A)$ dans NP et p n'a aucune entrée dans PTN. Alors, comme ce prédicat n'est pas employé par les prototypes, il est possible qu'il soit digne de marginalité. Cependant, ce prédicat pourrait être un prédicat hors propos. En effet, on se sert seulement du marginal ici pour formuler l'hypothèse que ce prédicat est possiblement discriminatoire vis-à-vis des prototypes.

Donc, nous introduisons deux nouvelles définitions pour démarquer entre eux les prédicats de NP afin de raffiner et d'élaguer les règles:

PP représente l'ensemble des prédicats prometteurs (*promising instances of predicates*), i.e. l'ensemble de tous les prédicats dont les noms sont dans la table des prototypes et dont les arguments ne sont pas dans la table:

$$PP = \{ p_i(A) \in I \mid p_i(A) \in NP \text{ and } p_i(B) \in PTN \text{ and } \text{ARGS}(p_i(A)) \neq \text{ARGS}(p_i(B)) \}$$

MP représente l'ensemble des prédicats marginaux (*marginal predicates*), i.e. l'ensemble contenant tous les prédicats de NP qui ne sont pas présents dans la table PTN:

$$MP = \{ p_i \in I \mid p_i \in NP \text{ and } p_i \notin PTN \}.$$

L'algorithme donne priorité aux prédicats contenus dans PP car ceux-ci sont moins susceptibles d'être des prédicats hors propos puisqu'ils se retrouvaient chez tous les prototypes.

Une autre raison pour donner priorité aux prédicats contenus dans PP repose sur la nécessité de ne pas introduire d'*incohérence* avec une nouvelle règle. En effet, une règle construite avec un prédicat de PP ne peut pas accepter de prototype et ainsi donner une deuxième explication pour ce prototype. Comme les arguments des prédicats dans PP sont différents des arguments des prédicats inclus dans PTN, la nouvelle règle ne permet pas à un prototype d'être accepté par la théorie avec plusieurs preuves différentes, si la réparation fait partie d'une règle étendant le concept. Il va de soi que toute règle étendante introduite ne doit pas inclure les négatifs ayant failli au même endroit. Or ces négatifs sont déjà indexés sur cet échec (section 4.3.1) et prêts à être utilisés pour justifier la règle.

Dans le cas d'une règle restreignant le concept, un prédicat pris dans PP ne peut pas éliminer des prototypes et aucune vérification n'est nécessaire à ce moment.

L'ensemble MP, quoique dangereux, peut jouer un rôle de "chien de garde". En effet, le fait de définir PTN comme une intersection peut être jugé trop restrictif, car on ne met dans PTN que ce qui est commun à tous les exemples de classe I. Cette définition très restrictive peut causer l'élimination de prédicats pouvant être employés de temps en temps pour des objets particuliers. MP pourra retrouver ces prédicats particuliers attachés à certains exemples.

5.2.1 Marginaux positifs (classe II) et ensemble PP

Reprenons l'exemple de la tasse en polystyrène de la section 3.7.2. Le module déductif donne l'information contenue à la figure 26 de la section 4.3.3. Maintenant, l'ensemble NP est analysé pour produire l'ensemble PP. Dans ce cas-ci, comme les prédicats *made*, *width* et *height* se retrouvent chez tous les prototypes dus à une extension du langage, ils sont tous dignes de faire partie de PP (figure 34). Cependant, la largeur et la hauteur de la tasse recoupent celles des prototypes. Donc, ces prédicats ne sont pas assez marginaux pour permettre de bien caractériser la tasse marginale. La partie gauche de la figure 34 résume ce cas. Nous remarquons que notre méthode converge rapidement vers le bon prédicat. Nous pouvons donc élaguer notre règle.

liftable(x)	<--	weight(x,light) & made(x,styrofoam)
-------------	-----	-------------------------------------

Figure 33. Règle produite avec une méthode faible pour le premier palier

La règle suggérée (figure 33) est la même que celle générée à la figure 28.

Cas 1: prédicat "made" est dans PP	Cas 2: prédicat "made" est dans MP
• table des prototypes PTN: made: iron, plastic, ceramic, porcelain width: min 5, max 9 height: min 4 max 8 • PCP: FC: liftable(x) AP: weight(x,light) FP: part_of(x,handle) NP: width: min 8 max 8 height: min 7 max 7 made: styrofoam • PP: made: styrofoam • MP: []	• table des prototypes PTN: width: min 5, max 9 height: min 4 max 8 • PCP: FC: liftable(x) AP: weight(x,light) FP: part_of(x,handle) NP: width: min 8 max 8 height: min 7 max 7 made: styrofoam • PP: [] • MP: made: styrofoam

Figure 34. Prédicats catégorisés selon la preuve pour la tasse "styrofoam"

Donc, pour un marginal positif, on doit utiliser les prédicats compris dans PP pour élaguer la règle, et il faut vérifier si cette règle introduit des négatifs de classe III. Si on introduit

des négatifs, il faut rendre la règle plus spécifique en employant d'autres prédicats compris dans NP qui démarquent ce marginal des négatifs indésirables.

5.2.2 Marginaux positifs et ensemble MP

Supposons maintenant que le langage soit étendu pour les dimensions physiques seulement. Alors, le prédicat *made* n'est pas commun à tous les prototypes, et le côté droit de la figure 34 indique que l'ensemble PP est vide. De nouveau, les prédicats *width* et *height* sont dans PTN et leurs arguments recoupent ceux de PTN. Ces prédicats semblent donc typiques.

Le prédicat *made* est peut-être un bon prédicat pour discerner ce marginal des prototypes. Cependant, il pourrait aussi bien être un prédicat hors propos.

Donc, un prédicat inclus dans MP si PP est vide peut nous aider à construire la même règle. Cependant, la confiance sera moindre pour cette suggestion.

5.2.3 Marginaux négatifs (classe IV) et ensemble PP

Reprenons l'exemple du petit seau de la section 3.7.2. L'ensemble NP contient les prédicats *made*, *width*, *height* et *placed*. Imaginons que tous ces prédicats peuvent faire partie d'une extension du langage et qu'ils se retrouvent chez les prototypes. De plus, si *placed* est le seul prédicat digne d'appartenir à PP, alors la marginalité est expliquée. Deux choix s'offrent à nous. Premièrement, on peut nier ce prédicat avec les arguments du marginal: *not placed(handle,top)*. Deuxièmement, on peut confirmer ce prédicat avec les arguments de la table des prototypes si l'ensemble représentant les arguments est composé d'un seul élément (*placed(handle,side)*), ou, en présence d'une BK, si les arguments étaient résumés sous une appellation unique.

Cas 1: le prédicat "placed" est dans PP	Cas 2: le prédicat "placed" est dans MP
• table des prototypes PTN: made: iron, plastic, ceramic, porcelain width: min 5, max 9 height: min 4 max 8 placed: handle*[side] • PCP: FC: []; AP: []; FP: [] NP: width: min 5 max 5 height: min 7 max 7 made: iron placed(handle,top) • PP: placed(handle,top) • MP: []	• table des prototypes PTN: width: min 5, max 9 height: min 4 max 8 • PCP: FC: []; AP: []; FP: [] NP: width: min 5 max 5 height: min 7 max 7 made:iron placed(handle,top) • PP:[] • MP: placed(handle,top)

Figure 35. Prédicats catégorisés par notre méthode faible pour l'exemple du seau

Donc pour un marginal négatif, on doit nier les prédicats compris dans PP ou d'affirmer les prédicats correspondants faisant partie de PTN s'ils ont des arguments uniques pour tous les prototypes. La restriction à imposer pour ce genre de marginal peut se fier sur le prédicat *not placed(handle,top)* ou *placed(handle,side)* selon qu'on veut restreindre la définition du concept en niant les marginaux négatifs ou qu'on veut restreindre le concept en donnant une définition plus étroite des prototypes.

5.2.4 Marginaux négatifs et ensemble MP

En regardant le côté droit de la figure 35, nous voyons une table des prototypes concernant les dimensions physiques seulement. L'ensemble MP permet de retrouver le prédicat *placed(handle,top)* et, comme PP est vide, on peut suggérer une restriction employant le prédicat *not placed(handle,top)*.

5.3 Marginaux négatifs et localisation de la restriction

Où placer la restriction? Pouvons-nous la rattacher à d'autres prédicats ou règles?

Les *erreurs de commission* [Pazzani 1988] faites par une théorie sont les erreurs les plus difficiles à corriger. En effet, tout exemple négatif ayant réussi une théorie ne fournit aucune information sur l'endroit de la théorie à réparer contrairement aux exemples positifs qui échouent (erreurs d'omission).

Retracer des règles servant de candidats possibles à la localisation de la restriction est souhaitable. Nous introduisons une autre catégorie de prédicats qui représenteront les conséquents capables de se lier avec les prédicats de PP ou MP:

PC représente les conséquents prometteurs (*promising consequents*), i.e. l'ensemble des prédicats pouvant recevoir les règles de MP ou PP afin de restreindre le concept.

$$PC = \{ p_i \mid p_j \in \text{ANTC}(I, p_i) \text{ and } p_k \in MP \cup PP \text{ and } \text{ARGS}(p_j) \cap \text{ARGS}(p_k) \neq \emptyset \}$$

En revenant à l'exemple précédent (figure 35), on retrouve dans PC la règle *liftable(x)*. En effet, "handle" est un argument de *part_of(x, handle)* qui fait partie des antécédents de *liftable(x)* et "handle" est un argument de *placed(handle, top)*.

On peut suggérer de spécialiser la règle *liftable(x)*. Si on se fie aux arguments dans PTN, la restriction est plus facilement digne du premier palier de représentation (figure 36). On restreint la portée de *liftable(x)*. Si la théorie du domaine sert à d'autres objets que les tasses, on restreint trop fortement la portée de *liftable(x)*; l'utilisateur doit alors vérifier la sémantique de la restriction.

$\text{liftable}(x) \quad \leftarrow \quad \text{weight}(x, \text{light}) \ \& \ \text{part_of}(x, \text{handle}) \ \& \ \text{placed}(\text{handle}, \text{side})$
--

Figure 36. Règle produite pour le premier palier concernant le seau

Une restriction au second palier peut certes associer cette restriction aux tasses, et en particulier aux tasses qui ont des anses.

$\text{restriction}(\text{cup}): \quad \text{liftable}(x) \quad \leftarrow \quad \text{weight}(x, \text{light}) \ \& \ \text{part_of}(x, \text{handle}) \ \& \ \text{placed}(\text{handle}, \text{side})$
--

Figure 37. Règle produite pour le second palier concernant le seau

Advenant un ensemble PC vide, il est évident qu'une représentation à deux paliers est plus souple et qu'il est plus facile de placer une restriction générale sur les tasses. Un ensemble PC vide indique qu'il est impossible de localiser un endroit de la théorie pouvant accueillir la restriction trouvée avec cette méthode. Pour les marginaux de classe IV, une représentation à deux paliers s'avère plus prometteuse et permet de restreindre la théorie de façon globale sans toucher aux règles existantes. LAP utilise une représentation à deux paliers.

6 PARTIE INDUCTIVE: ANALYSE DES PRÉDICATS AYANT FAILLI

Dans ce chapitre, on utilisera LAP pour réparer la théorie nous permettant le classement d'un objet dans une classe en particulier, i.e. une théorie résumant une taxonomie. De plus, nous voulons montrer une autre facette du mécanisme d'induction employé par LAP.

LAP aura à résoudre un problème inhérent à une taxonomie dû au fait qu'une taxonomie renferme une variété de prototypes dépendant de chaque classe, i.e. les prototypes d'une classe sont différents des prototypes d'une autre classe. Par exemple dans une taxonomie du monde animal, les prototypes des mammifères sont complètement différents des prototypes des oiseaux.

6.1 Notion de prototypes dans une taxonomie

Tout animal vivant peut être classé soit comme mammifère, soit comme oiseau, soit comme reptile, etc. Les mammifères typiques sont complètement différents des oiseaux typiques. LAP doit donc se servir d'une table de prototypes axée sur les classes et non sur le règne animal en général. Illustrons le tout par un exemple.

La théorie utilisée pour cet exemple n'englobe que les mammifères, les oiseaux et les reptiles. La théorie est incomplète et ne peut accepter que les cas typiques. La théorie est résumée à la figure 38. La table 3 représente une série de cas typiques acceptés par la théorie.

Cet exemple est très différent de l'exemple de la tasse. En effet, la marginalité pour un mammifère ne repose pas sur les mêmes raisons que pour un oiseau ou un reptile. Si on construit une seule table des prototypes pour un animal en général, alors on aura une tour de Babel et il sera impossible de bien discriminer les prédicats pouvant être un signe de marginalité. Pour un oiseau, les ailes sont un apport au prototype, tandis que pour la chauve-souris, les ailes sont un atout pour sa marginalité. Si la table des prototypes est trop vaste et étendue, il sera difficile de choisir les ailes comme étant le prédicat expliquant la marginalité de la chauve-souris.

Par cet exemple, on réalise que pour une taxonomie, plusieurs tables de prototypes doivent être construites. Pour trouver la raison de la marginalité d'une baleine, par exemple, on doit comparer celle-ci avec les prototypes de la classe des mamifères et non avec les prototypes de la classe des poissons.

animal(X)	<--	vivant(X) & espèce(X)
vivant(X)	<--	respire(X, oxygène)
espèce(X)	<--	mammifère(X)
espèce(X)	<--	oiseau(X)
espèce(X)	<--	reptile(X)
mammifère(X)	<--	circulation(X,mammi) & vie_sexuelle(X,mammi) & locomotion(X,mammi)
oiseau(X)	<--	circulation(X,oiseau) & vie_sexuelle(X,oiseau) & locomotion(X,oiseau)
reptile(X)	<--	circulation(X,reptile) & vie_sexuelle(X,reptile) & locomotion(X,reptile)
circulation(X,mammi)	<--	sang(X,chaud) & coeur(X,4chambres)
circulation(X,oiseau)	<--	sang(X,chaud) & coeur(X,4chambres)
circulation(X,reptile)	<--	sang(X,froid) & coeur(X,4chambres_imparfaites))
vie_sexuelle(X,mammi)	<--	fertilisation(X,interne) & soins_petits(X,mammi)
vie_sexuelle(X,oiseau)	<--	fertilisation(X,interne) & soins_petits(X,oiseau)
vie_sexuelle(X,reptile)	<--	fertilisation(X,interne) & soins_petits(X,reptile)
locomotion(X,mammi)	<--	mem_ant(X,pattes) & mem_pos(X,pattes) & déplace(X,terre)
locomotion(X,oiseau)	<--	mem_ant(X,ailes) & mem_pos(X,pattes) & déplace(X,air) & ailes(X,pennes)
locomotion(X,reptile)	<--	mem_ant(X,absent) & mem_pos(X,absent) & déplace(X,terre)
locomotion(X,reptile)	<--	mem_ant(X,pattes) & mem_pos(X,pattes) & déplace(X,eau)
soins_petits(X,mammi)	<--	nourrit(enfants,lait) & mammelle(X,oui) & develo_embr(X,placenta) & reproduction (X,vivipare)
soins_petits(X,oiseau)	<--	develo_embr(X,coquille) & reproduction (X,ovipare)
soins_petits(X,reptile)	<--	develo_embr(X,coquille) & reproduction (X,ovipare)

Figure 38. Taxonomie réduite pour le monde animal

Si la taxonomie des mammifères de la figure 38 séparait les mammifères en sous-classes, alors il faudrait bâtir une table pour chaque sous-classe. En effet, la couleur d'un ours blanc, par exemple, peut servir à le démarquer si la théorie n'accepte que les ours bruns et noirs. Dans ce cas, les ours typiques seraient bruns ou noirs et l'ours blanc serait un marginal.

Dans notre exemple, on vérifie que trois tables de prototypes construites au niveau des mammifères, des oiseaux et des reptiles sont efficaces pour traiter nos exemples.

	ours	cheval	pigeon	canard	boa	crocodile
respire	oxygène	oxygène	oxygène	oxygène	oxygène	oxygène
sang	chaud	chaud	chaud	chaud	froid	froid
cocur	4chambres	4chambres	4chambres	4chambres	4chamb_imp	4chamb_imp
fertilisation	interne	interne	interne	interne	interne	interne
mcm_ant	pattes	pattes	ailes	ailes	absent	pattes
mcm_post	pattes	pattes	pattes	pattes	absent	pattes
déplacement	terre	terre	air	air	terre	eau
nourrit_petits	lait	lait	XXXXXX	XXXXXX	XXXXXX	XXXXXX
mammelle	oui	oui	XXXXXX	XXXXXX	XXXXXX	XXXXXX
dev_embryon	placenta	placenta	coquille	coquille	coquille	coquille
reproduction	vivipare	vivipare	ovivipare	ovivipare	ovipare	ovipare
recouvert	poil	poil	plumes	plumes	écailles	écailles
ailes	XXXXXX	XXXXXX	pennes	pennes	XXXXXX	XXXXXX
mastication	dent	dent	bec	bec	dent	dent

Table 3. Cas typiques pour les mammifères, oiseaux et reptiles

6.2 L'ensemble FP et les prototypes

Au chapitre 5, on a vu qu'un marginal de classe II est exclu de la théorie par l'échec d'une règle. Dans cette règle, certains antécédents ont failli et sont réunis dans l'ensemble FP. Si ces antécédents ne se retrouvent pas chez le marginal, on consulte la table des prototypes pour trouver un prédicat digne de marginalité. Avant d'utiliser la table des prototypes pour chercher une raison à la marginalité, on doit certes essayer de vérifier si les prédicats ayant failli dans la théorie (ensemble FP) ne se retrouvent pas chez le marginal avec des arguments différents.

On peut classer les prédicats de l'ensemble FP en deux catégories: ceux qui faillissent car ils sont totalement absents chez le marginal et ceux ayant failli car ils diffèrent par leurs arguments. La tasse en polystyrène a failli car le prédicat '*part_of(X,handle)*' était absent et aucun prédicat de ce style n'était présent chez la tasse en polystyrène. Dans ce cas, il est justicieux de chercher dans la table des prototypes une raison à la marginalité.

LAP, avant de chercher dans la table des prototypes, regarde les prédicats membres de l'ensemble FP et vérifie si les prédicats sont présents chez le marginal avec des arguments différents. De plus, la table PTN ne doit pas contenir ce même prédicat avec les mêmes arguments. Si c'était le cas, alors ce prédicat ne serait pas une raison suffisante pour expliquer la marginalité.

Les cas de classe I sont des prototypes. La table PTN ne contient que les prédicats non-employés par la preuve. Les prédicats de bas-niveaux faisant partie de la théorie doivent servir eux aussi à chercher la raison de la marginalité.

La théorie peut jouer pour ses prédicats un rôle analogue à la table des prototypes. Donc en plus de la table des prototypes, nous aurons une table de la théorie prototypique PTI (Prototype Table for Instantiated predicates). Cette table est analogue à la table des prototypes PTN sauf qu'elle contient *tous les prédicats de bas-niveaux employés par la théorie* lors de l'élaboration de la preuve.

Son rôle est primordial dans la phase inductive de LAP. Pour réparer une règle fautive, LAP regarde tout élément de l'ensemble FP et essaie d'associer à cet élément un élément similaire chez le marginal. Donc, pour tout prédicat $p(X)$ de FP, il détermine $FUNC(p(X))$ et essaie de l'apparier avec le marginal. S'il réussit, il vérifie si les $ARGS(p(X))$ du marginal sont différents de ceux contenus dans PTI et inexistants dans PTN. Advenant que les arguments associés au marginal sont différents, LAP mise sur cette différence pour proposer une nouvelle règle. Dans le cas contraire, LAP passe à la deuxième facette de son pas inductif.

6.3 Nos animaux marginaux

Les tables 4 et 5 sont un résumé des marginaux que LAP aura à traiter.

On remarque que, pour ces animaux, les raisons de la marginalité sont très différentes. LAP pourra réparer la théorie pour la plupart d'entre eux par sa seule analyse de l'ensemble FP. Il utilisera sa première passe inductive avec succès, et la théorie pourra être révisée par le changement des arguments. Mais LAP pourra aussi souligner les autres prédicats du marginal qui seraient un apport à sa marginalité.

	baleine bleue	chauve-souris	kangourou	ornithorynque
respire	oxygène	oxygène	oxygène	oxygène
sang	chaud	chaud	chaud	chaud
cœur	4chambres	4chambres	4chambres	4chambres
fertilisation	interne	interne	interne	interne
mem_ant	nageoires	ailes	pattes	pattes
mem_post	absent	pattes	pattes	pattes
déplacement	eau	air	terre	terre
nourrit_petits	lait	lait	lait	lait
mammelle	oui	oui	oui	non
dev_embryon	placenta	placenta	marsupium	coquille
reproduction	vivipare	vivipare	vivipare	ovipare
recouvert	nuc	poil	poil	poil
ailes	XXXXXX	XXXXXX	XXXXXX	XXXXX
mastication	fanons	dent	dent	bec

Table 4. Marginaux pour les mammifères

	manchot	autruche	vipère d'europe	tortue
respire	oxygène	oxygène	oxygène	oxygène
sang	chaud	chaud	froid	froid
cœur	4chambres	4chambres	4cham_imp	4cham_imp
fertilisation	interne	interne	interne	interne
mem_ant	ailes	ailes	absent	pattes
mem_post	pattes	pattes	absent	pattes
déplacement	eau	terre	terre	eau
nourrit_petits	XXXXXX	XXXXXX	XXXXXXXX	XXXXXXXX
mammelle	non	non	non	non
dev_embryon	coquille	coquille	interne	coquille
reproduction	ovipare	ovipare	ovovivipare	ovipare
recouvert	plumes	plumes	écailles	carapace
ailes	petites_plumes	penne	XXXXXX	XXXXXXXX
mastication	bec	bec	dent	bec

Table 5. Marginaux pour les oiseaux et reptiles

Illustrons cette technique en considérant le marginal de classe II, la "baleine bleue".

Supposons que LAP a bâti nos deux tables PTI et PTN. La figure 39 résume le résultat de la première passe pour les tables. LAP, avant de pouvoir formuler une nouvelle règle en utilisant la table PTN, essaie de trouver un appariement entre les prédicats membres de FP et un prédicat similaire chez le marginal, i.e. membre de NP. Il faut que le prédicat similaire ne soit pas présent dans la table des prototypes PTN, puisqu'il faut expliquer la marginalité.

PTI	PTN
sang : chaud	recouvert: poil
coeur: 4chambres	mastication: dent
fertilisation: interne	respire: oxygène
nourrit_petits: lait	
mammelle: oui	
dev_embryon: placenta	
reproduction: vivipare	
mem_ant: pattes	
mem_post: pattes	
déplace: terre	

Figure 39. Les tables PTI et PTN pour les mammifères

La première chose que LAP fait dans la passe 2 est d'analyser les raisons de l'échec. La figure 40 nous résume les catégories (PCP) construites par LAP pour notre marginal.

FC: [locomotion(X,mammi)]
AP: []
FP: [mem_ant(X,pattes), mem_pos(X,pattes), déplace(X,terre)]
NP: [mem_ant(baleine bleue,nageoires), mem_post(baleine bleue,absent), déplace(baleine bleue, eau) recouvert(baleine bleue, nue), mastication(baleine bleue, fanons) , respire(baleine bleue,oxygène)

Figure 40. PCP et ses ensembles pour la baleine bleue

Comme les trois prédicats ayant failli (ensemble FP) se retrouvent chez le marginal mais avec des arguments différents, LAP les juge très prometteurs. Ayant vérifié qu'il ne sont pas membres de PTI et PTN, leur justesse pour ce marginal.est confirmée. LAP suggère alors la règle suivante:

locomotion(X,mammi)	<---	mem_ant(X,nageoire) & mem_post(X,absent) & déplace(X,eau)
---------------------	------	---

Figure 41. Règle suggérée pour la baleine

On remarque la justesse du premier pas inductif de LAP, lorsqu'il trouve une alternative aux trois membres de l'ensemble FP pour réparer la règle fautive. Avec son premier pas inductif, LAP pourrait réparer plus d'une règle, car LAP associe les prédicats fautifs un à un. Le deuxième pas inductif essaie de trouver d'autres raisons à la marginalité, mais il est impossible d'associer un à un les prédicats fautifs et les prédicats prometteurs. Pour la baleine bleue, LAP pourrait suggérer que cette *baleine était aussi marginale* par le prédicat *recouvert(X, nue)* et *mastication (X, fanons)*. Un expert, voulant raffiner une taxonomie pour la subdiviser, pourrait trouver ces prédicats et ceux de la règle du second palier très pertinents pour essayer de faire une subdivision à sa taxonomie.

6.4 Nos résultats.

Cette section présente un résumé de l'analyse des cas typiques et des marginaux pour la taxonomie des animaux de la figure 38.

6.4.1 Les mammifères.

A) Tables

PTI	PTN
sang : chaud	recouvert: poil
coeur: 4chambres	mastication: dent
fertilisation: interne	respire: oxygène
nourrit_petits: lait	
mammelle: oui	
dev_embryon: placenta	
reproduction: vivipare	
mem_ant: pattes	
mem_post: pattes	
déplace: terre	

Figure 42. Les tables PTI et PTN pour les mammifères

B) Kangourou

```
FC: [ soins_petits(X,mammi)]  
  
AP: [ nourrit_petits(X,lait), mamelle(X,oui), reproduction(X,vivipare)]  
FP: [ dev_embryon(X,placenta)]  
  
NP: [dev_embryon(kangourou,marsupium)  
      recouvert(kangourou, poils), mastication(kangourou, dents),  
      respire(kangourou,oxygène)]
```

Figure 43. PCP et ses ensembles pour le kangourou

Règle suggérée

```
soins_petits(X,mammi) <-- nourrit_petits(X,lait) & mamelle(X,oui) & reproduction(X,vivipare) &  
                           dev_embryon(X,marsupium)
```

Figure 44. Règle suggérée pour le kangourou

C) Chauve-souris

```
FC: [ locomotion(X,mammi)]  
  
AP: [ mem_pos(X,pattes)]  
FP: [ mem_ant(X,pattes), déplace(X,terre)]  
  
NP: [mem_ant(chauve-souris,ailes), déplace(chauve-souris,air)  
      recouvert(chauve-souris, poil), mastication(chauve-souris, dents),  
      respire(chauve-souris,oxygène)]
```

Figure 45. PCP et ses ensembles pour la chauve-souris

Règle suggérée

```
locomotion(X,mammi) <-- mem_ant(X,ailes) & mem_pos(X,pattes) & déplace(X,air)
```

Figure 46. Règle suggérée pour la chauve-souris

D) Ornithorynque

FC: [soins_petits(X,mammi)]
AP: [nourrit_petits(X,lait)]
FP: [mamelle(X,oui), reproduction(X,vivipare),dev_embryon(X,placenta)]
NP: [mamelle(ornithorynque,non),reproduction(ornithorynque,ovipare), dev_embryon(ornithorynque,coquilles) recouvert(ornithorynque,poil), mastication(ornithorynque, bec), respire(ornithorynque,oxygène)]

Figure 47. PCP et ses ensembles pour l'ornithorynque

Règle suggérée

soins_petits(X,mammi)	<--	nourrit_petits(X,lait) & mamelle(X,non) & reproduction(X,ovipare) & dev_embryon(X,coquilles)
-----------------------	-----	---

Figure 48. Règle suggérée pour l'ornithorynque

Lors du second pas inductif, le système a également noté l'évidence de marginalité concernant la présence d'un bec pour cet animal.

6.4.2 Les oiseaux

A) Tables

PTI	PTN
sang : chaud	recouvert: plumes
coeur: 4chambres	mastication: bec
fertilisation: interne	respire: oxygène
dev_embryon: oeufs	
reproduction: ovipare	
mem_ant: ailes	
mem_post: pattes	
déplace: air	
ailes: pennes	

Figure 49. Les tables PTI et PTN pour les oiseaux

B) Manchot

```
FC: [ locomotion(X, oiseau)]  
  
AP: [ mem_pos(X,pattes),mem_ant(X, ailes)]  
FP: [ ,déplace(X, air), ailes(X,pennes)]  
  
NP: [déplace(manchot, eau), ailes(manchot,petites_plumes),  
recouvert(manchot, plumes), mastication(manchot, bec),  
respire(manchot,oxygène)]
```

Figure 50. PCP et ses ensembles pour le manchot

Règle suggérée

```
locomotion(X,oiseau)    <--    mem_ant(X,ailes) & mem_pos(X,pattes) &  
                           déplace(X,eau) & ailes(X, petites_plumes)
```

Figure 51. Règle suggérée pour le manchot

C)Autruche

```
FC: [ locomotion(X, oiseau)]  
  
AP: [ mem_pos(X,pattes),mem_ant(X, ailes), ailes(X,pennes)]  
FP: [ déplace(X, air)]  
  
NP: [déplace(autruche, terre), ailes(autruche, penne),  
recouvert(autruche, plumes), mastication(autruche, bec),  
respire(autruche,oxygène)]
```

Figure 52. PCP et ses ensembles pour l'autruche

Règle suggérée

```
locomotion(X,oiseau)    <--    mem_ant(X,ailes) & mem_pos(X,pattes) &  
                           déplace(X,terre) & ailes(X, penne)
```

Figure 53. Règle suggérée pour l'autruche

6.4.3 Les reptiles.

A) Tables

PTI	PTN
sang : froid	recouvert: écailles
coeur: 4chamb_imp	mastication: dent
fertilisation: interne	respire: oxygène
dev_embryon: coquilles	
reproduction: ovipare	
mem_ant: absent, pattes	
mem_post: absent, pattes	
déplace: terre,eau	

Figure 54. Les tables PTI et PTN pour les reptiles

B) Vipère d'Europe

FC: [soins_petits(X,reptiles)]
AP: []
FP: [reproduction(X,ovipare),dev_embryon(X,coquilles)]
NP: [reproduction(vipère d' Europe,ovipare), dev_embryon(vipère d' Europe, oeufs), recouvert(Vipère d' Europe, écailles), mastication(Vipère d' Europe, dents), respire(Vipère d' Europe,oxygène)]

Figure 55. PCP et ses ensembles pour la vipère d'Europe

Règle suggérée

soins_petits(X,reptiles) <-- reproduction(X,ovipare) & dev_embryon(X,interne)

Figure 56. Règle suggérée pour la vipère d'Europe

6.5 Discussion

Les deux pas inductifs furent utilisés pour réparer une règle fautive. La méthode employée laisse voir de multiples possibilités. Il serait certes possible d'utiliser cette méthode pour modifier en profondeur des taxonomies en fonction des marginaux. Les marginaux permettraient de subdiviser la taxonomie existante, i.e. la raffiner. Cela serait une solution à des problèmes communs en représentation des connaissances, notamment pour les représentations taxonomiques.

7 TRAVAUX RELIÉS

Ce chapitre traite des travaux reliés touchant les points suivants:

1. Usage des prédicats non utilisés par EBL pour faire de l'induction;
2. Choix d'un remplaçant à une partie d'explications de EBL qui a failli;
3. Façon d'intégrer EBL et SBL dans un système.

7.1 IOU

La méthode IOU (Induction Over the Unexplained) [Mooney and Ourston 1989] est une méthode mixte d'apprentissage. IOU essaie de trouver une définition pour un concept découlant d'une multitude d'exemples. Il utilise une théorie du domaine incomplète mais correcte. Cette théorie du domaine englobe des définitions trop générales pouvant servir à plusieurs concepts (tasse, verre, boîte de conserve...). Son but est de spécialiser la définition trop générale fournie par la théorie.

7.1.1 Résumé

IOU utilise, de façon séquentielle, les deux méthodes principales en apprentissage. La majorité des autres méthodes mixtes entrelacent les deux méthodes; l'une guidant l'autre. Premièrement, EBL essaie de trouver une partie explicable C_e du concept à partir de la théorie trop générale. Devant s'appliquer à plusieurs exemples, C_e réunit les résultats de EBL communs à tous les exemples positifs.

Ce résultat étant vraiment trop général, puisque C_e peut s'apparier à quelques exemples négatifs. Alors SBL (induction) est utilisé pour trouver les caractéristiques inexplicables C_n (en opposition à la déduction qui explique). SBL emploie tous les exemples positifs et les exemples négatifs pouvant s'apparier à C_e ; ce sera le nouvel ensemble d'entraînement. L'ensemble est réduit de deux façons. Premièrement, le nombre d'exemples est réduit car certains négatifs peuvent être déjà éliminés par C_e . Deuxièmement, le nombre de prédicats décrivant les exemples est réduit car tous les prédicats utilisés dans C_e sont jugés non utiles pour spécialiser le concept.

IOU utilise un algorithme inductif pour trouver la composante C_n . Cette composante contient les restrictions pour spécialiser le produit d'une théorie trop générale (theory-based concept specialization problem).

Le concept final est formé de la conjonction $C_c \wedge C_n$. La figure 57 résume le système IOU.

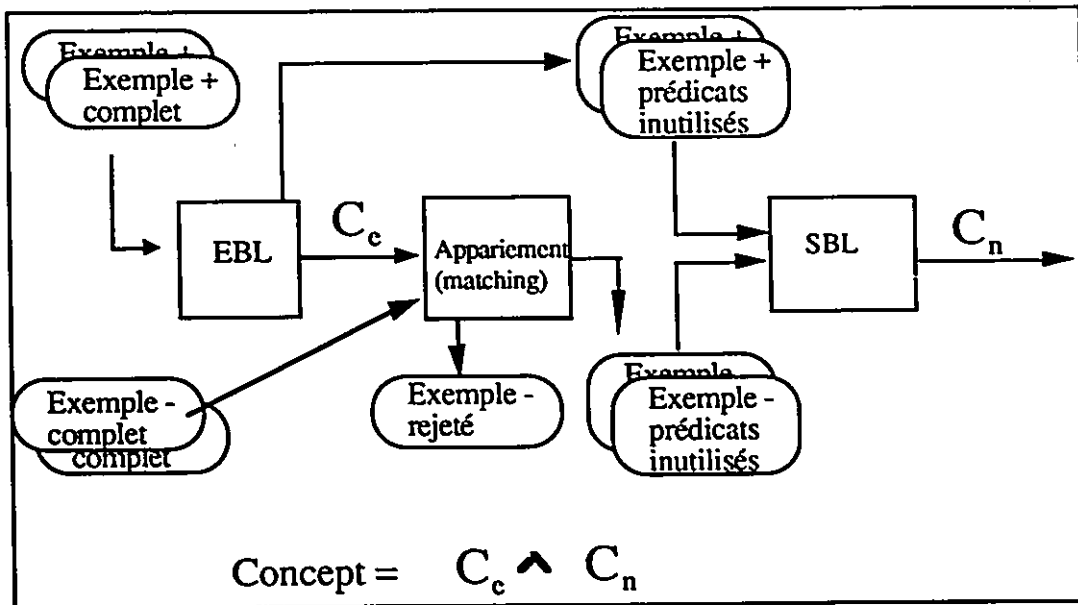


Figure 57. Le système IOU

7.1.2 Un exemple

Reprenons l'exemple de [Mooney and Ourston 1989]. La théorie est trop générale et concerne les récipients permettant de boire ('drinking-vessel') et on veut spécialiser le concept pour le cas des tasses. Les tasses doivent se démarquer des verres, des mugs et des boîtes de conserve.

La théorie de Mooney est résumée à la figure 58.

Drinking-Vessel(x)	<--	stable(x) & liftable(x) & open-vessel(x)
stable(x)	<--	has-bottom(x) & flat-bottom(x)
liftable(x)	<--	graspable(x) & light(x)
graspable(x)	<--	has-handle(x)
graspable(x)	<--	width(x,small) & insulating(x)
open-vessel(x)	<--	has-conca(x) & upward-pointing-conca(x)

Figure 58. Théorie pour 'drinking-vessel' utilisée par IOU.

La table 6 énumère les exemples d'entraînement utilisés par IOU.

nom(classe)	cup-1(+)	cup-2(+)	shot-glass(-)	mug1(-)	can-1(-)
bottom	yes	yes	yes	yes	yes
flat	yes	yes	yes	yes	yes
conca	yes	yes	yes	yes	yes
up	yes	yes	yes	yes	yes
light	yes	yes	yes	yes	yes
handle	yes	no	no	yes	no
width	small	small	small	med	small
insul	no	yes	yes	no	no
color	white	red	clear	copper	silver
volume	small	small	tiny	large	small
shape	cylinder	cylinder	cylinder	cylinder	cylinder

Table 6. Exemples d'entraînement pour le concept de TASSE (IOU)

Après l'usage de EBL, on obtient le concept C_e suivant:

has-bottom(x) & flat-bottom(x) & has-conca(x) & upward-pointing-conca(x) & light(x)
& [has-handle(x) OU { width(x,small) & insulating(x) }]

Ce concept C_e permet d'éliminer l'exemple négatif *can-1*. De plus, les exemples restant ont seulement trois attributs qui furent inemployés. La table 7 contient ces exemples restant et les attributs inemployés qui seront utilisés par SBL.

nom(classe)	cup-1(+)	cup-2(+)	shot-glass(-)	mug1(-)
color	white	red	clear	copper
volume	small	small	tiny	large
shape	cylinder	cylinder	cylinder	cylinder

Table 7. Exemples d'entraînement réduits pour concept de TASSE (IOU)

En employant un algorithme inductif (Version Space, ID3, A⁹), on restreint le concept avec la composante C_n .

Dépendamment du parti-pris de chaque algorithme, on pourra avoir pour C_n deux possibilités montrées à la figure 59.

plus général:	volume(x,small)
plus spécifique:	volume(x,small) & shape(x,cylinder)

Figure 59. Description de la composante C_n du concept tasse par IOU

7.1.3 Comparaison avec LAP

IOU cherche à restreindre une définition de concept engendrée en premier lieu par une théorie du domaine trop vaste. LAP peut aussi en un deuxième temps essayer de restreindre le concept, mais il agira en imposant des restrictions concernant la théorie du domaine particulière à ce concept.

IOU travaille avec un ensemble d'exemples et tous sont traités sur le même pied: les positifs doivent tous réussir EBL, et la théorie ne peut être incorrecte. IOU impose que tous les positifs sont prouvés par EBL. LAP, quant à lui, a comme but premier de s'occuper des marginaux. Les positifs non prouvés par la déduction sont ses exemples de base pour la partie inductive qui essaie alors de réparer la théorie pour ces exemples.

Les deux méthodes utilisent EBL suivi de SBL. Les deux méthodes passent à SBL un exemple contenant des réductions de prédicats. IOU passe tous les exemples d'un seul coup car il cherche une description commune à tous. LAP utilise son ensemble NP pour chaque marginal et tente d'induire en opposition avec les positifs typiques. La réduction faite par IOU tient compte de tous les exemples, celle de LAP ne tient compte que du marginal. La réduction d'IOU ressemble plus à la table des prototypes qui pourrait servir dans le même but qu'IOU avant ou après que les marginaux soient tous traités i.e. trouver des manques à la théorie.

La différence principale se résume en des méthodes différents. LAP traite d'abord les marginaux et IOU traite tous les cas sur le même pied pour restreindre un concept. La seconde différence en importance touche à la théorie: IOU ne veut pas réparer la théorie, elle est correcte. LAP veut l'ajuster pour traiter les marginaux et veut l'améliorer pour mieux couvrir les exemples normaux.

7.2 LISE

Genest et Matwin [Genest, et al. 1990] ont développé LISE, un système d'apprentissage mixte aidant à ajuster une théorie incomplète utilisant l'analogie de buts pour élargir la théorie du domaine.

La théorie du domaine prise comme exemple est composée d'une suite de transactions concernant un système quelconque (banque,...).

Pour un exemple d'entraînement d'une transaction, LISE essaie d'expliquer cet exemple en utilisant toutes les transactions connues du système. Si l'exemple ne peut satisfaire aucune des transactions connues, le système croit en la justesse de l'exemple et essaie d'expliquer cet exemple insatisfait en utilisant une approche en trois étapes. Notre comparaison avec LISE ne considère que les deux premières étapes.

D'abord, l'abduction est employée pour essayer de compléter une explication partielle. Advenant ce cas, l'exemple est reconnu, mais la théorie n'est pas ajustée.

Si l'abduction faillit, le raisonnement analogique est employé pour compléter l'explication de l'exemple. Le système essaie de créer une transaction inexistante et de la justifier. Si, par un raisonnement analogique, il réussit à expliquer l'exemple en créant une nouvelle transaction, alors la théorie du domaine est augmentée pour inclure cette nouvelle transaction.

Pour ce faire, LISE essaie toutes les transactions pour obtenir des explications, dites partielles, qui n'expliquent que certaines parties de l'exemple. LISE classe ces explications partielles par ordre de préférence. Il utilise ces explications à tour de rôle en essayant de remplacer les parties non prouvées, dues à des faits inexistants dans l'exemple, par des faits différents qui ont le même but (analogie). En cas de succès, LISE crée une nouvelle transaction en remplaçant les parties non prouvées par des parties inemployées de l'exemple, mais analogues. Le nom des transactions seront suggérés par l'utilisateur. La figure 60 résume LISE.

LISE possède un BK lui permettant de faire son raisonnement analogique en se fiant aux buts poursuivis par certains prédicats.

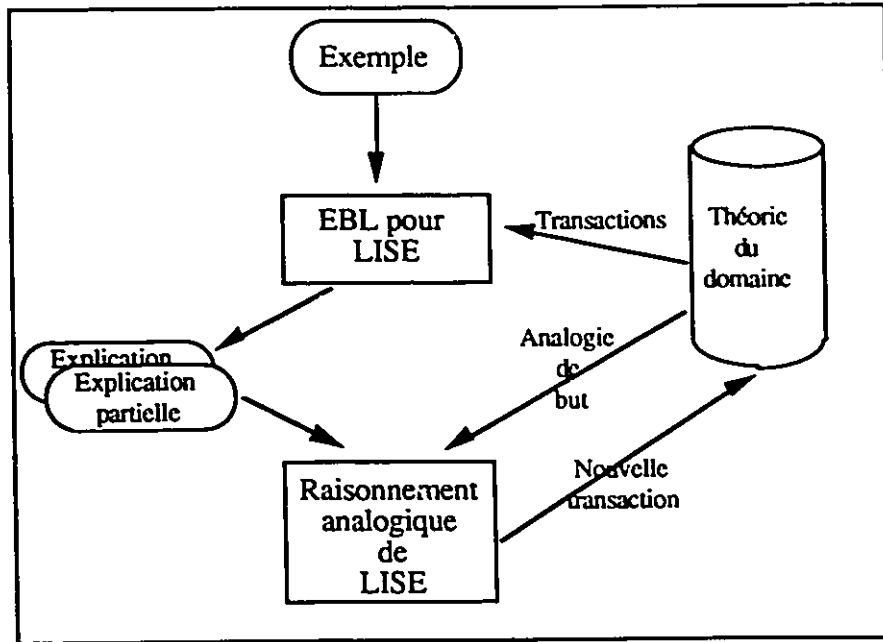


Figure 60. Le système LISE (deux premières étapes)

7.2.1 Un exemple

L'exemple suivant ne touche que le raisonnement analogique. La théorie du domaine est composée de deux transactions 'withdraw' et 'deposit' qui sont représentées sous forme de cadres (frames). Une transaction peut utiliser des actions qui sont des opérations de plus bas niveaux. Les figures 61 et 62 décrivent cette théorie du domaine.

<pre> withdraw(Person,Amount) isa: TRANSACTION precondition: account(Person,Account) goal(account,identify_client) procedure: debit(Account,Amount) issue_money(Person,Amount) goal(issue_money,dec_client_liquid) </pre>	<pre> deposit(Person,Amount) isa: TRANSACTION precondition: account(Person,Account) goal(account,identify_client) procedure: receive_money(Person,Amount) goal(receive_money,inc_client_liquid) credit(Account,Amount) </pre>
---	---

Figure 61 Transactions de LISE

La transaction 'withdraw(Person,Amount)' se résume ainsi. Une personne peut retirer un montant d'argent si elle a un compte. À ce moment, on applique l'action 'debit(Account,Amount)' qui vérifie si la balance dans le compte est supérieure au montant voulu. Si c'est le cas, le montant est accordé.

<pre>debit(Account,Amount) isa: ACTION precondition: balance(Account,Balance) goal(balance,protect_bank_interest) Balance > Amount procedure: sub_fm_balance(Account,Amount) goal(sub_fm_balance,record_transaction)</pre>	<pre>credit(Account,Amount) isa: ACTION precondition: nil procedure: add_to_balance(Account,Amount) goal(add_to_balance,record_transaction)</pre>
---	---

Figure 52. Actions de LISE

Il est à remarquer que les buts associés à certains antécédents ou conséquents font partie du cadre.

account(bob,acc_1), credit_margin(bob,3500), record_loan(bob,1000), issue_money(bob,1000), car(bob,car_of_bob), value(car_of_bob,12000).

Figure 63. Exemple pour 'borrow' de LISE

Supposons que l'exemple d'entraînement concernant la transaction 'borrow' de la figure 63 soit présenté à LISE. Comme cette transaction est inconnue, LISE produit à partir des deux transactions connues, deux explications partielles. La figure 64 résume l'explication partielle produite en utilisant 'withdraw' qui servira à créer la nouvelle transaction. Les lignes pointillées indiquent les parties non prouvées pour l'exemple.

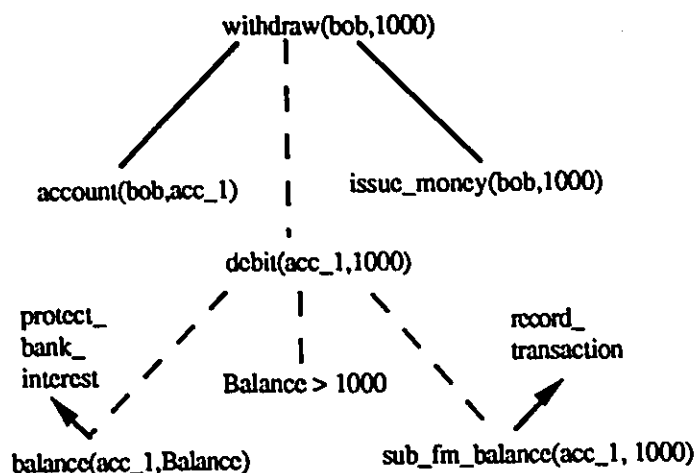


Figure 64. Explication partielle de 'borrow' utilisant 'withdraw'

En employant son raisonnement par analogie, LISE s'aperçoit que l'action 'debit' peut servir à construire une autre action, car par similitude de buts, LISE sait que

"balance(acc_1,Balance)" a le même but que "credit_margin(bob,3500)" et que "sub_fm_balance(acc_1,1000)" a le même but que "record_loan(bob,1000)".

Voici les transactions et actions faites par LISE.

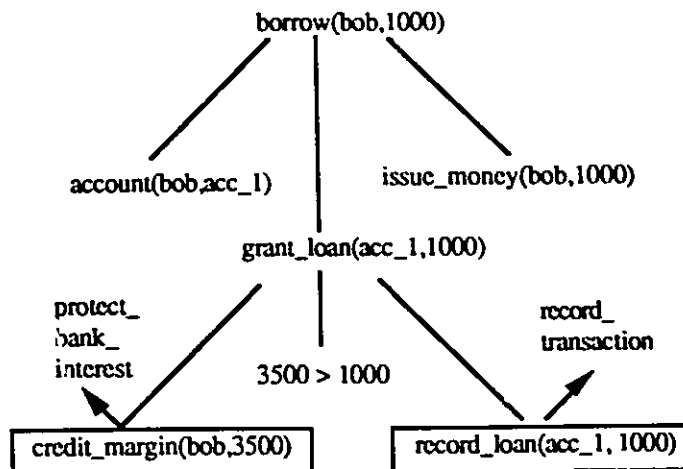


Figure 65. Nouvelles transactions créées par LISE

Ainsi, par analogie, l'action 'debit' sert à produire l'action 'grant_loan', et la transaction 'withdraw' conduit à 'borrow'.

7.2.2 Comparaison avec LAP

Comme LISE, LAP essaie de compléter la théorie. LAP essaie aussi de compléter la théorie pour des exemples négatifs, ce que LISE ne fait pas.

Les deux systèmes veulent reformuler ou ajuster des règles (transactions chez LISE). LAP ne possède aucune BK et il base son pas inductif sur la table des prototypes qu'il construit lui-même. LISE base son pas inductif sur la similarité des buts et se sert de BK pour compléter la théorie.

LISE, pouvant créer une transaction dans un milieu plus riche, peut accepter simultanément plusieurs endroits à réparer lors d'une explication partielle. De plus, LISE peut choisir les bons prédicats dans l'exemple parmi une multitude de prédicats. La similitude des buts permet de différencier les prédicats utiles des prédicats non pertinents. LAP peut aussi réparer plusieurs règles, mais pour cela il doit utiliser la première facette de son pas inductif.

7.3 POSTHOC

[Pazzani 1990], se fiant à des expériences au niveau cognitif, met en doute le modèle EBL où l'apprentissage peut se faire à partir d'un seul exemple: l'humain utilisera généralement plus qu'un exemple pour trouver les conditions nécessaires et suffisantes lors d'une tâche d'apprentissage.

L'utilisation des connaissances antérieures (*prior knowledge*) peut certes jouer un rôle important en apprentissage. Souvent, ces connaissances ne sont pas du style EBL (i.e. faites de généralisations universellement vraies et bien structurées), mais sont vues plutôt comme des jalons pouvant influencer l'apprentissage. EBL devient alors une théorie d'influence (*Influence theory*). La théorie du domaine doit contenir des généralisations de certaines tendances pouvant aider l'apprentissage, et la façon de combiner ces tendances peut être ignorée par la théorie. De plus, ces influences peuvent être incomplètes et incorrectes, et parfois même, pour certaines tâches, elles peuvent être complètement inconnues.

Deux tâches importantes semblent intéresser Pazzani: une tâche de prédiction (*predictive task*) utilisant une connaissance antérieure, une tâche d'identification de concept n'ayant normalement pas accès à une connaissance antérieure.

Donc, Pazzani veut créer un modèle informatique alliant EBL et SBL, mais dans lequel EBL doit être modifié pour devenir une théorie d'influence. De plus, son modèle doit refléter certains faits acceptés et prouvés en psychologie cognitive. Son modèle devrait fonctionner pour les deux genres de tâches, soit avec ou non l'utilisation de EBL.

7.3.1 POSTHOC: un modèle informatique

POSTHOC est un modèle informatique qui veut simuler l'apprentissage avec ou sans connaissances antérieures. De plus, POSTHOC veut être en accord avec les jalons acceptés en psychologie cognitive:

- Dans des tâches d'identification de concept, les concepts conjonctifs sont plus faciles à apprendre que les concepts disjonctifs;
- L'influence d'une connaissance antérieure domine l'influence de la forme logique du concept tel que décrit précédemment;

- Un concept disjonctif concernant un domaine connu peut être plus facile à apprendre qu'un concept conjonctif sans influence de connaissances;
- Un concept disjonctif avec influence est plus facile à apprendre qu'un concept disjonctif sans influence.

Le modèle doit intégrer EBL et SBL, mais EBL doit être modifiée pour jouer un rôle plus faible: ces généralisations, quoique possiblement vraies, ne représenteront que des influences éparses sur l'apprentissage. Le modèle doit être capable de fonctionner sans la présence de EBL ou de la théorie d'influence.

La théorie d'influence est représentée par deux composantes: les influences et les règles d'inférences permettant de reconnaître la présence d'influences dans les exemples d'entraînement. Une influence se représente comme le triplet (type facteur effet). Par exemple, l'influence 'Il est plus facile de gonfler un objet moins élastique' se traduit en '(easier less-elastic inflate)'.

Voici la théorie d'influences de PAZZANI pour simuler le cas de plusieurs expériences cognitives décrites dans [Pazzani 1990].

INFLUENCES:			
(easier	strong-actor		inflate)
(easier	less-elastic		inflate)
REGLES D'INFERENCE:			
(IMPLIES	[act stretch]		less-elastic)
(IMPLIES	[age adult]		old-actor)
(IMPLIES	[old-actor]		strong-actor)

Figure 66. Théorie d'influence pour POSTHOC

Cette théorie mentionne les faits pouvant rendre plus facile le gonflement d'un ballon, mais elle ne mentionne pas les relations entre les influences.

Un exemple est décrit avec un ensemble d'attributs et une classification par exemple, 'inflate' (figure 67).

[size small], [color yellow], [age adult], [act stretch]	∈	inflate.
--	---	----------

Figure 67. Un exemple de POSTHOC pour une tâche de prédiction

Cet exemple fait partie d'une tâche de prédiction car, par les règles d'inférence, on peut voir deux influences possibles sur l'effet '*inflate*': 'strong-actor' et 'less-elastic'. Si la classification ne correspond à aucun des effets de la théorie d'influence, alors la tâche en est une d'identification de concept et aucune théorie d'influence ne peut aider.

[size large], [color purple], [age child], [act dip]	∈	alpha.
--	---	--------

Figure 68. Un exemple de POSTHOC pour une tâche de classification

Ce deuxième exemple (figure 68) représente une tâche d'identification ou de classification de concept: en effet aucune influence dans la théorie de la figure 66 ne peut nous amener à 'alpha'.

Une seule hypothèse est considérée par POSTHOC; cette hypothèse se raffine ou se change complètement avec les exemples qui sont traités incrémentalement. L'hypothèse ne peut contenir que les attributs des exemples d'entraînements.

7.3.2 POSTHOC: un système à règles de production

POSTHOC, à partir de trois ensembles de règles de production, peut produire une nouvelle hypothèse pour trois cas spécifiques: initialisation, erreur de commission et erreur d'omission.

Pour chaque cas, POSTHOC donne priorité à la théorie d'influence pour produire son hypothèse. Si aucune influence n'est présente, POSTHOC passe à d'autres règles qui privilégient d'abord les conjonctions, puis les disjonctions.

Accordant plus de poids à la présence d'influences pour créer une nouvelle hypothèse, Pazzani suppose que les humains considèrent en premier les hypothèses consistantes avec leurs connaissances antérieures. D'ailleurs, [Pazzani 1990] contient des expériences cognitives vérifiant ce fait.

Examinons ces règles de production pour deux cas: l'initialisation et erreur d'omission. Les règles pour l'initialisation sont résumées à la figure 69.

- | | |
|----|--|
| 1. | SI une influence est présente dans l'exemple
ALORS initialiser l'hypothèse à une simple conjonction représentant les aspects de cette influence |
| 2. | SI VRAI
ALORS initialiser l'hypothèse à une conjonction de tous les aspects compris dans l'exemple initial. |

Figure 69. Règles de production pour l'INITIALISATION par POSTHOC

Pour le cas de l'initialisation, POSTHOC recherche la présence d'influence en premier et garde seulement une seule influence parmi plusieurs. En second lieu, si aucune influence n'est présente, il garde la conjonction décrivant l'exemple. POSTHOC donne priorité à la connaissance antérieure et ensuite à la forme logique de la conjonction. Il se fie aux autres exemples pour épurer ou améliorer son hypothèse.

Pour le premier exemple présenté (figure 67), POSTHOC se sert de sa théorie d'influence pour s'apercevoir qu'un adulte est un acteur assez fort pour gonfler un ballon. Advenant le cas que plusieurs influences sont présentes, il choisit, au hasard, une seule influence pour initialiser son hypothèse (figure 70). Les exemples suivants pourront servir à corriger l'hypothèse advenant un oubli lors de l'initialisation.

[age adult]	->	inflate.
-------------	----	----------

Figure 70. Une hypothèse de POSTHOC, employant la théorie d'influence

Le second exemple (figure 68) amène POSTHOC à choisir toute la conjonction comme hypothèse (figure 71) puisqu'aucune influence ne nous aide pour la classification 'alpha'.

[size large], [color purple], [age child], [act dip]	->	alpha.
--	----	--------

Figure 71. Une hypothèse de POSTHOC, sans employer de théorie d'influence

Les erreurs d'omission sont tous les exemples positifs qui ne peuvent pas être acceptés par l'hypothèse. La figure 72 présente les règles de production pour traiter un exemple omis.

- | | |
|----|---|
| 1. | SI l'hypothèse est consistante avec la théorie d'influence
ET un aspect dans l'exemple indique la présence d'une influence additionnelle
ALORS créer une disjonction de l'hypothèse courante et une conjonction
représentant les aspects de cette nouvelle influence. |
| 2. | SI l'hypothèse est une simple conjonction
ET un aspect de la conjonction est absent de l'exemple
ET la conjonction contient plus d'un aspect
ALORS éliminer cet aspect de la conjonction. |
| 3. | SI VRAI
ALORS créer une nouvelle disjonction de l'hypothèse courante et d'un aspect pris
au hasard dans l'exemple, puis simplifier la disjonction |

Figure 72. Système à règles de production pour les ERREURS D'OMISSION

Ici, la priorité est donnée aux influences, puis à la forme logique 'conjonction' qui est épurée et enfin à la disjonction.

7.3.3 POSTHOC: des résultats

Les résultats obtenus par POSTHOC se comparent à ceux obtenus lors d'expériences psychologiques:

- les conjonctions sont plus faciles à apprendre que les disjonctions;
- les disjonctions concernant une connaissance antérieure sont plus faciles à apprendre que les conjonctions sans cette connaissance;
- les disjonctions avec connaissance antérieure sont plus faciles à apprendre que celles sans connaissance antérieure.

POSTHOC, par son modèle, a conjecturé deux nouveaux aspects de l'apprentissage chez les humains:

- les concepts conjonctifs consistants (avec connaissance antérieure) seront plus faciles que les concepts conjonctifs sans cette connaissance;
- les humains, devant une multitude d'hypothèses plausibles, choisiront celles en confirmité avec leurs connaissance antérieures si présentes.

Pazzani a vérifié par des expériences avec des humains la pertinence de ces hypothèses. Il a utilisé son système pour voir l'effet de la théorie d'influence sur l'apprentissage d'un même concept dans les cinq cas suivants:

- théorie complète
- théorie incomplète
- théorie partiellement correcte
- théorie incorrecte
- aucune théorie.

Son système a réussi à apprendre le concept dans tous les cas avec son ensemble de règles de production. Il va de soi qu'il fut plus difficile d'apprendre avec la théorie incorrecte. Un des faits le plus surprenant est que le système a appris plus facilement avec une théorie partiellement correcte qu'avec aucune théorie.

POSTHOC possède plusieurs limitations. Par exemple il ne peut en aucun temps apprendre de nouvelles influences.

7.3.4 Comparaison avec LAP

POSTHOC veut apprendre un concept à partir d'exemples, et tous les exemples semblent être assez prototypiques. Son modèle est très innovateur car il voit la connaissance d'arrière-plan comme une théorie pouvant influencer son apprentissage. Dans EBL, tous les influences sont reliées entre elles et forment un arbre lors de la déduction. L'apport de POSTHOC est son modèle cognitif. Il ne peut pas traiter les marginaux comme LAP. De plus, POSTHOC ne peut apprendre de nouvelles influences tandis que LAP ne cherche qu'à trouver le moyen de raffiner la théorie. LAP travaille avec des théories très structurées et il cherche à améliorer cette structure dans le but de pouvoir accepter des marginaux.

8 TRAVAUX FUTURS

- La méthode proposée dans cette thèse réarrange et classifie toute l'information disponible. Cette classification est sous-employée par LAP présentement: il s'en sert seulement pour les marginaux. La table TPN pourrait servir à induire de nouvelles règles pour les prototypes, surtout lors de l'extension du langage. De nouveaux attributs de ce langage pourraient certes servir à mieux cerner les prototypes. En effet, toute théorie peut avoir des manques et un nouveau langage peut aider à les combler. Il est certes possible de modifier LAP pour qu'il puisse chercher les manques possibles compris dans toute théorie.
- Notre approche de la marginalité pourrait aider à raffiner une grammaire utilisée en langage naturel. La catégorisation des prédicats et les tables de prototypes aideraient à suggérer des règles pour traiter des structures marginales de phrases. Ces règles pourraient être ajoutées à la grammaire typique.
- Les deux pas inductifs pourraient être employés pour raffiner une taxonomie afin d'en déterminer des sous-classes. A partir d'une taxonomie très générale des mammifères, le système essaierait de trouver des ordres et sous-ordres.

9 CONCLUSION

Notre recherche propose une méthode générale et formelle permettant à un système jugé statique de se transformer en un système adaptatif, c'est-à-dire capable d'ajuster dynamiquement sa théorie du domaine en fonction d'exemples inconnus au moment de la formulation de la théorie.

Pour ce faire, nous employons les faiblesses et les forces de la théorie. La théorie est analysée en catégorisant les prédicats selon leur rôle joué dans la preuve construite pour ce marginal. La théorie sert de biais déclaratif très puissant pour classer l'information disparate contenue dans les exemples et pour rendre cette information plus fonctionnelle en lui associant une certaine signification 'sémantique'. Pour ce faire, le pouvoir déclaratif de la théorie sert à construire deux tables pour les prototypes: une table des prédicats utilisés par la théorie et une table contenant les prédicats non-utilisés par la théorie.

Une des innovations de notre méthode est de se servir de deux biais pour traiter les marginaux: le biais déclaratif de la théorie et le biais empirique des exemples utilisés.

Nous proposons une nouvelle façon d'employer un biais empirique. Habituellement, les méthodes inductives sont employées pour trouver les caractéristiques communes entre les exemples positifs. Dans LAP, pour notre marginal positif, on essaie plutôt de trouver une caractéristique différente des prototypes positifs. Le biais empirique ne concerne pas une succession d'exemples similaires de marginaux (ces derniers sont plutôt rares). Le biais empirique se retrouve dans les tables associées aux prototypes, et un marginal cherche à se différencier de ces prototypes: pour les marginaux, ces tables jouent un rôle similaire à l'ensemble G de "Version Space".

La catégorisation des prédicats, les tables de prototypes, et notre heuristique "vouloir trouver une raison à la marginalité" font du système un tout cohérent qui utilise une information non structurée et non significative attachée aux exemples pour la restructurer et lui donner une saveur sémantique de recherche de marginalité.

Un article tiré de cette thèse [Plante and Matwin 1990], fut présenté au symposium international "Methodologies for Intelligent Systems" ISMIS 90.

10 RÉFÉRENCES

Bergadano, F. and A. Giordana (1988), "A Knowledge Intensive Approach to Concept Induction", Proceedings of 5th International Conference on Machine Learning, Los Altos, Calif., Morgan Kaufman, pp.305-317.

Bergadano, F., S. Matwin, R. Michalski and J. Zhang (1988), "Measuring Quality of Concept Description", Proceedings of Proceedings of the Third European Working Session on Learning, Pitman Publishing, London, pp.1-14.

Dietterich, T. and R. Michalski (1981), "Inductive Learning of Structural Descriptions", *Artificial Intelligence*, vol. 16.

Dietterich, T. G. (1986), "Learning at the Knowledge Level", *Machine Learning*, vol. 1, no 3.

Dietterich, T. G. and N. S. Flann (1989), "A Study of Explanation-Based Methods for Inductive Learning", *Machine Learning*, vol. 4, pp.187-226.

Ellman, T. (1989), "Explanation-Based Learning: A Survey of Programs and Perspective.", *ACM Computing Surveys*, 2, pp.163-221.

Genest, J., S. Matwin and B. Plante (1990), "Explanation-Based Learning with Incomplete Theories: A Three-Step Approach", Proceedings of 7th International Conference on Machine Learning, Austin, Texas, Morgan Kaufmann.

Hall, R. J. (1988), "Learning By Failing to Explain: Using Partial Explanations to Learn in Incomplete or Intractable Domains", *Machine Learning*, vol. 3, no 1, pp.45-77.

Hirsh, H. (1988), "Knowledge as bias", Proceedings of Proceedings of the First International Workshop on Change of Representation and Inductive Bias, Briarcliff, NY, pp.186-192.

Holte, R., L. E. Acker and B. Porter (1989), "Concept Learning and the Problem of Small Disjuncts", Proceedings of 11th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, pp.813-818.

Kodratoff, Y. (1988), Introduction to Machine Learning, Morgan Kaufmann.

Kokar, M. (1990), "Specifying the Knowledge Level of A Learning Agent", Proceedings of Fifth International Symposium on Methodologies for Intelligent Systems, Knoxville, Tennessee, North-Holland, pp.396-403.

Krawchuk, B. J. and I. H. Witten (1988), "Problem Solvers That Learn", Proceedings of 3th European Working Session on Learning, Morgan Kaufmann, pp.93-106.

Michalski, R. S. (1983), "A Theory and Methodology of Inductive Learning" in Machine Learning, An Artificial Intelligence Approach, Morgan Kaufmann.

Mitchell, T. (1982), "Generalization as Search", *Artificial Intelligence*, vol. 18, no 2, pp.203-226.

Mitchell, T., R. M. Keller and S. T. Kedar-Cabelli (1986), "Explanation-Based Generalization: A Unifying View", *Machine Learning*, vol. 1, pp.47-80.

Mitchell, T., M., P. Utgoff E. and R. Banerji (1983), "Learning by experimentation: Acquiring and Refining Problem-Solving Heuristics" in Machine Learning, An Artificial Intelligence Approach, Morgan Kaufmann.

Mooney, R. and D. Ourston (1989), "Induction Over the Unexplained: Integrated Learning of Concepts with Both Explainable and Conventional Aspects", Proceedings of 6th International Workshop on Machine Learning, Los Altos, Calif., Morgan Kaufmann, pp.5-8.

Morin, J. (1990), "Apprentissage Incremental de Contraintes Procédurales par une Méthode Mixte", Ottawa, Maîtrise ès sciences appliquées en Informatique.

Pazzani, M. (1990), "A Computational Model of the Influence of Prior Knowledge on Concept Acquisition", Draft Paper University of California Irvine.

Pazzani, M. (1988), "Integrated Learning with Incorrect and Incomplete Theories", Proceedings of 5th National Conference on Artificial Intelligence, Morgan Kaufmann, pp.555-559.

Plante, B. and S. Matwin (1990), "Learning Flexible Concepts Through Theory Revision", Proceedings of Fifth International Symposium on Methodologies for Intelligent Systems, Knoxville, Tennessee, North-Holland, pp.412-420.

Puget, J. F. (1989), "Learning Concept Negation", Proceedings of 4th European Working Session on Learning, Morgan Kaufmann, pp.179-187.

Quinlan, J. R. (1986), "Induction on Decision Trees", *Machine Learning*, vol. 1, pp.81-106.

Utgoff, P., E. (1986), "Shift of Bias for Inductive Concept Learning" in Machine Learning, An Artificial Intelligence Approach, Morgan Kaufmann, pp.107-148.

Winston, P. H., T. O. Binford, B. Katz and M. Lowry (1983), "Learning Physical Description from Functional Definitions, Examples and Precedents", Proceedings of Natural Conference on Artificial Intelligence, Washington, D.C., Morgan Kaufmann, pp.433-439.