



HAL
open science

De l'ingénierie à l'apprentissage des connaissances d'adaptation en raisonnement à partir de cas

Amélie Cordier, Béatrice Fuchs, Alain Mille

► To cite this version:

Amélie Cordier, Béatrice Fuchs, Alain Mille. De l'ingénierie à l'apprentissage des connaissances d'adaptation en raisonnement à partir de cas. IC - 17èmes Journées francophones d'Ingénierie des Connaissances, Jun 2006, Nantes, France. pp.41-50. hal-01025803

HAL Id: hal-01025803

<https://inria.hal.science/hal-01025803v1>

Submitted on 18 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

De l'Ingénierie à l'Apprentissage des connaissances d'adaptation en Raisonnement à Partir de Cas

Amélie Cordier, Béatrice Fuchs, Alain Mille

LIRIS UMR 5205

CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/
Université Lumière Lyon 2/Ecole Centrale de Lyon Bâtiment Nautibus (710),
43, Boulevard du 11 Novembre 1918 - 69622 VILLEURBANNE CEDEX
{acordier,bfuchs,amille}@liris.cnrs.fr et <http://liris.cnrs.fr>

Résumé

Le raisonnement à partir de cas (RÀPC) consiste à résoudre un problème en se remémorant et en adaptant des cas passés déjà résolus. Les systèmes de RÀPC manipulent des connaissances de natures diverses : les cas, les connaissances du domaine, les connaissances de similarité et d'adaptation. Les cas sont collectés de manière graduelle lors de l'utilisation du système et la base de cas s'enrichit de façon incrémentale, alors que les autres types de connaissances sont généralement acquis au moment de la conception du système. En particulier les connaissances d'adaptation sont souvent difficiles à modéliser, imprécises et incomplètes. Proposer des mécanismes permettant l'acquisition ou l'apprentissage de telles connaissances pendant l'utilisation du système est donc particulièrement pertinent. Dans cet article, nous étudions dans un premier temps le raisonnement à partir de cas du point de vue des connaissances manipulées puis au travers du cycle de raisonnement. Nous montrons ensuite dans quelle mesure l'apprentissage des connaissances en RÀPC peut tirer profit de l'unification des connaissances de similarité et des connaissances d'adaptation. Nous identifions les différentes cibles d'apprentissage des connaissances d'adaptation, puis nous proposons d'approfondir l'un des scénarios d'apprentissage que nous aurons présenté.

Mots-clés : Raisonnement à partir de cas, apprentissage des connaissances d'adaptation.

1 Introduction

Le raisonnement à partir de cas (RÀPC) est un paradigme de raisonnement qui permet de résoudre de nouveaux problèmes en adaptant les solutions de problèmes passés déjà résolus. Le cycle du RÀPC se décompose en cinq étapes : élaboration, remémoration, adaptation, révision et apprentissage. Chaque étape joue un rôle particulier dans la résolution du problème et mobilise des connaissances spécifiques.

En RÀPC, les connaissances de base sont les expériences de résolution de problème : les cas. Elles donnent lieu, lors d'un cycle de raisonnement, à leur mémorisation dans une base de cas qui peut être éventuellement réorganisée en conséquence. La mémorisation du cas résolu est considérée comme l'approche la plus traditionnelle de l'apprentissage en RÀPC. Les cas mémorisés pourront être exploités pour des cycles de raisonnement ultérieurs et améliorent graduellement la compétence du système.

Les systèmes de RÀPC ont longtemps été considérés comme des alternatives intéressantes aux systèmes à base de connaissances car ils demandent théoriquement un moindre effort d'ingénierie des connaissances pour être mis en œuvre. En effet, le raisonnement à partir de cas est particulièrement adapté aux situations dans lesquelles la théorie du domaine est faible ou peu formalisable. Il a longtemps été argumenté que le RÀPC était une solution au goulet d'étranglement de l'acquisition des connaissances dans la mesure où il est plus facile de collecter un ensemble de cas que de construire une base de connaissances.

En revanche, le RÀPC ne permet pas de se passer d'une base de connaissances et on se trouve confronté au problème de leur acquisition. En effet, en plus des cas, les systèmes de RÀPC utilisent également d'autres types de connaissances sur lesquels ils s'appuient pour raisonner : une ontologie du domaine, des connaissances de similarité et des connaissances d'adaptation notamment.

Les connaissances de similarité sont utilisées pour se remémorer les cas candidats et les connaissances d'adaptation pour adapter les solutions des cas remémorés. Or, l'expérience montre que les connaissances de similarité et d'adaptation disponibles au départ sont difficile à modéliser, imprécises ou incomplètes, et de plus, elles peuvent évoluer au cours du temps. Par conséquent, il est souhaitable de disposer d'outils permettant d'aider à leur acquisition et/ou d'un processus d'apprentissage de ces connaissances permettant de les raffiner et de les faire évoluer au fur et à mesure de l'utilisation du système.

Ceci soulève la problématique de la gestion de la base de connaissances d'un système de RÀPC depuis sa concep-

tion jusqu'à son exploitation et sa maintenance. Dans cet article nous proposons d'aborder le RÀPC sous l'angle de la gestion des connaissances. Dans un premier temps, nous étudions la problématique de la gestion des connaissances lors de la conception et l'utilisation des systèmes de RÀPC puis nous analysons le cycle du raisonnement en mettant en évidence les différentes connaissances mobilisées à chaque étape du cycle. En particulier nous montrons à quel point les connaissances de similarité et d'adaptation, connaissances essentielles au raisonnement, sont difficiles à formaliser et, de plus, évoluent. Après avoir discuté du lien étroit entre ces deux types de connaissances, nous montrons comment différentes approches de l'apprentissage peuvent exploiter un tel rapprochement. Nous présentons le modèle d'adaptation par substitution sur lequel nous nous appuyons puis nous proposons plusieurs scénarios d'apprentissage des connaissances d'adaptation. Nous détaillons l'un de ces scénarios et mettons en évidence les problèmes soulevés dans ce type de démarche. Ceci nous permet d'évoquer le rôle que doit jouer l'utilisateur dans une telle démarche.

2 Les connaissances en RàPC : typologie

Les systèmes de RÀPC sont des systèmes à base de connaissances (SBC) qui, si l'on s'en tient à la proposition de Richter (18) exploitent quatre conteneurs de connaissances distincts : le vocabulaire de description du domaine (l'ontologie du domaine), les cas, les connaissances de similarité et les connaissances de transformation de la solution que nous appelons connaissances d'adaptation.

2.1 La gestion des connaissances dans un système de RàPC

Il est possible de distinguer plusieurs phases dans le cycle de vie d'un système de RÀPC : la phase de conception, la phase de production et la phase de maintenance.

Durant la phase de conception et de réalisation du système, les concepteurs définissent - en accord avec les experts du domaine - les méthodes de résolution de problèmes qui seront utilisées par la suite. Un effort d'ingénierie des connaissances important doit être fourni pour constituer les bases de connaissances du système, définir une base de cas résolus initiale, décrire les connaissances du domaine, les connaissances de similarité et les connaissances d'adaptation. Le système peut également être utilisé avec des cas connus afin d'instancier la base de cas et de fournir une base au raisonnement. La question du formalisme de représentation des connaissances se pose également à ce moment. Les principaux acteurs de cette phase sont bien entendu les experts qui sont de véritables vecteurs de la connaissance du domaine ainsi que les concepteurs qui facilitent le passage du niveau connaissance au niveau symbole (16).

Durant la phase de production, le système est utilisé pour résoudre - ou assister la résolution - de nouveaux problèmes. Le cycle de raisonnement appliqué pour mener cette tâche à bien est étudié dans le paragraphe suivant. Des problèmes peuvent être posés au système soit par des utilisateurs soit par des experts du système. Les interactions entre utilisateurs et système ont lieu en début et en fin de cycle mais aussi pendant la phase de production, comme nous le verrons par la suite. Dès que le système est mis en production, un processus de maintenance doit permettre de faire évoluer les connaissances acquises initialement. À la fin de chaque étape de résolution de problème, le nouveau cas résolu est mémorisé dans la base de cas pour pouvoir être réutilisé ultérieurement. Il en résulte une augmentation graduelle de taille de la base de cas, ce qui met donc en évidence le besoin d'organiser et de maintenir la base de cas tout au long de la vie du système. Pour répondre à ce besoin, plusieurs travaux proposent des techniques d'indexation ou de classification pour faciliter la remémoration des cas. D'autres approches s'appuient sur des stratégies de rétention et d'oubli (20) pour ne retenir que les cas les plus pertinents et éviter l'explosion de la taille de la base de cas. Parmi ces différentes approches, certaines ont lieu durant la phase de mémorisation tandis que d'autres sont effectuées indépendamment de la phase de production. Enfin, notons que le système peut effectuer seul les opérations de maintenance, être sollicité par l'expert ou l'utilisateur pour les accomplir ou bien encore solliciter l'expert pour qu'il lui apporte son aide.

2.2 Cycle de raisonnement

Comme nous l'avons déjà évoqué, le raisonnement à partir de cas résout de nouveaux problèmes en s'appuyant sur la remémoration et l'adaptation de problèmes passés déjà résolus. Le cycle du RÀPC est constitué de cinq étapes :

- *Élaboration.* Les informations nécessaires à la formulation d'un problème sont collectés et structurés de façon à constituer un nouveau cas : le cas cible. Lors de cette étape, le système sollicite l'utilisateur ou l'environnement extérieur (bases de données, systèmes d'information) pour obtenir l'ensemble des informations nécessaires à la poursuite du raisonnement.
- *Remémoration.* L'étape de remémoration consiste à chercher dans la base de cas un ou plusieurs cas passés résolus jugés similaires au cas cible. La sélection des cas similaires s'appuie sur une mesure de similarité. Certains systèmes conservent plusieurs cas remémorés pour les combiner par la suite, mais la plupart du temps, un seul cas est conservé pour la suite du processus. On l'appelle le cas source. La sélection du cas source parmi les cas remémorés peut être la conséquence d'une sélection par le système ou bien résulter d'un choix effectué par l'utilisateur.
- *Réutilisation.* Cette étape permet d'obtenir une solu-

tion au cas cible à partir de la solution du cas source sélectionné qui est tout d'abord recopiée puis éventuellement adaptée afin de satisfaire les contraintes du problème posé. L'adaptation s'appuie sur des connaissances d'adaptation qui peuvent prendre différentes formes selon les systèmes.

- *Révision*. Il est possible que la solution proposée par le système ne convienne pas à l'utilisateur ou bien qu'une fois mise en application, elle s'avère inapte à résoudre le problème posé. L'utilisateur a donc, pendant la phase de révision, l'opportunité de modifier, corriger ou même refuser la solution proposée. L'étape de révision permet d'identifier les causes éventuelles des échecs et de proposer des adaptations supplémentaires pour conduire à une solution satisfaisante : il s'agit du cas révisé. Cette étape est à la base du processus d'apprentissage permettant à la fois d'améliorer des connaissances d'adaptation existantes et d'en faire émerger de nouvelles.
- *Mémorisation*. Traditionnellement, on considère l'étape de mémorisation comme l'étape durant laquelle la base de cas est enrichie par le cas cible révisé. Cette mémorisation implique une mise à jour des index permettant de retrouver les cas et parfois un processus de maintenance pour réorganiser la base de cas. Mais l'étape de mémorisation est également le siège de l'apprentissage d'autres types de connaissances. En effet, c'est durant cette étape que l'on peut concrétiser les efforts effectués durant les autres étapes pour apprendre d'autres connaissances.

2.3 Acquisition et apprentissage des connaissances

L'étude du cycle de raisonnement d'un cas a mis en évidence la diversité des connaissances mobilisées en RàPC. Le tableau 1 en propose une synthèse. Pour chaque type de connaissance sont précisés : les différentes formes que peuvent prendre les connaissances, les étapes de la vie du système durant lesquelles ces connaissances peuvent être acquises et les méthodes mises en œuvre pour leur acquisition ou leur apprentissage.

On observe qu'à l'exception des connaissances du domaine, rares sont les connaissances qui sont formalisables *a priori* dans le système. En effet, même s'il est possible de représenter des connaissances de similarité ou d'adaptation dans la base de connaissances initiale, ces connaissances restent imprécises voire incertaines et doivent être améliorées au fur et à mesure de l'utilisation du système.

2.4 Dualité des connaissances de similarité et d'adaptation

La relation entre les connaissances de similarité et les connaissances d'adaptation est intéressante à étudier.

L'adaptation est l'une des étapes les plus difficiles du RàPC, par conséquent, tout effort visant à la faciliter est judicieux.

Dans (19), Smyth propose une approche de remémoration guidée par l'adaptabilité (cf. (21)). L'argument est que ce ne sont pas toujours les cas sources les plus similaires au cas cible qui sont les plus faciles à adapter, en particulier lorsque la mesure de similarité s'appuie sur des caractéristiques de surface. La remémoration doit donc rechercher non seulement des cas similaires mais surtout des cas facilement adaptables.

Dans le même esprit, Leake suggère qu'une bonne remémoration d'un cas permet de faciliter l'effort d'adaptation (11). En effet, les traditionnelles mesures de similarité sémantiques peuvent conduire à des résultats médiocres dans la mesure où elles permettent parfois de se remémorer des cas sources, certes très "ressemblants" au cas cible, mais difficiles voire impossibles à adapter. Cette observation montre la limite des mesures de similarité au regard du processus complet de raisonnement. Leake propose donc d'inclure à la mesure de similarité une notion de "coût d'adaptation" afin de la rendre plus pertinente. Ainsi, dans cette approche, l'évaluation de la similarité entre le cas cible et les différents cas de la base se fait en deux temps : tout d'abord, une mesure de similarité classique est effectuée par comparaison des descripteurs des cas, puis les cas les plus similaires remémorés sont priorisés en fonction de leur adaptabilité.

Lieber propose quant à lui une approche de l'adaptation qui passe par l'utilisation de chemins de similarité. Derrière la notion de chemins de similarité se trouve l'idée de décomposition de l'adaptation en sous tâches d'adaptation plus simples. Pour mettre en évidence des similarités entre deux problèmes complexes, il faut souvent disposer de connaissances dépendantes du domaine. L'approche proposée dans (12) vise à faire décroître la difficulté de l'adaptation en faisant croître la similarité entre les problèmes ce qui passe par la décomposition d'un problème complexe en plusieurs sous problèmes plus simples. Les problèmes intermédiaires sont liés les uns aux autres grâce à des relations. À chaque relation est associée une fonction d'adaptation spécifique permettant de passer d'un problème à un autre. Un chemin de similarité est donc constitué d'une succession linéaire de problèmes intermédiaires liés les uns aux autres par des relations. La première étape de l'adaptation qui consiste à construire le chemin de similarité peut être effectuée pendant la phase de remémoration. Durant la deuxième phase de l'adaptation, il ne reste plus qu'à calculer les petites adaptations. Dans (13), les auteurs montrent ainsi sur un cas concret (le traitement du cancer du sein) comment la notion de chemins de similarité peut apparaître comme un outil d'aide à l'acquisition et à la modélisation des connaissances d'adaptation.

Ces trois exemples mettent clairement en évidence la relation de dualité existant entre connaissances de similarité et connaissances d'adaptation. D'une façon générale, il n'est

Type de connaissance	Forme	Étape d'acquisition	Méthode
Cas	Parties problème et solution (ensembles de descripteurs), Traces du raisonnement (étapes conduisant du problème à la solution)	Conception : utilisation de cas connus pour "entraîner" le système Mémorisation : mise en mémoire du cas résolu lors d'un cycle de raisonnement	Classification Indexation
Connaissances du domaine	Concepts : propriétés et relations avec d'autres concepts, Règles, Dépendances	Acquisition initiale relativement aisée si la théorie du domaine est faible	Description et modélisation par l'expert
Connaissances de similarité	Mesures numériques prédéfinies, Mesures empiriques basées sur la comparaison de descripteurs, Mesures plus complexes avec prise en compte de l'adaptabilité, Poids, Chemins de similarités, Etc	Acquisition initiale non aisée, pas de méthodologie pour les concevoir, Mémorisation : acquisition de nouvelles connaissances et amélioration des connaissances existantes	Modélisation par l'expert, Apprentissage introspectif, Apprentissage symbolique automatique (fouille de données, réseaux de neurones ...) Etc
Connaissances d'adaptation	Règles d'adaptation, Opérateurs d'adaptation, Cas d'adaptation		

TAB. 1 – Typologie des connaissances du RÀPC

pas souhaitable de considérer les différentes étapes du RÀPC séparément et indépendamment les unes des autres mais plutôt comme concourant à un objectif commun. L'étape d'élaboration par exemple vise à améliorer la mémorisation en élaborant les descripteurs adéquats. De même, l'étape de mémorisation vise à faciliter l'adaptation en utilisant des critères d'adaptabilité pour choisir un cas source. L'adaptabilité d'un cas doit donc être prise en compte dès l'étape de mémorisation. C'est pourquoi l'apprentissage des connaissances d'adaptation revêt un enjeu particulier. Dans la section suivante, nous allons nous intéresser aux stratégies qui permettent l'apprentissage de ces connaissances.

3 Apprentissage des connaissances d'adaptation

3.1 Stratégies d'apprentissage

L'adaptation est étudiée selon trois directions principales (14) : les démarches unificatrices qui proposent des modèles généraux de l'adaptation, les catalogues de stratégies d'adaptation applicables à plusieurs domaines et les méthodes d'acquisition des connaissances d'adaptation qui,

dans un domaine particulier, cherchent à mettre en évidence des principes généraux permettant d'explicitier l'adaptation. Une revue récente des méthodes existantes est présentée dans (14). On distingue différentes approches d'acquisition des connaissances d'adaptation : celles, qualifiées de "*knowledge light*" selon (22), qui consistent à réutiliser des connaissances déjà présentes dans le système pour inférer de nouvelles connaissances et celles qui cherchent à acquérir de nouvelles connaissances en exploitant les interactions entre le système et son environnement. Les approches de la première catégorie se déroulent en dehors de la phase de résolution de problème, tandis que les approches de la seconde catégorie ont lieu durant le processus de résolution et offrent donc de nombreuses possibilités d'interactions avec l'utilisateur.

L'approche de (6) peut être classée dans la première catégorie : elle consiste à déterminer des paires de cas et à utiliser les différences entre les attributs pour construire des règles d'adaptation. Les règles d'adaptation ainsi construites sont ensuite raffinées et généralisées. À chaque règle sont associées des mesures de confiance calculées en fonction du degré de généralisation des règles.

Dans le même esprit, (2) proposent une approche d'apprentissage des connaissances d'adaptation qui s'appuie sur

une technique particulière de fouille de données appelée "extraction de motifs fréquents". L'idée principale est d'exploiter les différences entre les cas de la base de cas pris deux à deux. En effet, ces différences peuvent être interprétées comme le résultat d'un effort d'adaptation. Si tel est le cas, il est alors possible d'en déduire des connaissances d'adaptation.

Parmi les approches de la deuxième catégorie, on peut citer celle de (10). L'apprentissage de connaissances chez Leake revêt plusieurs formes. Dans un premier temps, Fox et Leake ont proposé une approche exploitant le raisonnement introspectif pour donner aux systèmes la possibilité d'apprendre de nouvelles connaissances leur permettant d'améliorer leur efficacité globale. Dans (4), les auteurs appliquent un raisonnement introspectif sur l'étape d'indexation des cas. Par la suite, ils vont étendre cette approche aux autres étapes du RàPC et en particulier à l'étape d'adaptation. Dans le système DIAL, terrain d'expérimentation de cette approche, le raisonnement proposé est essentiellement centré autour de l'adaptation des cas et l'apprentissage des différents types de connaissances est plus ou moins lié à cette étape. (9) voit l'adaptation d'un cas comme un processus combinant un ensemble de transformations abstraites avec des stratégies de recherche en mémoire. Une trace des actions qui ont lieu durant une phase d'adaptation est stockée et constitue un cas d'adaptation. Ainsi, lorsqu'un nouveau cas se présente, il peut être adapté soit "à partir de rien" soit en repartant de l'utilisation de cas d'adaptation et en utilisant un raisonnement introspectif (8). Les connaissances d'adaptation sont acquises grâce à un cycle de RàPC dans le cycle de RàPC principal. Cette approche d'apprentissage des connaissances d'adaptation permet donc de raffiner au fur et à mesure les stratégies d'adaptation en adaptant les cas d'adaptation (7). Leake propose également de faire évoluer les connaissances de similarité au fur et à mesure que les connaissances d'adaptation sont apprises. L'idée est d'utiliser les connaissances contenues dans les cas d'adaptation pour prédire des coûts d'adaptation. La méthode proposée est nommée RCR (Re-application Costs and Relevance). Elle permet essentiellement d'estimer la difficulté à adapter un problème et apporte donc une précision supplémentaire à la mesure de similarité (11).

Les approches qui visent à exploiter les connaissances déjà présentes dans le système pour inférer de nouvelles connaissances d'adaptation sont entre autres limitées au "vocabulaire" de la base de cas. Elles ne permettent pas d'inférer des connaissances qui ne sont pas "explicables" avec les connaissances présentes dans l'application. De plus, elles ne donnent à l'utilisateur qu'un rôle mineur : celui de validateur des connaissances inférées. À l'inverse, les approches qui permettent d'apprendre des connaissances pendant le processus de raisonnement offrent à la fois la possibilité d'ajouter de nouvelles connaissances dans le système et l'opportunité de faire de l'utilisateur un véritable acteur du processus. Nous adhérons à la seconde approche

et notre souhait est de placer l'utilisateur au centre du processus d'apprentissage pour qu'il puisse, de manière simultanée, participer à la résolution du problème et à l'apprentissage des connaissances d'adaptation.

3.2 Apprendre à mieux adapter

Dans ce travail, nous nous appuyons sur une formalisation de l'adaptation par substitution. Le cadre de l'étude dans lequel nous nous plaçons a été introduit dans (5). Les possibilités d'extensions de cette approche à d'autres types d'adaptation ne sont pas l'objet de ce travail.

Les connaissances d'adaptation sont essentiellement constituées de dépendances (telles que présentées dans les travaux de Michel Py (17)). Les dépendances que nous utilisons sont similaires à celles utilisées dans le raisonnement par analogie. À ce sujet, le lecteur intéressé trouvera dans (15) une analogie entre la formalisation du raisonnement analogique proposée dans (17) et la formalisation du raisonnement à partir de cas.

Après avoir présenté les notions et les notations utilisées, nous identifions les sources mobilisées ainsi que les cibles visées par l'apprentissage et nous proposons quelques stratégies permettant cet apprentissage. Nous illustrons les différentes stratégies dans le domaine de l'évaluation du prix d'un véhicule automobile d'occasion. Dans ce problème, les cas sont des véhicules caractérisés par un certain nombre de descripteurs ainsi que par leur prix de vente sur le marché de l'occasion. Il s'agit alors, étant donné un certain nombre de dépendances, de calculer le prix probable de vente d'un nouveau véhicule en fonction des descripteurs connus.

3.2.1 Un modèle d'adaptation

Nous nous appuyons sur le modèle d'adaptation proposé dans (5) que nous présentons brièvement ci-après. Nous faisons l'hypothèse qu'il est possible de représenter un cas à l'aide d'un ensemble de descripteurs. Un descripteur est constitué d'un nom d'attribut et d'une valeur. On note :

- d les descripteurs des parties problèmes et D les descripteurs des parties solutions,
- $\{d_i^s\}_{i=1..n}$ les descripteurs d'un problème source et $\{D_j^s\}_{j=1..N}$, les descripteurs de sa solution
- $\{d_i^c\}_{i=1..n}$ les descripteurs d'un problème cible et $\{D_j^c\}_{j=1..N}$ les descripteurs de sa solution calculés par l'adaptation

Pour deux cas sources donnés, la remémoration estime les écarts entre des couples de descripteurs de problèmes notés (Δd_i) . L'adaptation s'appuie sur un ensemble de relations entre le problème et sa solution appelées *dépendances* indiquant que certains descripteurs de problème ont une influence sur certains descripteurs de solution. Une dépendance est un triplet $(d_i, D_j, \mathcal{I}(D_j/d_i))$ indiquant la variation du descripteur de solution D_j en fonction de celle du descripteur de problème d_i . $\mathcal{I}(D_j/d_i)$ est appelée fonction

d'influence et indique comment calculer la variation de d_i connaissant la variation de D_j . L'adaptation combine ces fonctions d'influence $\mathcal{I}(D_j/d_i)$ avec les écarts Δd_i entre descripteurs de problèmes pour estimer les variations ΔD_j à appliquer sur les descripteurs de solution source D_j^s pour obtenir les descripteurs de solution cible D_j^c .

Les dépendances sont donc essentielles car elles sont porteuses, au travers des influences, des connaissances d'adaptation. Les dépendances sont des connaissances du domaine qui doivent être estimées dès la conception du système pour que celui-ci puisse raisonner. Mais ces connaissances restent empiriques et incertaines. Par conséquent, elles doivent être raffinées lors de l'utilisation du système. Cette observation est justifiée par l'existence même d'une étape de révision dans le cycle du RÀPC. En effet, si les connaissances d'adaptation étaient certaines, le système serait en mesure de garantir que le résultat de l'adaptation est correct.

La capacité à prédire en partie la suite du raisonnement permet d'augmenter considérablement l'efficacité globale des méthodes utilisées. Par exemple, comme nous l'avons vu dans la section 2.4, prévoir l'effort d'adaptation nécessaire à un cas donné dès la phase de remémoration permet de se remémorer non pas le cas le plus similaire du point de vue des traits de surface mais celui qui sera le plus facile à adapter. Dans le modèle d'adaptation présenté ici, les dépendances explicitent la relation étroite existant entre les connaissances de similarité et les connaissances d'adaptation. En effet, les fonctions d'influence sont porteuses des connaissances d'adaptation et mettent également en relation des descripteurs de problèmes avec des descripteurs de solution, ce qui met en évidence le rôle qu'elles jouent dans l'évaluation de la similarité.

3.2.2 Cibles d'apprentissage des connaissances d'adaptation

Le modèle d'adaptation présenté ci-dessus permet d'identifier les cibles de l'apprentissage : les fonctions d'influence, les dépendances et les classes de problèmes.

Les fonctions d'influence Les fonctions d'influence permettent de calculer la variation d'un descripteur de solution en fonction d'un descripteur de problème. Elles peuvent revêtir plusieurs formes et être de complexités variables, mais la plupart du temps, elles s'apparentent à des fonctions numériques plus ou moins simples. Ces fonctions, même si elles sont déterminées par des experts lors de la conception du système, peuvent être affinées au fur et à mesure des expériences de résolution de problème. On peut par exemple, d'après l'expérience, régler les seuils d'applicabilité de la fonction ou bien ajuster certains paramètres pour rendre les fonctions de plus en plus précises. Une fonction d'influence linéaire permet par exemple de calculer une variation du prix d'un véhicule en fonction d'une variation de son kilométrage par rapport à un véhicule de référence.

Les dépendances Lors de la résolution d'un nouveau problème, un échec de l'adaptation peut conduire à la découverte d'une nouvelle dépendance. En effet, il est probable qu'une expérience mette en évidence le fait qu'un descripteur de problème jusque là ignoré ait une influence, dans certaines conditions, sur un descripteur de solution. Dans ce cas, une nouvelle dépendance est élaborée et peut être associée à un groupe de dépendances donné. Dans le domaine des véhicules d'occasion, nous pouvons par exemple constater que le prix dépend non seulement du kilométrage mais aussi de l'âge du véhicule, une autre de ses caractéristiques.

Les classes de problèmes Une classe de problèmes correspond à un ensemble de problèmes qui se résolvent en utilisant des stratégies d'adaptation similaires. Concrètement, une classe de problèmes est constituée d'un ensemble de dépendances nécessaires à la résolution d'un problème. La découverte d'une classe de problèmes correspond donc quant à elle à l'identification d'une nouvelle catégorie de problèmes qui était jusque là inconnue et donc impossible à adapter. Identifier une classe de problèmes s'apparente donc également à un apprentissage de connaissances d'adaptation. Dans le domaine des véhicules automobiles d'occasion par exemple, on peut considérer que les méthodes utilisées pour calculer le prix d'un véhicule de tourisme et celles utilisées pour calculer le prix d'un véhicule de collection ne sont pas les mêmes. Ces deux problèmes correspondent donc à deux classes de problèmes différentes.

3.2.3 Méthodes d'apprentissage

L'échec d'une adaptation par un système de RÀPC traduit un manque de connaissances d'adaptation. C'est au moment de la révision que cet échec est constaté et se traduit par les modifications apportées à la solution ou par la constatation de l'incapacité de la solution proposée à répondre au problème. L'étape de révision est donc le point de départ du processus d'apprentissage. Nous proposons ci-dessous quelques méthodes d'apprentissage des connaissances d'adaptation applicables dans certaines situations.

Exploitation de l'étape de révision Le processus d'adaptation calcule un ensemble d'écarts sur les descripteurs de solution à partir des fonctions d'influence. On note ces écarts : $\Delta_{\text{adapte}} D_j$. Appliqués aux descripteurs de solution source, ils permettent de calculer les descripteurs de solution cible (D_j^c). Ces écarts représentent donc les modifications apportées par le système.

D'autres écarts sont introduits lors de la révision du cas par l'utilisateur. Ces écarts, notés $\Delta_{\text{revise}} D_j$, permettent quant à eux de quantifier la différence entre le descripteur de solution cible D_j^c avant et après la révision effectuée par l'utilisateur. Ces écarts représentent donc l'ajustement ef-

fectué par l'utilisateur. On notera D_j^{cr} le descripteur de solution cible j après révision.

On a donc :

- $D_j^c = D_j^s + \Delta_{adapte}D_j$, et
- $\Delta_{revise}D_j = D_j^{cr} - D_j^c$

Ces notations sont reprises dans la figure 1 qui présente les relations entre les différents descripteurs en particulier lors des phases de remémoration et d'adaptation. Les D_j^{cr} sont donnés par l'utilisateur. Dès qu'ils sont connus, il est possible de calculer les $\Delta_{revise}D_j$.

Les écarts de type $\Delta_{revise}D_j$ reflètent des lacunes dans les fonctions d'influence ayant permis d'inférer les valeurs des descripteurs. La constatation de tels écarts peut donc être à l'origine du déclenchement d'un processus d'apprentissage. En effet, si une solution a été révisée par l'utilisateur avant d'être mémorisée, il est possible d'exploiter les différences représentées par les $\Delta_{adapte}D_j$ et les $\Delta_{revise}D_j$ au sein d'un processus d'apprentissage.

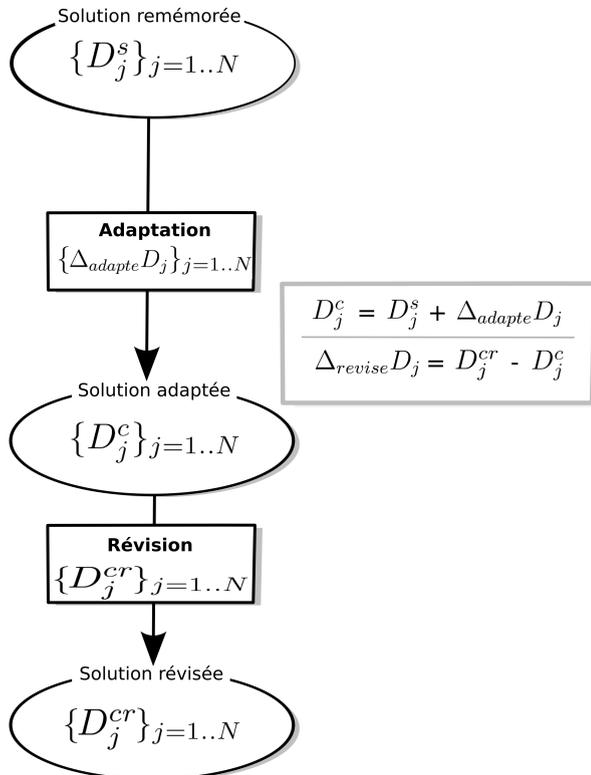


FIG. 1 – Relations entre les descripteurs de solution. Ce schéma présente les relations et les écarts entre les descripteurs de solution lors des phases d'adaptation et de révision en RàPC.

Une fonction d'influence est caractérisée par ses paramètres ainsi que par des seuils déterminant les domaines sur lesquels la fonction est applicable. L'étude des $\Delta_{revise}D_j$

et des $\Delta_{adapte}D_j$ peut permettre d'affiner chacun de ces éléments.

Dans le domaine des automobiles, il est facile de définir une fonction déterminant la variation du prix de vente (ΔD_j) en fonction de la variation du kilométrage de la voiture (Δd_i). Une fonction numérique simple permet d'indiquer qu'une variation d'un kilomètre induit une variation de prix d'un centime d'euro. Ainsi, un écart de 30 000km entre deux voitures induira un écart de 300 euros entre leurs prix respectifs. La figure 2 présente une fonction d'influence dans le cadre de l'exemple simplifié du calcul du prix d'un véhicule d'occasion.

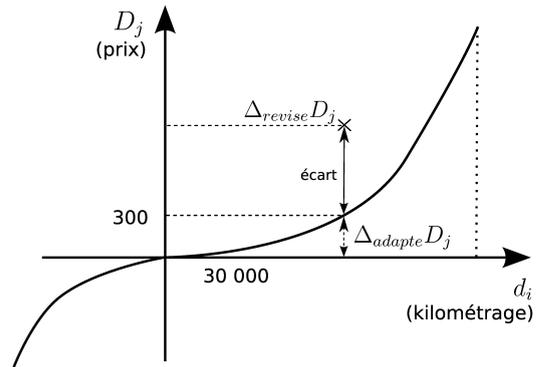


FIG. 2 – Représentation graphique d'une partie de fonction d'influence. On observe qu'au delà d'un certain seuil pour Δd_i , la fonction n'est plus définie : ceci signifie que le cas n'est plus adaptable. Ce schéma représente également l'écart entre un $\Delta_{adapte}D_j$ et un $\Delta_{revise}D_j$ pour illustrer une possibilité d'apprentissage de connaissances d'adaptation.

Remémoration sur les solutions Une autre possibilité pour l'apprentissage des connaissances d'adaptation consiste à effectuer une remémoration sur les solutions sources révisées de la base de cas et à ordonner les cas ainsi remémorés selon leur similarité avec la solution cible révisée. Si le meilleur cas du point de vue de la solution ne coïncide pas avec le cas source qui avait été utilisé pour résoudre le problème cible, alors on peut supposer qu'une ou plusieurs dépendances utilisées pour la remémoration sont incorrectes ou incomplètes et doivent être ajustées. Dans le domaine des voitures d'occasion, l'ajustement d'une dépendance peut se traduire par une modification de la fonction d'influence présentée ci-dessus ou bien par la découverte d'une nouvelle dépendance : le fait que la puissance fiscale intervienne dans le calcul du prix du véhicule par exemple.

Nous pensons que différentes méthodes peuvent permettre l'apprentissage des connaissances d'adaptation dans cette situation. Plusieurs pistes peuvent être explorées : l'ap-

plication d'un raisonnement introspectif portant sur une comparaison des descripteurs pour déduire les modifications à apporter aux fonctions d'influence ou bien par une coopération avec l'utilisateur du système durant laquelle ce dernier peut spécifier lui-même comment les dépendances doivent être corrigées.

Rejouer le cycle avec l'utilisateur Si l'étape de révision ne permet pas d'aboutir à une solution satisfaisante pour le problème posé, il est alors nécessaire d'impliquer l'utilisateur dans le cycle du raisonnement pour tenter de résoudre à nouveau le problème. Nous pensons qu'il est utile de rejouer tout le cycle du raisonnement avec l'utilisateur en l'assistant à chaque étape. Cette assistance peut consister à lui présenter les connaissances qui ont été utilisées et qui expliquent le raisonnement tenu par le système. En invitant l'expert à intervenir pour préciser ou compléter les connaissances exploitables pour le problème en cours, on espère aboutir à une solution plus satisfaisante.

Nous pensons qu'il est également possible d'apprendre des connaissances d'adaptation en exploitant la trace des actions de l'utilisateur. Ces connaissances peuvent certainement prendre la forme de cas d'adaptation, qui seront exploitables ultérieurement par le système, elles peuvent aussi permettre la découverte de nouvelles classes de problèmes ou même guider le choix d'une classe de problèmes pour un problème donné.

Scénario d'ensemble du processus d'apprentissage En guise de synthèse, la figure 3 présente différentes situations d'apprentissage possibles ainsi que des méthodes applicables dans chaque situation. Notons qu'il est souhaitable que la coopération avec l'utilisateur puisse avoir lieu dès le début du cycle de raisonnement et non uniquement après un échec du système.

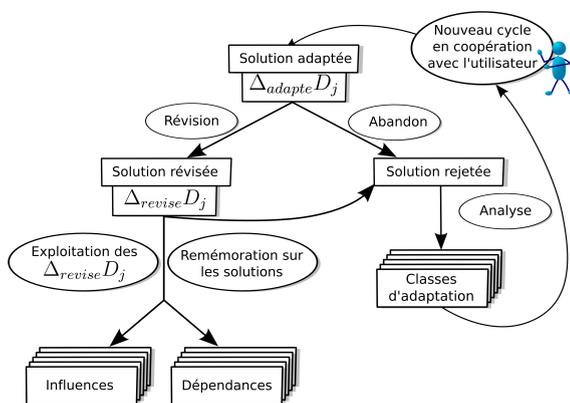


FIG. 3 – Scénario du processus d'apprentissage.

Enfin, il est possible de faire le lien avec les approches d'extraction des connaissances à partir des données (ECD)

qui peuvent avantageusement compléter les approches interactives décrites ci-dessus. Des approches telles que (2) peuvent apporter une assistance à la découverte de nouvelles dépendances candidates ou à la vérification de l'adéquation d'une fonction d'influence aux cas connus.

4 Apprendre à partir d'un cas révisé

Dans cette section, nous allons illustrer un scénario précis d'apprentissage des connaissances d'adaptation qui s'appuie sur l'exploitation des $\Delta_{revise} D_j$.

L'exemple que nous présentons se situe à nouveau dans le domaine des véhicules automobiles d'occasion. Dans ce problème, il s'agit, connaissant un ensemble de caractéristiques d'un véhicule, de calculer son prix probable de vente sur le marché de l'occasion. Le prix de vente du véhicule dépend de plusieurs caractéristiques, chacune d'elle ayant une influence plus ou moins importante sur la variation du prix.

Dans cet exemple, nous considérons qu'un cas source a été remémoré dans la base de cas en utilisant une méthode de remémoration guidée par des critères d'adaptabilité. Il s'agit maintenant d'utiliser les fonctions d'influence appropriées pour déterminer une solution adaptée au problème.

Par la suite, nous allons traiter deux cas possibles. Dans le premier cas, nous considérons que le descripteur de solution (le prix) ne dépend que d'un descripteur de problème (le kilométrage). Cet exemple nous permettra d'illustrer la modification d'un descripteur de solution par une fonction d'influence et en fonction d'une variation d'un descripteur de problème. Le deuxième exemple traite de la situation où le descripteur de solution (le prix) dépend de plusieurs descripteurs de problèmes (le kilométrage, l'état général du véhicule et son année de mise en circulation). Cet exemple plus concret est plus proche de la réalité et permet de mettre en évidence certains problèmes auxquels on se trouve immédiatement confronté lorsque l'on cherche à apprendre des connaissances d'adaptation.

Dans tout processus d'apprentissage, il est nécessaire de définir explicitement la cible de l'apprentissage. Dans notre cas, ce sont les connaissances d'adaptation, et plus précisément les fonctions d'influence. Pour chaque fonction à améliorer, il faut choisir si l'on privilégie une fonction simple mais dont l'erreur d'approximation sera plutôt importante ou une fonction plus complexe à déterminer mais aussi plus précise. Une fois la cible identifiée, il convient de sélectionner le critère à optimiser. Il peut s'agir de la précision de l'approximation par rapport à un paramètre dans le cas où la fonction fait intervenir plusieurs paramètres, du temps d'exécution, de la complexité de la fonction, etc. Reste à sélectionner une méthode d'apprentissage appropriée. Ces dernières font légion dans la littérature (1). A titre d'exemple, on citera les méthodes d'interpolation ou

d'extrapolation linéaires ou polynomiales (approches par moindres carrés par exemple), les approches de types *k-means* ou encore les diverses méthodes proposées par le courant des réseaux neuronaux.

Dans le cas où une seule influence est mise en jeu, l'adaptation est effectuée en appliquant une fonction d'influence permettant de calculer une variation de prix en fonction d'une variation de kilométrage par approximation polynomiale (cf. figure 2). A l'issue du processus d'adaptation, on obtient un $\Delta_{\text{adapte}}D_j$ qui correspond à l'écart entre le descripteur de solution prix du cas source et le descripteur de solution prix du cas cible. Il est possible d'associer à cette valeur un *seuil de confiance* qui peut être déterminé en fonction du nombre de cas proches qui étaient disponibles pour effectuer l'estimation.

Ce résultat est soumis à l'utilisateur du système qui, selon toute probabilité, va le corriger durant la phase de révision. On obtient alors un $\Delta_{\text{revise}}D_j$ qui correspond à l'écart entre la valeur du descripteur après la phase d'adaptation et sa valeur après la phase de révision. On peut alors initier une phase d'apprentissage. Étant donné ce nouveau point de confiance (puisque l'on suppose que l'utilisateur est satisfait de la solution), il est possible de recalculer la fonction d'influence en prenant en compte ce nouveau point. Certaines approches d'apprentissage, comme les moindres carrés par exemple, proposent des approches récursives qui permettent de modifier une fonction sans avoir à tout recalculer. Durant cette phase, il est possible de donner à l'utilisateur un rôle plus important dans l'ajustement des fonctions d'influence. En effet, il est possible d'exploiter à nouveau la notion de seuil de confiance en lui demandant d'ordonner les solutions en fonction de la confiance qu'il leur accorde. Les solutions les plus "fiabiles" au regard de l'utilisateur auront alors plus de poids dans la détermination de la fonction d'influence.

Dans le cas où plusieurs influences sont impliquées, un descripteur de solution dépend de plusieurs descripteurs de problèmes. L'adaptation consiste donc dans un premier temps à calculer des variations locales du descripteur de solution en considérant indépendamment chacun des descripteurs de problème et chacune des fonctions d'influence associées puis, dans un deuxième temps, de combiner ces variations locales à l'aide d'un opérateur spécifique au problème posé. Une trace du raisonnement effectué par le système peut être conservée afin de pouvoir indiquer, lors de la phase de révision, dans quelle mesure chacune des fonctions d'influence a eu un impact sur la variation du descripteur de solution considéré. Si un seul descripteur de problème impliqué dans ce calcul a varié entre le cas source et le cas cible, il est facile de savoir quelle fonction d'influence est en cause dans l'erreur produite mais si plusieurs descripteurs ont varié, même en conservant une trace du raisonnement effectué, il semble impossible de savoir quelles fonctions d'influences sont en cause dans l'erreur produite. Dans ce cas, il faut donc faire des hypothèses sur les fonc-

tions d'influence qui peuvent être mises en cause et présenter ces hypothèses à l'utilisateur qui pourra tenter de guider le système. Si ce processus permet d'identifier une ou plusieurs fonctions d'influence à corriger, on peut en revenir au processus décrit précédemment.

5 Conclusion

Dans cet article, nous avons présenté un panorama des connaissances mobilisées dans le raisonnement à partir de cas qui nous a permis de caractériser le RàPC du point de vue des connaissances. Nous avons montré dans quelle mesure le RàPC en général et l'apprentissage des connaissances d'adaptation en particulier pouvaient tirer profit de l'unification des connaissances de similarité et d'adaptation. Puis, à partir d'un modèle d'adaptation centré sur la notion de dépendance, nous avons identifié des cibles d'apprentissage dans ce modèle, nous avons suggéré plusieurs scénarios d'apprentissage des connaissances identifiées précédemment et nous avons illustré l'un de ces scénarios à travers d'un exemple plus détaillé. Actuellement, une expérimentation des notions de dépendance et d'influence est en cours au travers d'une implantation dans JColibri (3), un environnement de prototypage d'applications de RàPC. Nous projetons d'expérimenter les différents scénarios d'apprentissage grâce à cet outil.

Références

- [1] Antoine Cornuéjols et Laurent Miclet. *Apprentissage artificiel - Concepts et algorithmes*. Eyrolles, 2003.
- [2] Mathieu d'Aquin, Sébastien Brachais, Jean Lieber et Amedeo Napoli. Vers une acquisition automatique de connaissances d'adaptation par examen de la base de cas — une approche fondée sur des techniques d'extraction de connaissances dans des bases de données. In *12ème Atelier de Raisonnement à Partir de Cas - RàPC'04*, pages 41–52, Université Paris Nord, Ville-taneuse, France, 2004.
- [3] Belen Diaz-Agudo, Pedro A. Gonzalez-Calero, P. P. Gomez-Martin et M. A. Gomez-Martin. On developing a distributed cbr framework through semantic web services. In *OWL : Experiences and directions*, Galway, Ireland, 11-12 november 2005.
- [4] Susan Fox et David B. Leake. Using introspective reasoning to guide index refinement in case-based reasoning. In *Sixteenth Annual Conference of the Cognitive Science Society*, pages 324–329, Atlanta, GA, 1994.
- [5] Béatrice Fuchs, Jean Lieber, Alain Mille et Amedeo Napoli. An algorithm for adaptation in case-based reasoning. In W. Horn, éditeur, *14th European Confe-*

- rence on Artificial Intelligence - ECAI'2000, pages 45–49, Berlin, Germany, 2000. IOS Press.
- [6] Kathleen Hanney et Mark T. Keane. Learning adaptation rules from a case-base. In Lecture Notes In Computer Science, éditeur, *Proceedings of the Third European Workshop on Advances in Case-Based Reasoning*, 1996.
- [7] David B. Leake. Learning adaptation strategies by introspective reasoning about memory search. In AAAI Press, éditeur, *AAAI-93 Workshop on Case-Based Reasoning*, pages 57–63, Menlo Park, CA, 1993.
- [8] David B. Leake. Becoming an expert case-based reasoner : Learning to adapt prior cases. In *Eighth Annual Florida Artificial Intelligence Research Symposium*, pages 112–116, 1995. Invited Paper.
- [9] David B. Leake, Andrew Kinley et David Wilson. Acquiring case adaptation knowledge : A hybrid approach. In AAAI Press, éditeur, *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, Menlo Park, CA, 1996.
- [10] David B. Leake, Andrew Kinley et David Wilson. Multistrategy learning to apply cases for case-based reasoning. In *Third International Workshop on Multistrategy Learning*, pages 155–164, Menlo Park, CA, 1996. AAAI Press.
- [11] David B. Leake, Andrew Kinley et David Wilson. Case-based similarity assessment : Estimating adaptability from experience. In *Fourteenth National Conference on Artificial Intelligence*, pages 674–679, Menlo Park, CA, 1997. AAAI Press.
- [12] Jean Lieber. Reformulations and adaptation decomposition. In *International Conference on Case-Based Reasoning - ICCBR'99*, Munich, Germany, 1999. LSA, University of Kaiserslautern.
- [13] Jean Lieber, Pierre Bey, Florian Boisson, Benoît Bresson, Pierre Falzon, Anne Lesur, Amedeo Napoli, Maria Rios et Catherine Sauvagnac. Acquisition et modélisation de connaissances d'adaptation, une étude pour le traitement du cancer du sein. In *Journées ingénierie des connaissances - IC'2001*, pages 409–426, Grenoble, France, 2001. Presses Universitaires de Grenoble.
- [14] Jean Lieber, Mathieu D'aquin, Sébastien Brachais et Amedeo Napoli. Une étude comparative de quelques travaux sur l'acquisition de connaissances d'adaptation pour le raisonnement à partir de cas. In *12ème Atelier de Raisonnement à Partir de Cas - RàPC'04.*, pages 53–60, Université Paris Nord, Villetaneuse, France, 2004.
- [15] Alain Mille. Modèle conceptuel du raisonnement basé sur les cas. In Isabelle Bichindaritz, éditeur, *4ème Séminaire Français sur le Raisonnement à Partir de Cas*, pages 40–52, Université René Descartes (Paris V), 1995.
- [16] Allen Newell. The knowledge level. 19(2) :87–127, octobre 1982.
- [17] Michel Py. Un modèle conceptuel de raisonnement par analogie. *Revue d'Intelligence Artificielle*, 8 :63–99, 1994.
- [18] Michael M. Richter. Classification and learning of similarity measures. In *Studies in Classification, Data Analysis and Knowledge Organisation*. Springer, 1992.
- [19] Barry Smyth. *Case-Based Design*. Doctoral thesis, Trinity College,, Dublin, Ireland, 1996.
- [20] Barry Smyth et Mark T. Keane. Remembering to forget : A competence-preserving case deletion policy for case-based reasoning systems. In *IJCAI*, pages 377–383, 1995.
- [21] Barry Smyth et Mark T. Keane. Adaptation-guided retrieval : Questioning the similarity assumption in reasoning. *Artificial Intelligence*, 102(2) :249–293, 1998.
- [22] Wolfgang Wilke, Ivo Vollrath, Klaus-Dieter Althoff et Ralph Bergmann. A framework for learning adaptation knowledge based on knowledge light approaches. In *Adaptation in Case Based Reasoning : A Workshop at ECAI 1996*, Budapest, 1996.