



Maintenance d'un système de raisonnement à partir de cas.

Mohamed-Karim Haouchine, Brigitte Chebel-Morello, Noureddine Zerhouni

► To cite this version:

Mohamed-Karim Haouchine, Brigitte Chebel-Morello, Noureddine Zerhouni. Maintenance d'un système de raisonnement à partir de cas.. International Conference on Control, Modelling and Diagnosis, ICCMD'06., May 2006, Université Badji Mokhtar, Annaba., Algérie. 6 p., 2006. <hal-00331739>

HAL Id: hal-00331739

<https://hal.archives-ouvertes.fr/hal-00331739>

Submitted on 17 Oct 2008

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.

MAINTENANCE D'UN SYSTEMES DE RAISONNEMENT A PARTIR DE CAS

Karim Mohamed Haouchine, Brigitte Chebel-Morello, et Noureddine Zerhouni

Laboratoire d'Automatique de Besançon
LAB – UMR CNRS 6596
24 rue Alain Savary – 25000 Besançon
{karim.haouchine, bmorello, zerhouni}@ens2m.fr

RESUME

La maintenance des systèmes de Raisonnement à partir de cas, intéresse un certain nombre de travaux, dont nous dressons un état de l'art. Parmi les méthodes déployées, ayant trait particulièrement à la maintenance de la base de cas, nous situons notre contribution dans la réduction de la base de cas, et plus particulièrement sur une stratégie de suppression de cas basée sur un critère : la compétence. Une mesure est proposée, inspirée des travaux existant dans la littérature et est illustrée par un premier test fait sur une base de 69 cas.

Mots-clés : raisonnement à partir de cas, sources de connaissances dans les systèmes raisonnement à partir de cas, maintenance de la base de cas, politiques et stratégies de la maintenance de la base de cas.

1. Introduction

Le Raisonnement à Partir de Cas (RàPC) est une approche de résolution de problème qui consiste à réutiliser des expériences passées afin de résoudre un nouveau problème [1]. Un cas peut être défini par une paire : (Prob, Sol(Prob)) et la solution associée au problème et est stocké dans une mémoire appelée Base de Cas (BC). La partie problème comprend les données d'observations et la situation comportementale, tandis que la partie solution contient la description de la solution apportée par le raisonnement. Un cas stocké dans la BC est nommé cas source et noté (srce, Sol(srce)) dont on va s'inspirer pour résoudre un nouveau cas qu'on appellera cas cible. Le mode de raisonnement se décompose souvent en cinq étapes principales : l'élaboration, la remémoration, l'adaptation, la validation et la mémorisation. A partir d'un cas à résoudre, la phase d'élaboration construit un cas cible (cible, ?), en complétant ou filtrant la description d'un problème à partir d'une description éventuellement incomplète. Puis, suivant une métrique de similarité, on remémore des cas sources similaires au cas cible, qui seront adapter par construction d'une solution Sol(cible), ensuite vient la validation de la solution en cas de besoin. La mémorisation consiste à

stocker le nouveau cas (cible, Sol(cible)) une fois celui-ci validé, si ce stockage est jugé opportuniste.

Tout système informatique arrivé à maturité, notamment les systèmes de raisonnement à partir de cas (RàPC) nécessitent en phase de fonctionnement d'être maintenu, pour garantir la qualité de ce système. En effet, la maintenance du système de RàPC devient nécessaire pour des systèmes qui sont conçus pour fonctionner sur de longues périodes et/ou qui seront amenés à traiter de grands volumes de données et de cas. On note que la qualité d'un système de RàPC est liée à la définition et à la représentation d'un cas, l'organisation de la base de cas, les diverses indexations utilisées et la définition de *bonnes* mesures de similarités pour la recherche des cas et le lien recherche adaptation des cas. Il existe différents travaux dans ce domaine, allant de la modélisation du cycle de RàPC soulignant les phases relatives à la maintenance, en passant par le contrôle des différentes sources de connaissances constituant le système de RàPC, jusqu'à la maintenance de la base de cas qui sera développé au troisième paragraphe. On s'intéressera à l'optimisation de la base de cas et plus particulièrement aux stratégies d'ajout et de suppression de cas ainsi que les critères associés. Puis enfin, nous présenterons une méthode de suppression de cas en proposant une mesure de compétence (MC), permettant d'optimiser la dimension de la BC.

2. LA MAINTENANCE D'UN SYSTEME DE RÀPC

Des travaux de Roth-Berghofer [14] et de Reinartz [11] ont été fait sur la modélisation du cycle de vie du système de RàPC intégrant les phases liées à la maintenance. Ainsi deux métaphases ont été proposées l'une concernant l'application et l'autre la maintenance.

D'autres travaux, comme ceux de Richter voient un système de RàPC à travers des sources ou des conteneurs de connaissances [12]. La maintenance dans cette optique consiste à développer des techniques afin de contrôler et réagir face aux changements de ces différentes sources de connaissances. Ces sources sont :

- *La source de vocabulaire* : contient toutes les informations sur les définitions et les structures utilisées

- *La source des mesures de similarité* : contient les mesures nécessaires pour la recherche des cas
- *La source d'adaptation* : contient les règles de transformation de la solution
- *La source de la base de cas* : représente le contenu et l'organisation de la BC

Les trois premières sources de connaissances, sont élaborées avant que le système tourne, alors que la BC est remise à jour normalement dynamiquement. Selon Richter, chaque source peut porter presque toute la connaissance disponible, et les manipulations sur une source ont de petites répercussions sur les autres. A chaque source de connaissance on peut associer une maintenance. Thomas Gabel a travaillé sur l'apprentissage des mesures de similarité; comme l'évaluation de poids associés aux descripteurs d'un cas ou l'acquisition de mesures de similarité grâce à des fonctions d'utilité issues d'une boucle de retour sur le traitement associées à des applications de RàPC [2]. La BC a un rôle central ce qui explique que la majorité des travaux faits dans ce domaine, est essentiellement fondé sur la **Maintenance de la Base de Cas (MBC)** [4]. De plus les connaissances d'un système de RàPC sont liées aux cas puisque ceux-ci sont affectés par tout changement dans les sources de connaissances. La BC est la source de connaissance la plus sensible aux changements dans le système de RàPC, sa consultation est la plus appropriée pour déclencher les opérations de maintenance.

3. LA MAINTENANCE DE LA BASE DE CAS

La MBC est la mise en œuvre des politiques permettant de réviser l'organisation et/ou le contenu (représentation, domaine d'application, contenu d'informations, ou l'implémentation) de la BC afin d'améliorer le raisonnement futur [4]. La MBC est un ensemble de réalités différentes, telle que l'élimination de la redondance de cas, de la suppression des cas incohérents, de la sélection de groupes de cas permettant d'éliminer la redondance et d'améliorer le pouvoir de raisonnement du système, les cas peuvent être réécrit afin de réparer les problèmes d'incohérences [18].

3.1. Politiques de la MBC

On peut diviser les approches proposées pour la MBC en deux politiques, une concernant l'optimisation et l'autre le partitionnement de la BC (voir *figure1*).

L'objectif visé par ses approches est de réduire le temps de recherche des cas soit dans l'optimisation en supprimant les cas les moins pertinents, suivant deux stratégies l'ajout et la suppression de cas, soit dans le partitionnement, en partitionnant la BC en plusieurs espaces de recherche permettant ainsi de sélectionner de manière incrémentale les attributs qui sont riches en information et qui peuvent couvrir la structure de la BC [24]. On compte des travaux sur les réseaux de neurones, soient dynamiques [6] ou statiques [13].

Plusieurs propriétés des cas de la BC ont été proposées pour effectuer une évaluation concernant la BC.

3.2. Critères d'évaluation de la qualité de la BC

Une BC est de bonne qualité si elle permet au système de RàPC de résoudre le plus grand nombre de problèmes possibles de manière correcte en un temps raisonnable. Plusieurs critères permettant d'évaluer la qualité de la BC ont été proposés dans la littérature, on cite ceux qui nous intéressent concernant notre travail [10], [14] et [16] :

- **Compétence**, est mesurée par le nombre de problèmes différents pour lesquels le système apporte une bonne solution.
- **Performance**, d'un système est mesurée par le temps de réponse qui lui est nécessaire pour proposer une solution à un cas cible. Cette mesure est liée directement aux coûts d'adaptation et aux coûts de recherche.
- **Recouvrement** d'un cas de la BC représente l'ensemble des cas cibles que ce cas peut résoudre.
- **Atteignabilité**, d'un cas cible est l'ensemble de cas sources qui peuvent être utilisés pour le résoudre.

Après avoir donné les définitions des différents critères qui permettent d'évaluer une BC, nous allons voir comment certaines d'eux sont utilisés et évalués dans les différentes stratégies de la MBC. Dans ce qui suit, nous allons développer les deux stratégies, celle de l'ajout et de la suppression de cas.

4. STRATEGIE D'AJOUT DE CAS

A partir d'une BC vierge par l'ajout successif de cas, maximisant un critère on construit une BC réduite. Il existe deux méthodes, l'une maximisant le critère de compétence l'autre celui de performance.

4.1. Méthode maximisant le critère de compétence

Smyth et McKenna ont proposé de construire une BC en utilisant un algorithme hybride, celui des plus proches voisins (CNN) en l'associant à une mesure qui sera appliquée pour chaque cas dans la BC originale [21]. Cette mesure est la valeur du recouvrement relatif RC (Relative Coverage). Elle détermine la valeur de la contribution d'un cas à la compétence de la BC. Par conséquent, les cas sont rangés selon leurs valeurs de « RC » dans la BC complète. Ensuite l'algorithme CNN est appliqué pour construire une nouvelle BC réduite. Ceci permet de sélectionner les cas ayant une grande contribution au recouvrement de la BC. Q. Yang et J. Zhu déterminent le recouvrement par une mesure de similarité et des coûts d'adaptation [25]. Dans ce cas, le recouvrement d'un cas est considéré comme le voisinage du cas dans certaines limites d'adaptabilité.

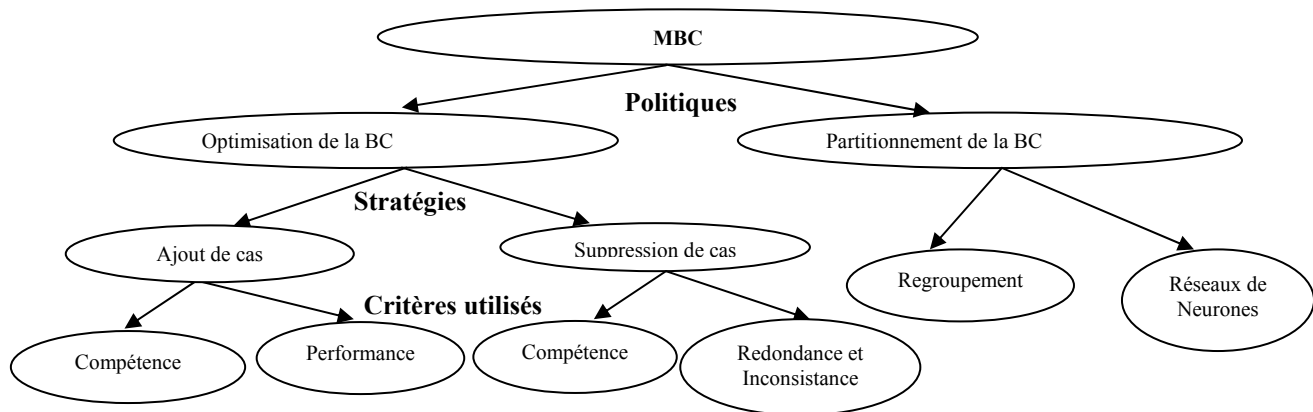


Figure 1. Schéma des différentes stratégies et critères utilisés dans la MBC

4.2. Méthode maximisant le critère de performance

Leake et Wilson ont définis une mesure de performance relative RP (Relative Performance) inspirée des travaux de la mesure « RC » qui est basée sur le coût de l'adaptation de chaque cas de la BC [5]. La valeur de la performance relative d'un cas « RP(c) » reflète sa contribution à la performance d'adaptation comparée à celle des autres cas. Avant d'ajouter un cas dans la BC réduite, sa contribution à la performance d'adaptation est estimée. Un ensemble de mesures incluant la mesure du bénéfice de performance produit par l'ajout de chaque cas est estimée. De ce fait, cette mesure utilisée pour guider l'ajout de cas en favorisant les cas ayant une grande valeur de « RP ». Dans le même cadre une autre mesure a été développée concernant le bénéfice de la performance relative PB (relative Performance Benefit) qui est basée sur les mêmes notions que la mesure « RP », qui donne les mêmes performances, c'est à dire le même taux de réduction de la taille de la BC [6]. Cependant, la méthode RC-CNN fournit un taux de réduction de la taille de la BC supérieure par rapport à celui des méthodes RP-CNN et PB-CNN. Par contre, ces deux dernières donnent un résultat concernant le coût d'adaptation des cas dans la BC obtenue nettement meilleur par rapport à RC-CNN.

5. STRATEGIE DE SUPPRESSION DE CAS

A partir d'une BC, cette stratégie évalue les cas suivant un critère afin de pouvoir les supprimer et ramener la BC à un nombre de cas donné. Les critères d'évaluation tels que la compétence, la redondance et l'inconsistance ont été utilisés dans différentes méthodes.

5.1. Méthode de suppression par balayage de la base de cas

Dans ce cas, la BC est balayée entièrement dès que sa taille atteint un seuil prédéfini, il y a suppression de cas.

- *la suppression aléatoire*, supprime aléatoirement un cas de la BC [8]. Cette méthode sera un point de référence pour la comparaison avec les autres méthodes.
- *Ironically*, qui s'apparenterait à un facteur d'oubli de la connaissance la moins utilisée dépend de la valeur

minimum de la *fréquence de remémoration* de chaque cas dans la BC [9].

- *Suppression en se basant sur le problème d'utilité* (Utility Deletion : UD), cette stratégie supprime les cas qui ont une valeur de l'utilité négative [9]. *Le problème d'utilité* est défini par la différence entre la taille de la BC, la qualité de la solution et le problème d'efficacité associée aux grandes BC. *L'efficacité* du système est mesurée par le temps moyen mis pour résoudre un problème cible. La réduction du temps de la solution correspond à l'augmentation de l'efficacité. *La qualité* de la solution est lié au pourcentage de bonnes réponses fournit par le système. Elle augmente avec la taille de la BC [17].
- *Suppression basée sur la redondance et l'inconsistance*, cette méthode est dotée de deux modules de détection l'un de *redondance* et l'autre d'*inconsistance*. Suivant une série de test de chaque cas de la BC par ces deux modules, ceux-ci sont supprimés ou non de la BC avec l'approbation de l'utilisateur [10].
- *Suppression basée sur la taille et la densité de la BC*, cette stratégie de suppression a été proposée par B. Smyth et M.T. Keane et étudie la taille de la BC, la densité et la distribution des cas dans la BC. Elle essaye de garder l'homogénéité de la densité des cas dans la BC [19].

5.2. Méthode à partir de catégorisation des cas

Ces méthodes font appel à une modélisation de la compétence de la BC qui ont été proposés par B. Smyth et M.T. Keane [16], [19], [20], [22], [23]. Une catégorisation de cas dans la BC est créée selon leurs compétences. Cette catégorisation prend appui sur deux notions importantes qui sont *le recouvrement (coverage)* et *l'atteignabilité (reachability)*.

Soit une base de cas $C = \{c_1, \dots, c_n\}$ et $c \in C$ l'ensemble des cas cibles dans la BC. Formellement :

- Pour $c \in C$, $\text{Recouvrement}(c) = \{c' \in C : \text{résout}(c, c')\}$
- Pour $c \in C$, $\text{Atteignabilité}(c) = \{c' \in C : \text{résout}(c', c)\}$

Ainsi, 4 classes de cas sont considérées :

- *Pivot* : $\text{Pivot}(c) \text{ ssi } \text{Atteignabilité}(c) - \{c\} = \emptyset$

- *Support* : $\text{Support}(c) \text{ ssi } \exists c' \in \text{Atteignabilité}(c) - \{c\} : \text{Recouvrement}(c') \subset \text{Couverture}(c)$
- *Couverture* : $\text{Couverture}(c) \text{ ssi } \text{Pivot}(c) \wedge \text{Recouvrement}(c) \cap \bigcup_{c' \in \text{Atteignabilité}(c) - \{c\}} \text{Couverture}(c') \neq \emptyset$
- *Auxiliaire* : $\text{Auxiliaire}(c) \text{ ssi } \exists c' \in \text{Atteignabilité}(c) - \{c\} : \text{Recouvrement}(c) \not\subset \text{Recouvrement}(c')$

Les méthodes développées dans ce cadre génère un ensemble de cas cibles afin de catégoriser la BC. Deux hypothèses de base sous tendent ces modèles à savoir la BC correspond à un échantillon de cas cibles ou cas potentiels et l'espace de problèmes est régulier, ce qui veut dire que les problèmes similaires ont des solutions similaires.

- *Suppression de traces* (footprint deletion : FD), cette stratégie de suppression de cas vise à maximiser la compétence en minimisant la taille de le BC en la guidant vers une configuration optimale. Elle s'appuie sur la catégorisation des cas de la BC pour pouvoir supprimer tous les cas dit auxiliaires, de couverture et de ne laisser qu'un cas de support dans chaque groupe de support et de garder tous les cas pivots. Cette stratégie préserve la compétence mais ne s'occupe pas du problème de la performance.
- *Suppression trace-utilité* (footprint-utility deletion : FUD), Cette stratégie améliore la précédente en tenant compte de la performance des différents cas quand il faut en garder un parmi d'autres à éliminer. Elle s'occupe conjointement de la compétence et de la performance [15].

6. PROPOSITION DE METHODE DE MODELISATION DE LA COMPETENCE DE LA BASE DE CAS

6.1. Proposition d'une stratégie de suppression de cas basée sur la compétence

Nous proposons d'exploiter dans une stratégie de suppression de cas, le critère de compétence et de proposer une métrique inspirée de celles développées dans la stratégie d'ajout de cas, celle du RC pour la compétence et celle du RP et du PB pour la performance. La méthode proposée est une combinaison de la méthode de Smyth et Mckenna et de la mesure du recouvrement relatif RC [20].

6.2. La mesure de la compétence (MC) pour la sélection des cas

Lorsqu'on parle de la compétence d'un cas, on se réfère à son pouvoir de résolution sur un certain nombre de problèmes cibles.

On considère une BC « C » contenant un certain nombre de cas « c_i » : $C = \{c_i \mid i \in (1, \dots, n)\}$

Lorsqu'on veut calculer la compétence d'un cas « c_k » dans la BC, on considère l'espace des cas cibles tous les cas appartenant à la BC y compris le cas « c_k » en question.

Notre mesure (MC) se base essentiellement sur les deux notions de *recouvrement* et d'*atteignabilité*.

On attribut à chaque recouvrement et atteignabilité d'un cas une valeur qu'on notera *valeur recouvrement* « Vr » et *valeur atteignabilité* « Va ».

On définit une mesure (MC) devant estimer la contribution individuelle de la compétence d'un cas « c_k » en fonction de la taille de l'ensemble du recouvrement de ce dernier.

$$\text{« } \text{MesureCompétence}(c) = \frac{Vr(c)}{Va(c)} \text{ »}$$

Cette mesure est utilisée pour guider la suppression des cas dans la BC « Favoriser les cas qui possède une grande valeur de 'MC' et supprimer ceux qui ont une petite valeur »

6.3. Méthode de sélection et de suppression des cas

Cette méthode suit le principe des stratégies de suppression de cas, c'est à dire en prenant une BC complète telle qu'elle est, ensuite on procède à la suppression selon la valeur de « MC » conformément aux étapes suivantes :

Soit une BC contenant des cas ainsi que l'ensemble de cas cibles définit précédemment.

- Calcul de Vr(c_i), Va(c_i) et MC(c_i).
- Créer les groupes de recouvrement et d'atteignabilité
- Suppression des cas de support, cas intra-auxiliaires et inter-auxiliaires

Les cas intra-auxiliaires sont les cas auxiliaires dans une classe donnée, et les cas inter-auxiliaires sont les cas auxiliaires qui se chevauchent entre plusieurs classes.

Le détail de l'étape « a » est décrit ci-dessous :

Algorithme de calcul de la valeur du recouvrement des cas de la BC

Pour tout cas_i ∈ BC

Pour tout cas_j ∈ BC (avec j≠i)

Si cas_i apporte une solution à cas_j

Augmenter valeur Vr_i

Mémoriser l'index du cas cas_j

Fin si

Fin Pour

cas_i ← (Vr_i, index cas_j, classe)

Fin Pour

Par analogie, on applique ce même principe pour calculer la valeur de Va(c_i).

De ce fait, lorsqu'on applique les deux algorithmes sur l'ensemble des cas dans la BC, on obtiendra pour chaque cas sa valeur Vr(c), Va(c) ainsi que les différents groupes de recouvrement et atteignabilité. A partir de ces deux valeurs, on peut caractériser les cas et garder les cas pivots.

Propriétés

- Quand Va(c_i) = 1 et MC(c_i) = 1 → cas pivot (selon Smyth et Mckenna)
- Quand Va(c_i) > 1 et MC(c_i) = 1 → suppression du cas (cas auxiliaire)

- Quand on a un groupe de cas où il y'a que des cas qui ont les mêmes $Vr(c_i)$ et $Va(c_i) > 1$, donc les même MC → suppression de tous les cas de ce groupe en ne laissant qu'un seul cas (cas de supports)
- Le critère d'arrêt est lorsque la valeur de $CompétenceGlobale(BC)=1$ et que chaque cas recouvre seulement lui même parmi les cas présent dans sa classe.

Objectif

La mesure proposée chiffre le taux de recouvrement et d'atteignabilité des cas et par un jeu de test permet une catégorisation de cas dans la BC.

Cette mesure est destinée à maximiser la compétence de la BC donc son recouvrement et à minimiser l'atteignabilité des cas dans la BC, avec un critère d'arrêt tenant compte de la compétence globale de la BC et non pas d'un seuil fixé arbitrairement (comme le nombre de cas de la BC).

Exemple

Soit une BC contenant quatre cas $C=\{c_1, c_2, c_3, c_4\}$. La figure 2 montre l'espace de recouvrement et d'atteignabilité de ces cas.

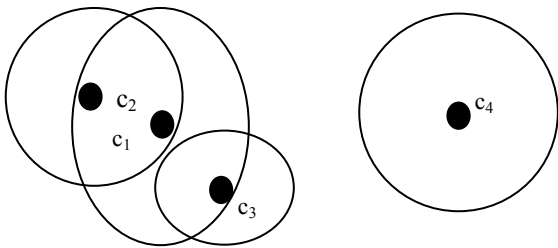


Figure 2. Une BC contenant quatre cas $\{c_1, c_2, c_3, c_4\}$ avec leur espace de recouvrement et d'atteignabilité.

Tout d'abord, on va calculer les valeurs de recouvrement « Vr » et d'atteignabilité « Va » de chaque cas.

- Recouvrement(c_1)= $\{c_1, c_2, c_3\}$ → $Vr(c_1)=3$
- Atteignabilité(c_1)= $\{c_1, c_2\}$ → $Va(c_1)=2$
- Recouvrement(c_2)= $\{c_1, c_2\}$ → $Vr(c_2)=2$
- Atteignabilité(c_2)= $\{c_1, c_2\}$ → $Va(c_2)=2$
- Recouvrement(c_3)= $\{c_3\}$ → $Vr(c_3)=1$
- Atteignabilité(c_3)= $\{c_1, c_3\}$ → $Va(c_3)=2$
- Recouvrement(c_4)= $\{c_4\}$ → $Vr(c_4)=1$
- Atteignabilité(c_4)= $\{c_4\}$ → $Va(c_4)=1$

Ensuite, on déduit les valeurs de MC pour chaque cas.

$MC(c_1) = 1.5, MC(c_2) = 1, MC(c_3) = 0.5, MC(c_4) = 1.$

- On constate que $MC(c_2) = MC(c_4) = 1$. Ce sont deux cas complètement différents. Le cas c_4 est un cas pivot, on supprime donc le cas c_2 .

- On constate aussi que $MC(c_1) > MC(c_3)$ → suppression du cas c_3 .

On obtient donc la suppression de deux cas (c_2 et c_3) et une BC contenant les cas (c_1 et c_4).

En recalculant la valeur de MC pour chaque cas on trouve $MC(c_1) = MC(c_4) = 1$ avec $Va(c_1) = Va(c_4) = 1$ et $Vr(c_1) = Vr(c_4) = 1$.

Ainsi on obtiendra une BC réduite possédant le moins de cas possible et un recouvrement maximal.

La valeur de la compétence globale de la BC réduite est égale à la somme des MC des cas dans cette BC divisée par le nombre de cas présents.

$$\text{« Compétence Globale (BC) = } \frac{\sum_{i=1}^n MC(c_i)}{\sum_{i=1}^n c_i} \text{ »}$$

Application

On a appliqué notre méthode sur une BC contenant 69 cas. Cette BC est dédiée pour une plate forme d'e-maintenance concernant le transfert flexible SORMEL [3]. Les résultats obtenus sont illustrés dans le tableau suivant :

Groupes de support	Cas de supports	Cas auxiliaires
19	49	4

Tableau 1. Statistique de la Base de Cas

Concernant les cas auxiliaires il a été recensé deux cas inter-auxiliaires et deux cas intra-auxiliaires.

Après la suppression des 4 cas auxiliaires et des cas de supports en laissant seulement un cas de support dans chaque groupe de support (donc suppression de 30 cas de supports), on obtient une BC réduite contenant 35 cas. Ce qui nous donne un taux de réduction de 49,27%.

7. CONCLUSIONS

Dans les recherches faites dans la communauté de RàPC, Une majorité des travaux est consacrée à la MBC. Ces travaux ont comme objectif d'assurer la compétence, la performance et le maintien dans le temps de la qualité des systèmes de RàPC. Parmi les politiques et stratégies de MBC existantes, nous proposons une stratégie de suppression de cas en lien avec la méthode de Smith. Nous prenons appui sur la caractérisation faite des différents cas dans la base de cas, et nous retrouvons celle-ci par une méthode basée sur un critère de compétence. A partir de cette catégorisation, la suppression de cas devient évidente. La notion de compétence sera quantifiée par une mesure de maintenance MC, qui est établie sur des notions de recouvrement et d'atteignabilité associé à une mesure du recouvrement relatif RC. L'intérêt de cette mesure est de quantifié aussi bien la compétence et la performance d'une base de cas. A la suite de ce travail, une étude comparative avec les autres mesures existantes s'impose, sur des bases de cas existantes, et d'une manière plus général sur des systèmes complets de RàPC.

8. REFERENCES

- [1] A. Aamodt et E. Plaza, "Case-based reasoning: Foundational issues, methodological variation, and systems", *AI Communication*, 7(1) : 36-59, 1994.
- [2] T. Gabel, "On the Use of Vocabulary Knowledge for Learning Similarity Measures", 2004.

- [3] M.K. Haouchine, "Maintenance d'une base de cas dédiée au diagnostic et à sa réparation", Mémoire de Master Recherche, 12 Juillet 2005.
- [4] D.B. Leake et D.C. Wilson, "Categorizing case-base maintenance: dimensions and directions", *Advances in Case-Based Reasoning, 4th European Workshop, EWCBR 98*, Proceedings, Springer-Verlag, Berlin, 1998: 196-207, 1998.
- [5] D.B. Leake et D.C. Wilson, "Remembering Why To Remember: Performance-guided case-base maintenance", *Advances in Case-Based Reasoning: Proceeding of EWCBR-2K*, Springer-Verlag, 2000a.
- [6] D.B. Leake et D.C. Wilson, "Guiding Case-Base Maintenance: Competence and Performance", *Online Proceedings of the ECAI'2000 Workshop on Flexible Strategies for Maintaining Knowledge Containers*, 2000b.
- [7] M. Malek, "Hybrid approaches for integrating neural networks and case based reasoning: From loosely coupled to tightly coupled models", *In Soft Computing in Case Based Reasoning*, éditeurs Tharam S. Dillon Sankar K. Pal et Daniel S. Yeung, pages 73-94, 2000.
- [8] S. Markovitch et P.D. Scott, "The Role of Forgetting in Learning", *In Proceedings of the Fifth International Conference on Machine Learning*, pages 459-465, 1988.
- [9] S. Minton, "Qualitative Results Concerning the Utility of Explanation-Based Learning", *Artificial Intelligence*, 42: 363-391, 1990.
- [10] K. Racine et Q. Yang, "On the consistency Management of Large Case Bases: the Case for Validation", *To appear in AAAI Technical Report, Verification and Validation Workshop*, 1996.
- [11] T. Reinartz, I. Iglezakakis et T. Roth-Berghofer, "Review and restore for Case Base Maintenance", *Computational Intelligence Journal*, 17(2), 2001.
- [12] M.M. Richter, "Introduction", *In Case-Based Reasoning Technology: From Foundations to Applications*, Edited by M. Lena, B. Bartsc-Sporl, H. D. Burkhard, et S. Wess. Springer-Verlag, Berlin, pp. 1-15, 1998.
- [13] T.H. Roh, K.J. Oh, et I. Han, "The collaborative filtering recommendation based on som clustering-indexing cbr", *In Expert Systems with Applications*, Elsevier Science, volume 25, pages 413-423, 2003.
- [14] T. Roth-Berghofer et I. Iglezzakis, "Six Steps in Case-Based Reasoning: Towards a maintenance methodology for case-based reasoning systems", *Includes Proceedings of the 9th German Workshop on CBR, GWCBR*, Germany, 2001.
- [15] C.K.S. Simon et D.S. Yeung, "Transferring case knowledge to adaptation knowledge: an approach for case-base maintenance", *Computational Intelligent*, Volume 17, Number 2, 2001.
- [16] B. Smyth et M.T. Keane, "Remembering To Forget: A competence Preserving Deletion Policy for Case-Based Reasoning Systems", *In: Proceeding of the 14th International Joint Conference on Artificial Intelligent*, Morgan-Kaufmann. 377-382, 1995.
- [17] B. Smyth et Pádraig Cunningham, "The Utility Problem Analysed: A Case-Based Reasoning Perspective", *Third European Workshop on Case-Based Reasoning*, Lausanne, Switzerland, 1996.
- [18] B. Smyth, Case-base maintenance, "Tasks and Methods in Applied Artificial Intelligence", *11th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, IEA-98-AIE. Proceedings. Springer-Verlag, Berlin, Germany, pages 507-516 vol.2, 1998a.
- [19] B. Smyth et E. McKenna, "Modelling the Competence of Case-Bases", *Advances in case-based reasoning, Lecture notes in computer science*, Dublin, 1488: 208-220, 1998b.
- [20] B. Smyth et E. McKenna, "Footprint-Based Retrieval", *In Proceedings of the Third International Conference on Case-Based Reasoning, ICCBR '99*, Springer Verlag, Berlin, Germany, 27-30 July 1999a
- [21] B. Smyth et E. McKenna, Building Compact Competent Case-Bases, Case-based reasoning research and development, See on Monastery, Lecture notes in computer science, 1650: 329-342, 27-30 July 1999b.
- [22] B. Smyth et E. McKenna, "Competence guided incremental foot-print", *Journal of Knowledge-Based Systems*, In press, 2002a.
- [23] B. Smyth et E. McKenna, "Competence models and the maintenance problem", *Computational Intelligence: Special Issue on Maintaining Case-Based Reasoning Systems*, In Press, 2002b.
- [24] Q. Yang et J. Wu, "Keep it simple: A case-base maintenance policy based on clustering and information theory", *13th Biennial Conference of the Canadian Society for Computational Studies of Intelligence, AI 2000: advances in artificial intelligence*, Proceedings Lecture Notes in Artificial Intelligence, Montréal PQ, Springer-Verlag, Berlin, Vol.1822. 2000: 102-114, 14-17 May 2000.
- [25] Q. Yang et J. Zhu, "A case addition policy for case-base maintenance", *Computational Intelligence Journal, A Special Issue on Case-Base Maintenance*, Boston MA UK, Blackwell Publishers, 17(2) : 250-262, 2001.