



HAL
open science

Evaluation automatique de connaissances par la théorie des fonctions de croyance

Patrick Taillandier, Catherine Dominguès

► **To cite this version:**

Patrick Taillandier, Catherine Dominguès. Evaluation automatique de connaissances par la théorie des fonctions de croyance. Rencontres des jeunes chercheurs en intelligence artificielle, 2009, Hammamet, France. pp.209-226. hal-00691257

HAL Id: hal-00691257

<https://hal.science/hal-00691257>

Submitted on 25 Apr 2012

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.

Evaluation automatique de connaissances par la théorie des fonctions de croyance

Patrick Taillandier^{1,2} et Catherine Domingues²

¹Laboratoire MSI, IFI,
ngo 42 Ta Quang Buu, Hai BaTrung, Ha noi, Viet nam
patrick.taillandier@gmail.com

²Laboratoire COGIT, IGN,
2/4 avenue Pasteur, St Mandé, France
catherine.domingues@ign.fr

Résumé : Les humains comme les systèmes artificiels utilisent fréquemment une approche par essais/erreurs pour résoudre des problèmes. Ce type d'approche implique généralement de disposer de connaissances fiables en vue de guider efficacement la recherche d'une solution optimale. Malheureusement, ces connaissances sont rarement parfaites. De plus, il s'avère souvent difficile pour les systèmes artificiels comme pour les humains d'évaluer leurs propres connaissances. Notre objectif est de proposer une approche d'évaluation automatique de la qualité des connaissances utilisées pour la recherche d'une solution optimale. Nous nous intéresserons dans ce cadre à un type spécifique d'approche de résolution de problèmes par essais/erreurs : l'exploration informée d'arbres d'états. L'approche que nous proposons consiste à analyser les traces du système et à utiliser la théorie des fonctions de croyance pour évaluer la qualité globale des connaissances. Nous présentons, au travers d'une expérimentation, une application réelle de notre approche dans le domaine de la généralisation cartographique. Cette expérimentation montre des résultats encourageants pour notre approche.

Mots-clés : Evaluation des connaissances, Théorie des fonctions de croyance, Exploration informée d'arbres d'états, Généralisation cartographique.

1 Introduction

Une stratégie classique de résolution de problème consiste à utiliser une approche par essais/erreurs. Ce type d'approche se révèle souvent très efficace. Néanmoins, dans le cadre de problèmes admettant un espace de recherche très important, le recours à des connaissances de contrôle pertinentes devient nécessaire afin de limiter le nombre d'essais pour trouver une solution optimale. Malheureusement, il est rare de disposer de connaissances parfaites. Ce problème est encore plus vrai dans le contexte de systèmes artificiels parce qu'il est difficile de transférer les connaissances d'experts humains dans un formalisme utilisable par une machine. Edward Feigenbaum a formulé ce problème en 1977 comme celui du « goulot d'étranglement de l'acquisition de connaissances » (Feigenbaum, 1977). S'il est difficile d'acquérir des connaissances parfaites, il l'est également d'auto-évaluer ses propres connaissances. En effet, donner

un diagnostic global de ses propres connaissances quand différents types de connaissances sont en jeu est généralement complexe. Or ce type de diagnostic peut s'avérer extrêmement utile pour le maintien de bases de connaissances. En particulier, pour déterminer quand réviser les connaissances et quelles parts des connaissances réviser. Nous nous intéressons dans cet article au problème de l'auto-évaluation de connaissances de contrôle dans le cadre de systèmes fonctionnant par essais/erreurs. Nous proposons pour faire face à ce problème une approche fondée sur l'analyse de traces et sur l'utilisation de la théorie des fonctions de croyance.

En partie 2, nous introduisons le contexte général de notre travail ainsi que les difficultés posées par ce problème d'évaluation. La partie 3 est dédiée à la présentation de notre approche. Nous décrivons en partie 4 une application réelle de notre approche dans le cadre du domaine de la généralisation cartographique. Enfin, la partie 5 conclut et présente les perspectives.

2 Contexte

2.1 Description du type de problèmes d'optimisation considéré

Nous nous intéressons dans cet article à un type spécifique de problèmes d'optimisation qui consiste à trouver, par application d'actions, l'état d'une entité qui maximise une fonction d'évaluation.

Soit P , un problème d'optimisation caractérisé par :

- Une entité Ent se trouvant dans un état donné.
- $\{action\}_P$: un ensemble d'actions qui peuvent être appliquées sur Ent . L'application d'une action modifie l'état de Ent . Le résultat de l'application d'une action est supposé non prévisible.
- Q_P : une fonction d'évaluation de la qualité de l'état dans lequel se trouve Ent . Q_P renvoie, pour tout état de Ent , une valeur définie sur l'ensemble des réelles.

Une instance de P définit l'état initial dans lequel se trouve Ent . Résoudre p signifie trouver, à partir de l'état initial de Ent et par application d'actions de $\{action\}_P$, l'état de Ent qui optimise Q_P .

Considérons l'exemple suivant : Soit $Probot$ un problème d'optimisation dans lequel un robot, considérant son état initial dans un labyrinthe, cherche la sortie.

- Ent_{Probot} : un robot caractérisé par sa position initiale dans le labyrinthe.
- $\{action\}_{Probot} : \{avancer, tourner \text{ à droite}, tourner \text{ à gauche}\}$
- Q_P : distance séparant le robot de la sortie du labyrinthe.

Une instance de $Probot$ est caractérisée par la position initiale du robot. Résoudre une instance de $Probot$ signifie trouver, par application d'action de déplacement, la sortie ou au moins la position la plus proche de la sortie.

Il existe de nombreuses approches pour résoudre un tel problème d'optimisation. Dans cet article, nous nous intéresserons aux systèmes résolvant ce type de problème d'optimisation par un type spécifique d'approche par essais/erreurs : l'exploration informée d'arbres d'états. Cette approche consiste à chercher l'état optimal d'une entité par exploration d'un arbre d'états, la transition entre deux états correspondant à l'application d'une action. L'aspect « informé » provient de l'utilisation de connaissances de contrôle (par exemple, le choix de l'action appliquer) pour guider l'exploration de l'arbre d'états. De part son efficacité, ce type d'approche est couramment employée pour résoudre des problèmes réels.

2.2 Description du système de résolution de problèmes considéré

Nous présentons dans cette partie, le système générique pour lequel est dédiée notre approche d'évaluation des connaissances. Ce système est fondé sur une exploration en profondeur de type "meilleur d'abord" d'un arbre d'états. Le passage d'un état à un autre correspond à l'application d'une action. La figure 1 présente un exemple d'arbre d'états.

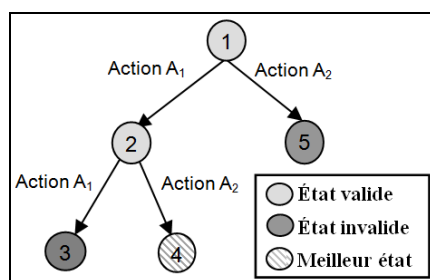


Fig. 1 – Exemple d'arbre d'états

De façon à construire un tel arbre d'états, le système s'applique un cycle d'actions. La figure 2 présente un cycle d'actions classique.

Le principe de ce cycle d'actions est le suivant : le système caractérise et évalue, dans une première étape, l'état de l'entité traité en utilisant la fonction Q_p . Il teste ensuite le critère de fin de cycle. Si ce critère est vérifié, le cycle d'actions s'arrête, dans le cas contraire, le système passe à l'étape suivante. Le système teste ensuite la validité de l'état de l'entité. Un état valide représente un état à partir duquel il est a priori intéressant de continuer l'exploration. Ainsi, si l'état est valide, le système, afin de continuer l'exploration à partir de cet état, crée une liste d'actions pouvant être appliquées pour cet état de l'entité ; dans le cas contraire où l'état est invalide et où il est a priori peu intéressant de continuer l'exploration à partir de cet état, le système fait revenir l'entité à son état précédent. Le système vérifie ensuite s'il reste des actions dans la liste d'actions. Si ce n'est pas le cas, le système fait revenir l'entité à son état précédent. Si l'entité revient à son état initial (état que l'entité avait en début de cycle d'actions) et qu'il n'y a plus d'action à appliquer pour cet état, le cycle

d'actions s'arrête. Dans le cas où la liste d'actions n'est pas vide, le système choisit, selon une heuristique, la meilleure action à appliquer, l'applique sur l'entité et retire l'action de la liste d'actions, puis revient à la première étape du cycle d'actions.

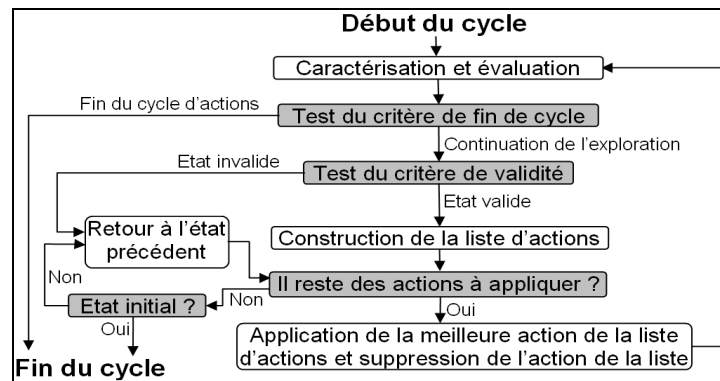


Fig. 2 – Cycle d'actions

Ce système générique utilise trois types de connaissances de contrôle :

- *Connaissances d'application des actions* qui déterminent, pour un état donné de l'entité, les actions à appliquer et dans quel ordre les appliquer.
- *Critère de validité* qui détermine, en fonction de l'ensemble des états parcourus, si l'état courant est valide ou non, c'est-à-dire s'il est a priori intéressant de continuer l'exploration à partir de cet état là.
- *Critère de fin de cycle* qui détermine, en fonction de l'ensemble des états visités, si le système doit continuer ou non l'exploration.

Les performances d'un tel système sont directement dépendantes de la qualité de ses connaissances. La qualité des connaissances peut être exprimée en termes d'efficacité et d'efficacités. L'efficacité concerne la qualité du résultat obtenu par le système, c'est-à-dire, la valeur Q_p du meilleur état trouvé. L'efficacités concerne l'aspect temps de calcul de la résolution des instances, c'est-à-dire la vitesse d'exploration de l'arbre d'états. Un bon jeu de connaissances permet donc de trouver un état de bonne qualité en parcourant peu d'états.

2.3 Difficultés de l'évaluation de la qualité des connaissances

Comme nous l'avons défini dans (Taillandier, 2008), diagnostiquer la qualité des connaissances de contrôle d'un système implique de faire face à trois types de difficultés.

La première concerne les interdépendances pouvant exister entre connaissances : il n'est parfois pas possible de déterminer si une connaissance est réellement déficiente ou si le problème provient d'une autre connaissance qui influence les résultats obtenus par la première connaissance.

La seconde difficulté concerne le type d'information pouvant être extrait de l'analyse des arbres d'états. Par exemple, s'il est possible d'extraire des informations concernant les erreurs de faux positifs pour le critère de validité (cas d'un état qui n'a mené qu'à des états non intéressants et qui aurait donc dû être invalide), il n'en est pas de même pour les erreurs de faux négatifs (cas d'un état invalide qui aurait dû être valide). En effet, on ne peut pas savoir, par une simple évaluation de l'arbre d'états, s'il aurait été possible de trouver un meilleur état si cet état avait été valide.

La dernière difficulté concerne les instances du problème utilisées pour établir le diagnostic. L'échantillon d'instances peut s'avérer ne pas être représentatif de l'ensemble des instances du problème et donc non fiable pour établir une évaluation.

3 Approche proposée

3.1 Approche générale

Notre objectif est d'évaluer automatiquement la qualité des connaissances de contrôle utilisées par un système fondé sur l'exploration informée d'arbres d'états. Nous proposons d'appliquer la même approche que celle que nous avons proposée dans (Taillandier, 2008). Cette approche est fondée sur l'analyse des traces d'exécution et sur l'utilisation d'une méthode de décision multicritère (figure 3).

A chaque fois qu'une instance de problème est résolue, le module de diagnostic analyse, durant une *phase d'évaluation*, les succès et les échecs rencontrés par chaque connaissance. Il vérifie ensuite si le nombre d'instances du problème résolues depuis le dernier diagnostic (*Nb_instances*) est suffisant pour établir un nouveau diagnostic. Si ce nombre n'est pas suffisamment élevé, le système met fin au processus de diagnostic et d'autres instances peuvent être traitées. S'il est suffisamment élevé, le module déclenche une phase de diagnostic qui consiste à évaluer séparément chaque connaissance, puis à utiliser la théorie des fonctions de croyance afin de fournir une évaluation qualitative globale de l'ensemble des connaissances.

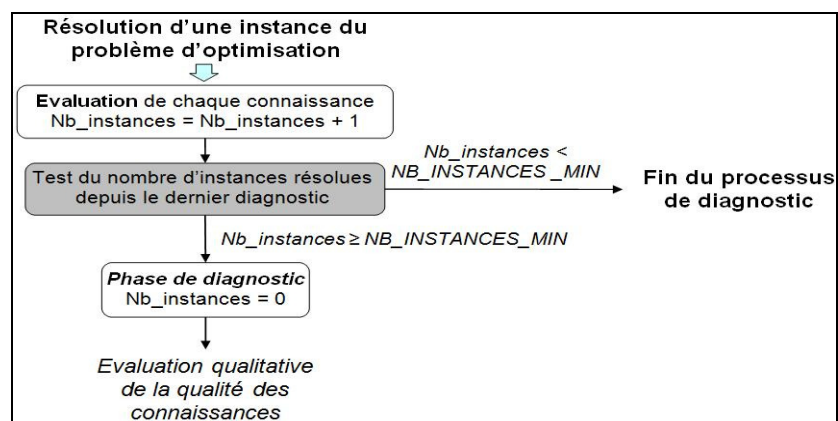


Fig. 3 – Approche générale de diagnostic

3.2 Phase d'évaluation

Pour chaque instance du problème résolue, deux types d'informations sont extraites : une évaluation locale de chaque connaissance et une évaluation globale de l'efficacité du système, c'est-à-dire sa capacité à obtenir un bon résultat.

3.2.1 Evaluation locale

Nous proposons de caractériser la qualité des connaissances à l'aide de quatre critères : le nombre de faux négatifs (nb_{FN}), le nombre de faux positifs (nb_{FP}), le nombre de vrais négatifs (nb_{VN}) et le nombre de vrais positifs (nb_{VP}). En vue d'évaluer ces critères, nous proposons d'appliquer la même approche que celle que nous avons proposée dans (Taillandier, 2008). Celle-ci est fondée sur l'analyse des meilleurs chemins. Un meilleur chemin est une séquence d'au moins deux états, qui a pour racine la racine de l'arbre d'états (ou d'un sous-arbre) et pour état final, le meilleur état de l'arbre (ou du sous-arbre). Une fois l'ensemble des meilleurs chemins définis pour un arbre d'états, l'évaluation des succès et des échecs des différentes connaissances revient à analyser ces meilleurs chemins. L'évaluation des faux négatifs, des faux positifs, des vrais négatifs et des vrais positifs est différent pour chaque type de connaissances. Elle dépend en effet de la nature de celle-ci. Par exemple, comme évoqué en partie 2.3, le nombre de faux négatifs peut ne pas avoir de sens pour certaines connaissances.

Pour le critère de validité, un faux positif est un état valide qui n'appartient pas à un meilleur chemin (donc inutile) alors que son prédécesseur appartient à un meilleur chemin. Un vrai positif est un état valide qui appartient à un meilleur chemin. Enfin, un faux négatif est un état invalide qui n'appartient pas à un meilleur chemin alors que son prédécesseur appartient à un meilleur chemin.

Concernant le critère de fin de cycle, un faux négatif est un état pour lequel le critère a proposé de continuer l'exploration alors que le meilleur état avait déjà été trouvé. Un vrai négatif est un état appartenant au meilleur chemin et qui n'est pas le meilleur état (et pour lequel il est donc pertinent de continuer l'exploration). Enfin, un vrai positif représente le cas où le critère de fin de cycle a proposé d'arrêter l'exploration de l'arbre d'états juste après que le meilleur état a été trouvé.

Pour les connaissances d'application des actions, un faux positif est un cas où, à partir d'un état appartenant à un meilleur chemin, l'application de l'action a mené à un état qui n'appartient pas à ce meilleur chemin. Un faux négatif est un cas où, à partir d'un état appartenant à un meilleur chemin, l'application de l'action a mené à un état du même meilleur chemin mais pour lequel l'action n'a pas été appliquée en priorité (d'autres actions ont été testées auparavant). Un vrai positif est un cas où, à partir d'un état d'un meilleur chemin, l'action a été testée en priorité et a mené à un autre état du même meilleur chemin. Enfin, un vrai négatif est un cas pour lequel, à partir d'un état appartenant à un meilleur chemin, l'action n'a pas été proposée (et n'a donc pas mené à des états inutiles). La figure 4 donne un exemple de résultats obtenus après analyse locale d'un arbre d'états.

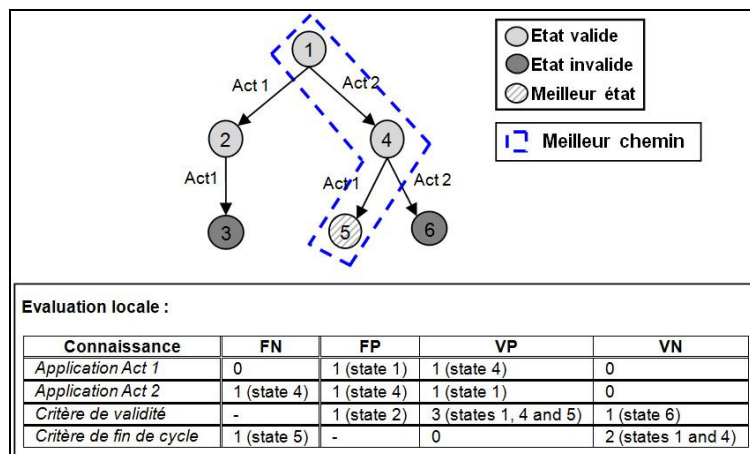


Fig. 4 – Exemple de résultats pour une analyse locale

3.2.2 Evaluation globale

L'évaluation locale des connaissances permet d'obtenir des informations sur l'efficacité du système (le temps nécessaire à l'exploration). En effet, cette évaluation permet de savoir si les connaissances guident bien le système vers les meilleurs états ou si elles entraînent la visite de nombreux états inutiles. Néanmoins, comme évoqué en partie 2.3, l'évaluation locale ne permet pas de fournir d'information concernant l'efficacité du système (la qualité du meilleur état trouvé). Effectivement, il n'est pas possible de savoir s'il aurait été possible de trouver un meilleur état si plus d'états avaient été visités. Nous proposons donc de définir un nouveau critère chargé d'évaluer l'efficacité du système. L'évaluation de ce critère est fondée sur la qualité du meilleur état obtenu (la valeur de Q_p pour le meilleur état trouvé) pour chaque instance du problème résolue.

3.3 Phase de diagnostic

La phase de diagnostic consiste à évaluer la qualité globale des connaissances. Cette évaluation est réalisée à partir de différents critères : la qualité des différentes connaissances prises individuellement et l'efficacité du système.

3.3.1 Critères de diagnostic

Afin d'établir une évaluation globale des connaissances, nous proposons différents critères.

Les premiers concernent la qualité individuelle de chaque connaissance. Nous proposons de caractériser la qualité de chaque connaissance sous la forme d'une note comprise entre 0 et 1. Une note de 0 signifie que la connaissance est a priori très déficiente. Une note de 1 que la connaissance est a priori parfaite. La note donnée

pour une connaissance K et pour un ensemble d'instances résolues I dépend des succès/échecs obtenus pour l'ensemble des instances de I (cf. 3.2.1) :

$$Note_K(I) = \frac{nb_{VP} + nb_{VN}}{nb_{VP} + nb_{VN} + nb_{FP} + nb_{FN}}$$

Ainsi, la qualité individuelle d'une connaissance est égale au rapport du nombre de succès connus par la connaissance par rapport au nombre d'échecs qu'elle a connu.

Un autre critère est le critère d'efficacité. Notre approche demande de définir une fonction $Efficacité(I)$, qui donne une note à l'efficacité du système pour la résolution de l'ensemble des instances de I . Cette note dépend de la qualité obtenue pour chaque instance de I (cf. 3.2.2). Nous rappelons que chaque instance résolue est caractérisée par la valeur de Q_p pour le meilleur état trouvé pour cette instance. La note d'efficacité, qui est comprise entre 0 et 1, doit être définie en fonction des besoins de l'utilisateur par rapport au système de résolution de problèmes. Un exemple de fonction $Efficacité(I)$ est définie en partie 4.3.

L'évaluation de ces critères donne une indication à l'utilisateur du système sur les connaissances à réviser en priorité. Elle peut également donner des informations sur la nécessité d'explorer plus d'états (si la note d'efficacité est mauvaise). Néanmoins, en raison du nombre important de critères et du caractère numérique des notes données, l'analyse seule de celles-ci peut s'avérer insuffisant pour qu'un utilisateur du système puisse prendre une décision quant à la nécessité de réviser les connaissances. Il est donc intéressant de disposer d'une méthode capable, à partir de ces notes, de fournir une évaluation des connaissances dans leur globalité. Nous proposons d'autre part, afin de faciliter l'analyse du résultat de cette évaluation globale, que le résultat de l'évaluation prenne la forme d'une valeur qualitative.

3.3.2 Décision multicritère

Nous proposons donc d'évaluer globalement la qualité du jeu de connaissances à partir des notes obtenues par chaque critère. Nous définissons cinq niveaux de qualité pour la qualité du jeu de connaissances courant : {*très mauvais, mauvais, moyen, bon, très bon*}.

L'objectif de l'évaluation globale du jeu de connaissances est donc de déterminer, à partir des notes obtenues, à quel niveau de qualité appartient le jeu de connaissances courant.

La qualité du jeu de connaissance courant est caractérisée par un vecteur de valeurs correspondant aux valeurs courantes des critères (les notes données par chaque connaissances ainsi celle donnée pour l'efficacité du système).

Nous posons l'hypothèse que chaque critère peut être évalué individuellement par un niveau de qualité (*très mauvais, mauvais, moyen, bon, très bon*) et que pour chaque critère il est possible de définir une valeur comprise entre [0,1] caractérisant chaque niveau de qualité. Ainsi, les cinq niveaux de qualité globale du jeu de connaissance courant sont aussi chacun caractérisé par un vecteur de valeurs.

Nous notons respectivement $V^{courant}$, $V^{trèsMauvais}$, $V^{mauvaie}$, V^{moyen} , V^{bon} et $V^{trèsBon}$, les vecteurs de notes caractérisant : le jeu de connaissances courant, le niveau de qualité du jeu de connaissances *très mauvais*, *mauvais*, *moyen*, *bon* et *très bon*. L'objectif de l'évaluation globale est donc de déterminer avec quel vecteur de notes appairer le vecteur de notes courant.

Il existe dans la littérature de nombreuses approches visant à résoudre ce type de problèmes. Parmi elles, certaines cherchent à agréger l'ensemble des critères en un critère unique (fonction d'utilité) qui est utilisé pour prendre une décision (Jacquet-Lagreze & Siskos, 1982). D'autres se fondent sur la comparaison par paire de décisions possibles à l'aide d'une relation de surclassement (Siskos, Wäscher & Winkels, 1983 ; Roy, 1991 ; Yu, 1992). Enfin, certaines approches, fortement interactives, proposent de chercher la meilleure décision par modifications successives d'une décision initiale (Benayoun, Laritchev, de Mongolfier, & Tegny, 1971 ; Geoffrion, Dyer & Feinberg, 1972).

Dans le cadre de notre problème d'évaluation globale des connaissances, nous disposons de critères de décisions peu fiables. En effet, le problème d'interdépendance entre connaissances (cf. 2.3) peut fausser les notes obtenues par certaines connaissances. Il est donc important de choisir une méthode de décision multicritère permettant de prendre cet aspect en compte.

Nous avons proposé dans (Taillandier, 2008) de résoudre un problème similaire (mais avec seulement deux niveaux de qualité pour les connaissances) à l'aide de la méthode ELECTRE TRI (Yu, 1992). Nous avons ainsi comparé le vecteur de notes courant à un vecteur de notes de référence représentant la frontière entre les deux niveaux de qualité à l'aide de la relation de surclassement définie par la méthode ELECTRE TRI. Ce type d'approche permet de faire face au problème d'incomparabilité entre critères, mais au prix d'un manque de clarté des résultats (Ben Mena, 2000). Nous proposons dans cet article d'utiliser la théorie des fonctions de croyance (Shafer, 1976) pour définir le niveau de qualité du jeu de connaissances courant. Cette théorie permet de gérer l'incomplétude, l'incertitude et l'imprécision des critères et est donc particulièrement adaptée à notre problème de décision multicritère.

3.3.3 Application de la théorie des fonctions de croyance

Généralité sur la théorie des fonctions de croyance

La théorie des fonctions de croyance est sur fondée sur le travail de Dempster en 1967 (Dempster, 1967) sur les distributions de probabilité. Cette théorie fut appliquée avec succès pour de nombreux problèmes (Omrani, Ion-Boussier & Trigano, 2007 ; Olteanu-Raimond & Mustière, 2008).

Soit Θ le cadre de discernement. Ce cadre contient l'ensemble des hypothèses correspondant aux solutions du problème. Un exemple d'hypothèse pour notre problème est « le jeu de connaissances courant est de qualité moyenne ». Afin de simplifier l'énoncé des formules données dans la suite de cet article, nous notons V^i l'hypothèse « le jeu de connaissances courant est de qualité i ».

Pour notre application le cadre de discernement est composé de cinq hypothèses (correspondant aux cinq niveaux de qualité) :

$$\Theta = \{V^{\text{trèsMauvais}}, V^{\text{mauvais}}, V^{\text{moyen}}, V^{\text{bon}}, V^{\text{trèsBon}}\}$$

Ce cadre de discernement permet de définir l'ensemble des propositions possibles, c'est-à-dire l'ensemble des combinaisons d'hypothèses possibles. Soit 2^Θ l'ensemble de tous les sous-ensembles possibles de Θ .

$$2^\Theta = \{\emptyset, \{V^{\text{trèsMauvais}}\}, \{V^{\text{mauvais}}\}, \dots, \{V^{\text{trèsMauvais}}, V^{\text{mauvais}}\}, \dots, \Theta\}$$

Chaque ensemble $\{V^i, \dots, V^j\}$ représente la proposition selon laquelle la solution du problème (le niveau de qualité du jeu de connaissances dans notre cas) est l'une des hypothèses de cet ensemble. \emptyset signifie qu'aucune hypothèse n'est vraie et Θ (qui est le cadre de discernement) que la solution est l'une des hypothèses.

La théorie des fonctions de croyance est, comme son nom l'indique, fondée sur l'utilisation de fonction de croyances. Pour une proposition donnée $P \in 2^\Theta$, ces fonctions affectent une masse de croyance basique, $m_j(P)$, qui traduit le degré de croyance pour un critère j que la proposition P soit vraie. Les masses de croyances sont comprises entre 0 et 1 et sont définies telles que :

$$\sum_{P \in 2^\Theta} m_j(P) = 1$$

Méthode de prise de décision

Notre méthode de prise de décision, qui est sur fondée sur la théorie des fonctions de croyance, est composée de 4 étapes (figure 5).

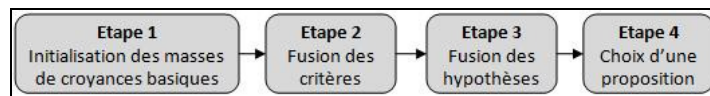


Fig. 5 – Méthode de prise de décision

Etape 1

La première étape consiste à initialiser les masses de croyance basiques. Pour cette étape, nous proposons d'utiliser le travail de (Appriou, 1991) qui propose de « spécialiser » les critères pour une hypothèse du cadre de discernement. Ainsi, les critères donnent uniquement trois types d'opinions : en faveur de l'hypothèse, en sa défaveur et ne se prononce pas. Pour chaque niveau de qualité i , un sous-ensemble S^i de 2^Θ est défini :

$$S^i = \{V^i, \neg V^i, \Theta\}$$

Evaluation de connaissances par la théorie des fonctions de croyance

- V^i : cette proposition signifie que la qualité du jeu de connaissances courant est i
- $\neg V^i = \Theta - \{V^i\}$: cette proposition signifie que la qualité du jeu de connaissances courant n'est pas i
- Θ : cette proposition marque l'ignorance : elle correspond au cadre de discernement, chaque hypothèse peut être vraie.

Ainsi, initialiser les masses de croyances basiques revient à calculer, pour chaque critère j et pour chaque niveau de qualité i , les masses de croyances basiques $m_j(V^i)$, $m_j(\neg V^i)$ et $m_j(\Theta)$.

En vue de calculer ces masses de croyance basiques, nous proposons, en figure 6, différentes fonctions de croyances. Ces fonctions permettent de donner une valeur aux masses de croyance basiques. Par exemple, si pour le critère d'efficacité le jeu de connaissances courant obtient une note de 1 et que la valeur pour ce critère pour le niveau de qualité *très mauvais* est égale à 0, nous obtiendrons pour les masses de croyance basiques :

$$m_{eff}(V^{trèsMauvais}) = 0, m_{eff}(\neg V^{trèsMauvais}) = 1 \text{ et } m_{eff}(\Theta) = 0$$

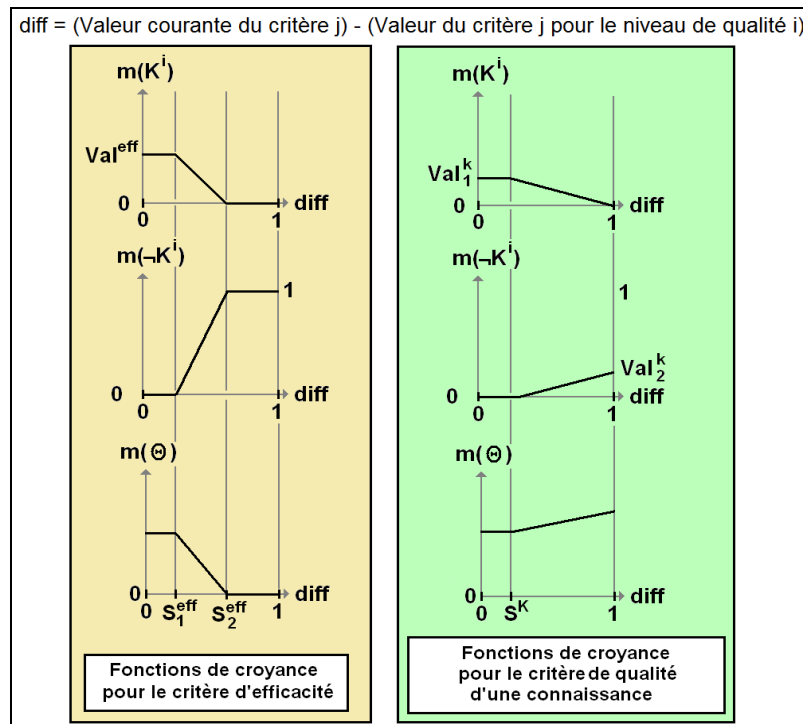


Fig. 6 – Fonctions de croyance

Pour le critère d'efficacité, les fonctions de croyances sont définies de telle sorte que les petites variations de la valeur du critère ne soient pas significatives pour la valeur de la masse de croyance. Néanmoins, dès que la différence dépasse un certain seuil, le critère peut directement rejeter une proposition.

Concernant les critères de qualité des connaissances, les fonctions de croyance que nous avons définies permettent également de ne pas prendre en considération les petites variations. Les variations commencent à être prises en compte dès qu'elles dépassent un certain seuil. La mise en œuvre de notre approche demande de définir différents seuils (Val^{eff} , S_1^{eff} et S_2^{eff} pour le critère d'efficacité et Val_1^k , Val_2^k et S^k pour le critère de qualité de chaque connaissance).

Etape 2

Cette étape consiste à fusionner les critères entre eux. Nous proposons pour cela d'utiliser l'opérateur de fusion introduit par (Smets, 1990) pour calculer les masses de croyance résultantes de la fusion de deux critères :

$$\forall P \in 2^\Theta, m_{i_2}(P) = \sum_{P' \cap P'' = P} m_j(P') m_2(P'')$$

Cet opérateur est commutatif et associatif. Il est donc possible de combiner à nouveau avec un nouveau critère des masses de croyance résultantes d'une première fusion.

La fusion des critères peut introduire un conflit, noté ϕ , c'est-à-dire un cas où un critère a une masse de croyance non nulle en faveur d'une proposition V^i alors qu'un autre critère a une masse de croyance non nulle pour la proposition $\neg V^i$. Ce conflit sera pris en compte pour la décision finale.

Soit C , l'ensemble des critères. A la fin de cette étape de fusion des critères, nous obtenons, pour chaque niveau de qualité i , les masses de croyance combinés $m_C(V^i)$, $m_C(\neg V^i)$, $m_C(\Theta)$ et $m_C(\phi)$.

Etape 3

Cette étape consiste à combiner entre elles les hypothèses. Il est intéressant de procéder à cette fusion de façon à prendre en compte, dans la décision finale, le fait que certains critères ont rejeté des hypothèses ($\neg V^i$).

Nous proposons d'utiliser l'opérateur de Dempster (Dempster, 1967) pour combiner les masses de croyance résultantes de la fusion de deux hypothèses :

$$\forall P \in 2^\Theta, m_{V^i, V^j}(P) = \frac{1}{1 - m_{V^i, V^j}(\phi)} \sum_{P' \cap P'' = P} m_{V^i}(P') m_{V^j}(P'')$$

Cet opérateur est également commutatif et associatif. Il est donc possible de combiner à nouveau avec une nouvelle hypothèse des masses de croyance résultantes d'une première fusion.

Cet opérateur permet en plus de normaliser les masses de croyances obtenues par l'intermédiaire du facteur $\frac{1}{1 - m_{V^i, V^j}(\phi)}$

A la fin de cette étape, nous obtenons une masse de croyance pour chaque proposition $m_C(V^{\text{trèsMauvais}})$, $m_C(V^{\text{mauvais}})$, ..., $m_C(V^{\text{trèsMauvais}} \text{ or } V^{\text{Mauvais}})$, ..., $m_C(\Theta)$ et $m_C(\phi)$.

Etape 4

Cette dernière étape consiste à sélectionner la meilleure proposition. Nous souhaitons choisir une unique hypothèse (le niveau de qualité du jeu de connaissances) et non un ensemble d'hypothèses. Nous proposons donc d'utiliser la probabilité pignistique (Smets & Kennes, 1994), classiquement utilisé pour ce type de problème, afin de sélectionner la meilleure hypothèse (celle qui maximise cette probabilité).

La probabilité pignistique d'une proposition A est donnée par la formule suivante :

$$P(A) = \sum_{A \subseteq B} m(B) \frac{|A|}{|B|}$$

Analyse de la robustesse de la décision

Un point important dans une décision multicritère concerne la robustesse de la décision (Roy, 1998). Une approche pour tester sa robustesse est d'analyser les variations de décision lorsque l'on fait varier les différents paramètres introduits. Dans le cadre de notre application, nous proposons de prendre une décision à partir de plusieurs jeux de paramètres (les seuils des fonctions de croyance) et de procéder par vote majoritaire pour définir la décision finale. Le pourcentage de vote en faveur de la décision majoritaire permet de quantifier la fiabilité de la décision. Si ce pourcentage est faible, il devient important d'analyser les pourcentages de votes obtenus par les autres décisions afin d'obtenir une meilleure évaluation de la qualité du jeu de connaissances courant.

4 Application au contexte de la généralisation cartographique

4.1 La généralisation cartographique

Nous proposons d'appliquer notre d'évaluation des connaissances dans le cadre d'un système de généralisation cartographique. La généralisation cartographique est le processus qui consiste à dériver, à partir de données géographiques très détaillées, des données qui puissent être affichées ou imprimées à une échelle donnée. L'objectif de ce processus est d'assurer la lisibilité de la carte tout en conservant l'information essentielle des données initiales. Ce processus demande d'appliquer de nombreuses opérations telles que des grossissements d'objets, des suppressions, des déplacements, etc. La figure 7 donne un exemple de généralisation cartographique.

La généralisation automatique est un contexte applicatif particulièrement intéressant. En effet, ce problème est loin d'être résolu. De plus, il intéresse directement les agences cartographiques telles que l'Institut Géographique National qui souhaitent améliorer leur ligne de production de cartes. Enfin, la multiplication des sites Internet proposant de concevoir sa propre carte augmente les besoins en méthodes fiables et efficaces de généralisation automatique. L'automatisation de la généralisation est un problème complexe. Une approche pour y faire face est de procéder par résolution locale à l'aide de connaissances (Brassel & Weibel, 1988). Cette approche implique de maintenir une base de connaissances. En particulier, d'adapter les connaissances quand de nouveaux éléments (par exemple, nouvel algorithme de généralisation) sont intégrés dans le système ou lorsque les besoins de l'utilisateur (spécification des cartes) changent. Aujourd'hui, ces adaptations sont réalisées « à la main » par des experts en généralisation. Le processus d'adaptation est souvent long et fastidieux. L'intégration dans le système d'un module de diagnostic en ligne, capable de déterminer quand les connaissances nécessitent d'être révisées et de quelles parts des connaissances sont déficientes est donc particulièrement intéressant.

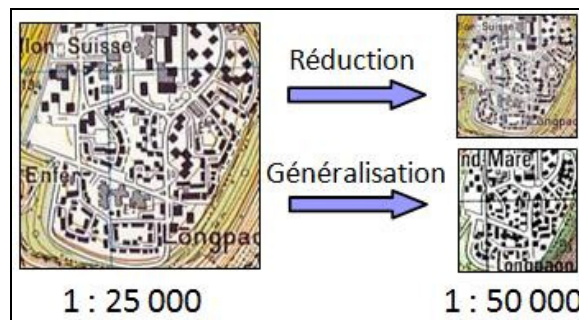


Fig. 7– Généralisation cartographique

4.2 Le système de généralisation

Le système de généralisation que nous avons utilisé pour ces expérimentations est fondé sur le modèle AGENT (Barrault et al., 2001 ; Ruas & Duchêne, 2007). Il suit les spécifications présentées en partie 2.2.

Le système généralise les objets ou les groupe d'objets géographiques par l'intermédiaire d'une exploration informée d'arbres d'états. Les états représentent les états géométriques des objets ou des groupes d'objets. Ils sont évalués par une fonction de satisfaction qui traduit le respect de contraintes cartographiques (qui traduisent les spécifications de la carte). Par exemple, une contrainte cartographique peut être pour un bâtiment d'être suffisamment gros pour être lisible. La satisfaction d'un état est un réel compris entre 1 et 10. Une satisfaction de 10 représente un état qui satisfait parfaitement les contraintes cartographique. Un état dont la satisfaction est inférieure à 5 n'est pas acceptable d'un point de vue cartographique.

4.3 Application de notre approche

Nous avons appliqué notre approche d'évaluation de la qualité des connaissances sur le système de généralisation.

Quant à la fonction d'évaluation de l'efficacité, nous avons définie empiriquement la fonction suivante :

$$\text{Efficacité}(I) = \left(\frac{\text{PremierQuartile}(\{S(obj)\}_{obj \in I}) + \text{Moyenne}(\{S(obj)\}_{obj \in I})}{20} \right)^2$$

Avec $S(obj)$ renvoyant la meilleure satisfaction trouvée pour la généralisation de l'objet obj .

Cette fonction permet de prendre en compte la moyenne des satisfactions obtenues sur l'ensemble des objets généralisés et de pondérer ce résultat par la valeur du premier quartile des satisfactions. L'intérêt de cette pondération provient du fait qu'il est souvent préférable pour les agences cartographiques d'obtenir trois quarts d'objets très bien généralisés et un quart mal généralisés (qui peut être retouché par un technicien) que d'obtenir un résultat homogène moyen qui nécessiterait beaucoup plus de travail de retouche. Le facteur $1/20$ est utilisé pour normaliser la valeur de la fonction. Effectivement, nous rappelons que la satisfaction d'un objet géographique est comprise entre 1 et 10. Nous avons introduit une puissance 2 afin d'accentuer les écarts entre les valeurs. La définition de cette fonction est un point clef de notre approche. Dans ce cadre, d'autres fonctions plus complexes, par exemple basées sur les moyennes pondérées ordonnées (Yager, 1998), auraient pu être utilisées.

Concernant les paramètres propres à la méthode de décision multicritères, nous avons défini leur valeur empiriquement par le biais d'expérimentations menées sur d'autres zones géographiques et avec d'autres jeux de connaissances. En règle générale, les valeurs des paramètres ont été choisies de manière à privilégier l'efficacité du système plutôt que son efficacité. En effet, notre priorité est d'obtenir de bons résultats. Si ces derniers sont très mauvais, quelle que soit l'efficacité du système, les connaissances devront être révisées. Vingt jeux de paramètres ont été utilisés pour tester la robustesse de la décision prise.

4.4 Expérimentation

Nous avons testé notre approche de diagnostic dans le cadre de la généralisation des groupements de bâtiments. La généralisation de ces groupements est particulièrement intéressante en raison de la difficulté existante à définir de bonnes connaissances pour celle-ci et en raison de son coût computationnel élevé.

Nous avons défini, avec l'aide d'experts en généralisation, six contraintes cartographiques ainsi que cinq actions de généralisation.

Nous avons testé notre approche de diagnostic avec quatre jeux de connaissances. Chacun de ces jeux correspond à un scénario différent de l'utilisation du système de généralisation :

- $K_{Efficient}$: jeu de connaissances qui ne propose aucune action. Ce jeu de connaissances assure de ne visiter que l'état initial et permet donc au système de maximiser son efficacité. La qualité du résultat (qui correspond à l'état initial) est toujours très mauvaise.
- $K_{Efficace}$: jeu de connaissances qui propose d'appliquer tous les actions possibles pour tous les états et qui utilise un critère de validité faible. Ce jeu de connaissances assure de trouver, pour les groupements généralisés, le meilleur état possible compte tenu des actions et des contraintes définies. Il demande, en revanche, de visiter un très grand nombre d'états et n'est donc pas efficace du tout.
- K_{Expert} : jeu de connaissances défini par un expert en cartographie qui est également un expert du modèle AGENT. Les résultats obtenus avec ce jeu de connaissances sont bons en termes d'efficacité mais moyens en termes d'efficacité.
- $K_{Révisé}$: jeu de connaissances obtenu après révision du jeu de connaissances K_{Expert} . Les connaissances ont été révisées hors-ligne à l'aide de l'approche que nous avons proposée dans (Taillandier et al., 2008). Les résultats obtenus avec ce jeu de connaissances sont bons à la fois en termes d'efficacité et d'efficacité.

Nous avons testé notre approche de diagnostic à partir de la généralisation de 30 groupements de bâtiments. Nous avons choisi ces derniers aléatoirement afin de se rapprocher du scénario réaliste dans lequel un utilisateur souhaite généraliser un grand nombre de groupements de bâtiments qui sont généralisés dans un ordre aléatoire. Le nombre 30 a été défini empiriquement. Il est suffisant pour donner une évaluation fiable de la qualité des connaissances et pas trop élevé afin de donner rapidement cette évaluation. En effet, si un utilisateur doit généraliser 5000 groupements de bâtiments, il est préférable de le prévenir dès le 30^{ième} groupement de bâtiments que les connaissances ne sont pas bonnes plutôt que d'attendre la fin des 5000 généralisations pour le prévenir.

4.5 Résultats

La table 1 présente les résultats obtenus pour l'évaluation de nos quatre jeux de connaissances. $K_{Efficient}$ a été classé comme mauvais avec un pourcentage de votes de 100%. $K_{Efficace}$ et K_{Expert} ont été classés comme moyen, avec un pourcentage de votes de 50% pour le premier et de 100% pour le second. Enfin, $K_{Révisé}$ a été classé comme bon avec un pourcentage de votes de 100%.

Ces résultats sont cohérents avec les jeux de connaissances testés. En effet, le seul jeu de connaissances classé comme mauvais est $K_{Efficient}$ avec lequel aucune action n'est jamais appliquée : les résultats cartographiques obtenus sont donc très mauvais. Le seul jeu de connaissances classé comme bon est $K_{Révisé}$, qui est déjà une version

révisée de K_{Expert} et qui donne de bons résultats à la fois en termes d'efficacité et d'efficacités. Les deux autres jeux de connaissances ont été classés comme moyen. Ces deux jeux de connaissances permettent d'obtenir de bons résultats en termes d'efficacité mais pas en termes d'efficacités. K_{Expert} est le meilleur des deux en termes d'efficacités. Le pourcentage de votes pour le classé comme moyen est de 100%. Le pourcentage de vote pour classer $K_{Efficace}$ comme moyen n'est que de 50%. En fait, le module de diagnostic a hésité entre classer ce jeu de connaissances comme moyen ou comme mauvais. Le niveau de qualité mauvais a obtenu, pour ce jeu de connaissances, 40% des votes. Ce constat est cohérent avec les performances obtenues avec ce jeu de connaissances.

Table 1. Résultat de l'évaluation des connaissances : niveau de qualité et pourcentage de votes en faveur de ce niveau.

	Niveau de qualité	Pourcentage de votes en faveur du niveau majoritaire
$K_{Efficent}$	Mauvais	100%
$K_{Efficace}$	Moyen	50%
K_{Expert}	Moyen	100%
$K_{Révisé}$	Bon	100%

5 Conclusion

Dans cet article, nous avons proposé une approche d'évaluation automatique de la qualité des connaissances fondée sur la théorie des fonctions de croyance. Nous avons testé notre approche au travers d'une expérimentation menée dans le domaine de la généralisation cartographique. Cette expérimentation a montré que notre approche pouvait fournir une évaluation pertinente de la qualité d'un jeu de connaissances.

Un point clef de notre approche concerne la fonction d'évaluation de l'efficacité. Définir cette fonction peut se révéler complexe. Ainsi, une perspective intéressante serait de développer des méthodes permettant d'aider l'utilisateur à définir cette fonction. Différents travaux existant peuvent servir de base pour le développement d'une telle méthode (Hubert & Ruas 2003, Christophe 2008).

Une autre perspective intéressante serait d'étudier plus en détails l'influence des valeurs des paramètres dans la prise de décision. En effet, la qualité de l'évaluation est directement liée à la pertinence des valeurs choisies pour ces paramètres. Il est donc important de choisir leur valeur judicieusement.

Références

APPRIOU A.(1991). Probabilité et incertitude en fusion de données multi-senseurs, *Revue Scientifique et Technique de la Défense 1*, pp. 27--40.

- BARRAULT M., REGNAULD N., DUCHÊNE C., HAIRE K., BAEJIS C., DEMAZEY Y., HARDY P., MACKANESS, W., RUAS A. & WEIBEL R. (2001). Integrating multi-agent, object-oriented, and algorithmic techniques for improved automated map generalization. *ICC*. pp. 2100–2116
- BENAYOUN R., LARITCHEV O., DE MONGOLFIER J. & TEGNY J. (1971). Linear programming with multiple objective functions: STEP method (STEM). *Math. Program.*, 1, 3, pp. 366–375.
- BEN MENA S. (2000), Introduction aux méthodes multicritères d'aide à la décision, *Biotechnol. Agro. Soc. Environ.* 4(2), pp. 83–93.
- BRASSEL K. & WEIBEL R. (1988). A review and conceptual framework of automated map generalization. *IJGIS*.
- CHRISTOPHE S. (2008). Creative Cartography based on Dialogue, in 'In proceedings of AutoCarto', Shepherdstown, West Virginia, CD-ROM.
- DEMPSTER A. (1967). Upper and lower probabilities induced by multivalued mapping, *Annals of Mathematical Statistics* 38, 1967, pp. 325--339.
- FEIGENBAUM E. (1977). The art of artificial intelligence 1: Themes and case studies of knowledge engineering, Technical report, Stanford University,
- GEOFFRION A., DYER J. & FEINBERG A. (1972). An interactive approach for multicriterion optimisation with an application to the operation of an academic department. *Manage. Sci.*, 19, 4.
- HUBERT F. & RUAS A. (2003). A method based on samples to capture user needs for generalisation, *workshop on progress in automated map generalisation*, Paris, 2003.
- JACQUET-LAGREZE E. & SISKOS J. (1982). Assessing a set of additive utility functions for multicriteria decision making. *The UTA method. Eur. J. Oper. Res.*, 10, 2, 151–164.
- OLTEANU-RAIMOND A.M. & MUSTIÈRE S. (2008). Data matching-a matter of belief, *The International Symposium on Spatial Data Handling (SDH)*, Montpellier, France, pp. 501--519.
- OMRANI H., ION-BOUSSIER L. & TRIGANO P. (2007). A new approach for impacts assessment of urban mobility, *WSEAS trans. on Information science and applications* 4(3), pp. 439--444.
- ROY B. (1998). A missing link in OR-AD: Robustness analysis. *Foundations of Computing and Decision Science*, 23, 3, 49-73.
- ROY B. (1991). The outranking approach and the foundations of ELECTRE methods. *Theory and Decision*, 31.
- RUAS A. & DUCHÊNE C. (2007). A Prototype Generalisation System Based on the Multi-Agent Paradigm. *Generalisation of Geographic Information: Cartographic Modelling and Applications*.
- SHAFFER G. (1976). A mathematical theory of evidence, *Princeton University Press*.
- SISKOS J., WÄSCHER G. & WINKELS H. (1983). A bibliography of outranking approaches. *Cahiers du Lamsade*, Paris Dauphine.
- SMETS P. (1990). Constructing the pignistic probability function in a context of uncertainty. *Uncertainty in Artificial Intelligence* 5, pp. 29--39.
- SMETS P. & KENNES R. (1994). The transferable belief model, *Artificial Intelligence* 66(2).
- TAILLANDIER P. (2008). Knowledge diagnosis in systems based on an informed tree search strategy: application to cartographic generalisation, *CSTST Student Workshop*, Cergy-Pontoise (Paris), France, pp. 589--594.
- TAILLANDIER P., DUCHENE C. & DROGOU A. (2008). Knowledge revision in systems based on an informed tree search strategy: application to cartographic generalization. *CSTST*. Cergy-Pontoise (Paris), France, pp. 273--278.
- YAGER R.R. (1988). On ordered weighted averaging aggregation operators in multicriteria decision-making. *IEEE Trans, Systems, Man & Cybernetics*, 18, 183-190.
- YU.W. (1992). Aide multicritère à la décision dans le cadre de la problématique du tri : Concepts, méthodes et applications. *Thèse à Université Paris Dauphine*.