

# Étude comparative d'approximation de fonctions de croyances généralisées

## Comparative study of approximations of generalized beliefs functions

Pascal Djiknavorian<sup>1</sup>

Arnaud Martin<sup>2</sup>

Dominic Grenier<sup>1</sup>

Pierre Valin<sup>3</sup>

<sup>1</sup> Université Laval, LRTS

<sup>2</sup> ENSIETA, E<sup>3</sup>I<sup>2</sup> EA3876

<sup>3</sup> DRDC, C2 DSS section

1065, av. de la Médecine, Québec, QC, G1V 0A6, Canada, dgrenier@gel.ulaval.ca

2 rue François Verny 29806 Brest Cedex 9, Arnaud.Martin@ensieta.fr

Défense R&D Canada, Valcartier, Québec, Canada, Pierre.Valin@drdc-rddc.gc.ca

### Résumé :

Ce travail est une analyse comparative du comportement d'une implémentation de l'approximation externe de structures de croyances utilisant l'approche de regroupement hiérarchique sur les fonctions de croyances généralisées appliquées à un cas de combinaison d'information en provenance de capteurs ESM (Electronic Support Measures). L'objectif est de maintenir un taux de bonnes décisions du système d'aide à la décision fondé sur cette approximation par rapport à un système sans approximation tout en obtenant un gain en termes de temps d'exécution.

### Mots-clés :

Fonctions de croyances, approximation, décision.

### Abstract:

This paper is a comparative behaviour analysis of an implementation of the outer approximation of belief structures using a hierarchical clustering approach applied to general belief functions applied for a case of information fusion coming from an ESM (Electronic Support Measures) sensor. The objective is to maintain a good decision rate for the decision support system based on this approximation relatively to a system without approximation, while obtaining a gain in terms of execution time.

### Keywords:

Belief functions, approximation, decision.

## 1 Introduction

Ce travail consiste en une analyse comparative du comportement d'une implémentation de l'approximation externe de structures de croyance utilisant l'approche de regroupement hiérarchique [7] sur les fonctions de croyances

généralisées appliquée à un cas de combinaison d'information en provenance d'un système d'identification ami/ennemi à partir d'un capteur ESM (Electronic Support Measures). L'objectif est de conserver une bonne décision tout en réduisant le temps d'exécution.

Nous considérons ici que le système en sortie du capteur ESM fournit une information distribuée parmi trois classes : *Ami*, *Neutre*, et *Ennemi*. Ce capteur n'est pas en mesure de spécifier une classe consistant en une intersection de classe telle que la théorie de Dezert-Smarandache le permet. Le système d'aide à la décision doit donner une réponse compatible au standard de l'OTAN, le STANAG 1241, qui classe l'allégeance parmi cinq classes : *ami*, *neutre*, *ennemi*, *préssumé ami*, et *suspect*. Nous considérons ici un système de fusion dynamique de l'information. C'est-à-dire qu'il combine une paire d'informations issue du capteur, datant du temps  $t-1$  avec celle du temps  $t$ , au temps  $t$ .

La prochaine section rappelle la théorie sur les fonctions de croyances, les méthodes de combinaison et de décision. La section 3 porte sur les méthodes d'approximations. La section 4 présente les résultats de simulations avant de conclure par un bilan des conclusions tirées.

## 2 Fonctions de croyances

Le cadre de discernement est l'ensemble qui définit la liste des singletons ou hypothèses possibles d'un contexte précis. Il suppose des hypothèses exclusives et exhaustives. Le cadre de discernement est défini ci-dessous :

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_n\} \quad (1)$$

Du cadre de discernement, on obtient le cadre de raisonnement, lequel présente la liste des ensembles sur laquelle le raisonnement pour la décision se déroule. Pour la théorie de Dempster-Shafer, ce cadre de raisonnement correspond à l'ensemble de puissance  $2^\Theta$  [6,9] :

$$2^\Theta = \{\emptyset, \{\theta_1\}, \{\theta_2\}, \dots, \{\theta_n\}, \{\theta_1 \cup \theta_2\}, \dots, \{\theta_1 \cup \theta_2 \cup \dots \cup \theta_n\}\} \quad (2)$$

Il est également possible de considérer un cadre de raisonnement étendu permettant les intersections [4]. Ce modèle permet ainsi de lever l'hypothèse d'exclusivité des éléments du cadre de discernement. Le nouveau cadre de raisonnement ainsi créé est désigné d'ensemble d'hyperpuissance.

L'équation (3) montre à quoi correspond l'ensemble d'hyperpuissance pour un cas de cardinal de cadre de discernement de 3 :

$$D^\Theta (|\Theta| = 3) \equiv \{\{\theta_1\}, \{\theta_2\}, \{\theta_3\}, \{\theta_1 \cup \theta_2\}, \{\theta_1 \cup \theta_3\}, \{\theta_2 \cup \theta_3\}, \{\theta_1 \cup \theta_2 \cup \theta_3\}, \{\theta_1 \cap \theta_2\}, \{\theta_1 \cap \theta_3\}, \{\theta_2 \cap \theta_3\}, \{\theta_1 \cap \theta_2 \cap \theta_3\}, \{(\theta_1 \cup \theta_2) \cap \theta_3\}, \{(\theta_1 \cup \theta_3) \cap \theta_2\}, \{(\theta_2 \cup \theta_3) \cap \theta_1\}, \{(\theta_1 \cap \theta_2) \cup \theta_3\}, \{(\theta_1 \cap \theta_3) \cup \theta_2\}, \{(\theta_2 \cap \theta_3) \cup \theta_1\}, \{(\theta_1 \cap \theta_2) \cup (\theta_1 \cap \theta_3) \cup (\theta_2 \cap \theta_3)\}\} \quad (3)$$

Il est possible d'ajouter des contraintes dans un modèle libre en indiquant les intersections correspondant à une classe d'équivalence vide. Par exemple, on peut considérer :  $\theta_1 \cap \theta_2 = \emptyset$ .

La fonction de masse est définie telle que  $m: 2^\Theta \rightarrow [0,1]$ . La somme des masses de tout ensemble possible correspond toujours à

l'unité de façon à ce que  $\sum_{A \subseteq 2^\Theta} m(A) = 1$ . Sous l'hypothèse d'un monde fermé on a  $m[\emptyset] = 0$ . Les éléments focaux correspondent aux éléments de masse non nulle.

## 2.1 Méthodes de combinaison

Les méthodes de combinaison considérées ici sont celles de Dempster-Shafer, de Dubois-Prade et PCR6 ; elles sont toutes fondées sur un principe de conjonction. Elles diffèrent ensuite sur la méthode de redistribution de la masse associée à l'ensemble vide ainsi que sur le cadre de raisonnement pour lequel elles ont été conçues.

La règle de Dubois-Prade correspond également à la règle de combinaison hybride dans le cas où l'on travaille avec des fonctions de croyances généralisées avec des cas ne requérant pas de contraintes de non-existence modifiant le cadre de discernement de façon dynamique.

La règle de combinaison de Dempster-Shafer [6,9] est la règle de combinaison classique permettant de combiner l'information et fonctionne pour un cadre de raisonnement correspondant à l'ensemble de puissance  $2^\Theta$ . La règle consiste simplement à procéder à une conjonction, puis les masses allant aux intersections sont mises à zéro et la fonction de croyance résultante est normalisée.

$$(m_1 \oplus m_2)(C) = \frac{1}{1 - K} m_c(C) \quad \forall C \subseteq \Theta$$

$$m_c(C) = \sum_{A \cap B} m_1(A) m_2(B) \quad (4)$$

$$K = \sum_{A \cap B = \emptyset} m_1(A) m_2(B) \quad A, B \subseteq \Theta$$

La règle de combinaison de Dubois-Prade (DP) fonctionne telle la règle conjonctive tant qu'il n'y a pas de conflits partiels (intersections vides). Au moment où cela a lieu, une disjonction au lieu de la conjonction. Cette disjonction fonctionne dans un ensemble

de puissance qui exclut l'ensemble vide. On note qu'une contrainte sur une disjonction n'est pas possible. Aussi, il n'est pas permis d'avoir une masse pour l'ensemble vide.

$$\begin{aligned}
m_{DP}(A) = & \sum_{\substack{X, Y \in 2^\Theta \\ X \cap Y = A \\ X \cap Y \neq \emptyset}} m_1(X)m_2(Y) \\
& + \sum_{\substack{X, Y \in 2^\Theta \\ X \cup Y = A \\ X \cap Y = \emptyset}} m_1(X)m_2(Y) \quad \forall A \in 2^\Theta \setminus \emptyset \quad (5) \\
m_{DP}(\emptyset) = & 0
\end{aligned}$$

La combinaison hybride (DSmH) de Dezert-Smarandache [11-13] fonctionne dans le cadre étendu des fonctions de croyances. Elle est décrite sous forme procédurale dans [6]. Cette règle fonctionne comme la règle conjonctive en absence de contrainte, autrement elle s'apparente à la règle de Dubois-Prade mais permet en plus des contraintes d'inexistences sur des singletons.

La règle de redistribution de conflit généralisée PCR6 est définie par [5] :

$$\begin{aligned}
m_{PCR6}(X) = & m_c(X) + \\
\sum_{\substack{Y \in D^\Theta \\ X \cap Y = \emptyset}} & \left( \frac{m_1(X)^2 m_2(Y)}{m_1(X) + m_2(Y)} + \frac{m_2(X)^2 m_1(Y)}{m_2(X) + m_1(Y)} \right) \quad (6)
\end{aligned}$$

Elle est similaire à la PCR5 pour deux fonctions de masses et consiste à reporter une masse conflictuelle aux ensembles ayant causé ce conflit, proportionnellement à leur masse.

## 2.2 Décision

Pour la prise de décision, la probabilité pignistique est généralement employée comme compromis. La transformation pignistique généralisée qui est en mesure de fonctionner avec les fonctions de croyances généralisées ou non dans la DSMT. La probabilité

pignistique [11] est définie par l'équation (7) où  $C_M(A)$  est le cardinal de l'ensemble A.

$$\Pr\{A\} = \sum_{X \in D^\Theta} \frac{C_M(X \cap A)}{C_M(X)} m(X) \quad \forall A \in D^\Theta \quad (7)$$

## 3 Méthodes d'approximation

Si les fonctions de masse généralement définies à l'origine comportent peu d'éléments focaux, le fait de les combiner de façon dynamique augmente le nombre d'éléments focaux de façon exponentielle. Le temps d'exécution est particulièrement crucial pour l'application considérée ici. Il devient important de trouver un moyen, telle une méthode d'approximation réduisant le nombre d'éléments focaux, afin de diminuer les temps de calcul sans dégrader la décision. Deux approches d'approximation sont présentées ici.

### 3.1 Méthode *k*lx de Tessem

La méthode d'approximation de Tessem, ou *k*lx, (présentée dans [14] et étudiée dans [1,5]) implique trois paramètres : *k*, le nombre minimum d'éléments focaux à conserver, *l*, le nombre maximum d'éléments focaux à conserver, et *x*, le seuil maximal du total de masse retirée par l'approximation.

L'algorithme se présente comme suit:

1. Sélectionner les *k* éléments focaux de masses les plus élevées;
2. Tant que la somme de leur masse est moindre que 1-x, et tant que leur nombre est moindre que l, ajouter le prochain élément de masse supérieur;
3. Normaliser les valeurs de masse de la fonction de croyance approximée.

### 3.2 Méthode externe par regroupement hiérarchique de Denœux

La méthode externe par regroupement hiérarchique de Denœux [7] fonctionne, tout comme la méthode de Tessem, de façon itérative jusqu'à l'atteinte d'un critère d'arrêt

ou de la valeur du paramètre d'approximation,  $k$ . Cette valeur  $k$  est le nombre d'éléments focaux recherché après approximation.

La boucle qui s'effectue itérativement consiste à construire la matrice de similarité entre tous les éléments focaux de la fonction de masse à approcher. Il y a ensuite regroupement des deux éléments focaux de similarité maximale par le biais d'une union des deux éléments focaux concernés. Les masses des deux éléments focaux regroupés par une union sont sommées.

### 3.3 Implémentation logicielle

L'implémentation des algorithmes d'approximation est réalisée à partir de la boîte à outils Matlab<sup>TM</sup> développée dans [8]. Cette implémentation utilise une codification de l'information particulièrement efficace, spécifiquement au fait qu'il permet d'employer les fonctions élémentaires de Matlab<sup>TM</sup>. De plus, cette implémentation aura l'avantage, par rapport au développement de [6], qu'une simple mesure de longueur du nom d'un ensemble permet d'obtenir le cardinal de cet ensemble. Cette méthode d'implémentation aura par contre le désavantage d'avoir à encoder puis à décoder les noms des ensembles afin de les lire pour avoir une signification dans le monde réel. Cette implémentation procède à l'approximation des corps d'évidences avant leur combinaison.

## 4 Résultats de simulations

### 4.1 Cas d'application pratique

La figure 1 présente sous forme de diagramme de Venn la structure de classification des allégeances possibles d'une cible. Plus exactement, on y retrouve les cinq classes de la classification du STANAG 1241, soit la classe *Ennemi* ( $\theta_3$ ), *Ami* ( $\theta_1$ ), *Neutre* ( $\theta_2$ ), *Présumé Ami*, et *Suspect* [3,4]. La figure 2 présente quant à elle les classes d'allégeances possibles pour le cas simulé ici.

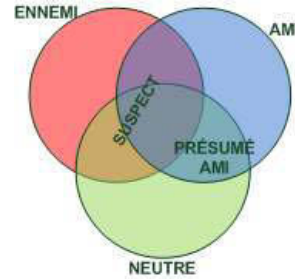


Figure 1 – Cinq classes d'allégeances possibles.

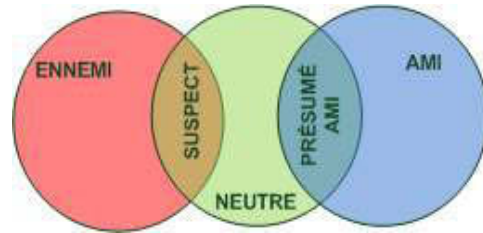


Figure 2 – Les classes d'allégeances possibles pour le cas simulé dans cette étude.

À partir des fonctions de croyances portant sur les ensembles décrits par les figures 1 et 2, nous calculons la probabilité pignistique *via* la transformation pignistique généralisée. Avec ces probabilités pignistiques, de simples opérations arithmétiques nous donneront ce que l'on désigne probabilités STANAG. :

$$\begin{aligned}
 Ami &= \{\theta_1 - \theta_1 \cap \theta_2\} \\
 Présumé Ami &= \{\theta_1 \cap \theta_2\} \\
 Neutre &= \{\theta_2 - \theta_1 \cap \theta_2 - \theta_2 \cap \theta_3\} \\
 Suspect &= \{\theta_2 \cap \theta_3\} \\
 Ennemi &= \{\theta_3 - \theta_2 \cap \theta_3\}
 \end{aligned} \tag{8}$$

Ce que l'on considèrera comme étant une bonne décision pour notre évaluation c'est une évaluation amicale (*Ami* + *Présumé Ami*) lorsque la cible est en réalité amie, et une évaluation hostile (*Ennemi* + *Suspect*) lorsque la cible est en réalité ennemie.

### 4.2 Cas de simulation typique

Notre cas de simulation typique [10], pour l'application présentée précédemment emploie comme entrée des fonctions de croyances simples où une masse de 80% est attribuée à

un singleton et le reste, 20%, est envoyée sur l'ignorance totale.

La figure 3 montre un cas typique d'une séquence d'information en provenance d'un système d'identification Ami/Ennemi que l'on combinera séquentiellement [3]. Les données sont générées aléatoirement avec une précision de 80% par rapport à la vérité qui est 'Ami' pour les 50 premières itérations, puis 'Ennemi' pour la suite. Ce changement à mi-temps simule un changement d'allégeance. Aussi, ce cas particulier comporte une séquence de contre-mesures (erreur du système) importante, soit de 4, aux temps 81-85.

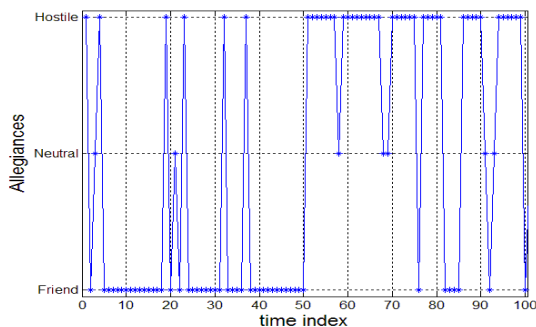


Figure 3 – Entrées à combiner en provenance d'un système d'identification Ami/Ennemi.

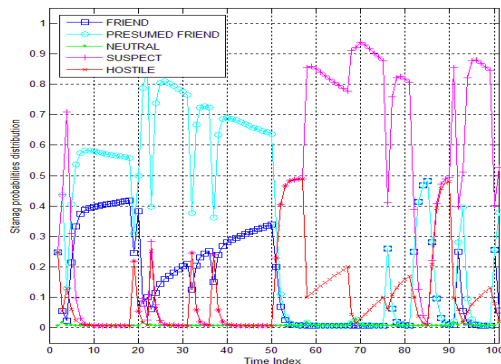


Figure 4 – Répartition des probabilités STANAG après combinaison DSsmH de la séquence d'information montrée à la figure 3.

La figure 4 montre le comportement de la méthode de combinaison DSsmH par rapport à un tel scénario. Dans [3,4] on y présente une variété de cas comparables avec différentes

règles de combinaison ainsi que leurs performances et comportement.

### 4.3 Qualité de la décision

La qualité de la décision est évaluée par le biais du taux de bonnes décisions. Cette mesure est tirée de la comparaison entre la vérité sur l'allégeance et le comportement que l'on détermine qu'elle a par le biais des probabilités STANAG, tel que défini antérieurement. Ainsi, lorsque l'on déterminera que la cible a un comportement amical (Ami + Présumé Ami) lorsqu'elle est amie en réalité, ce sera considéré comme une bonne décision.

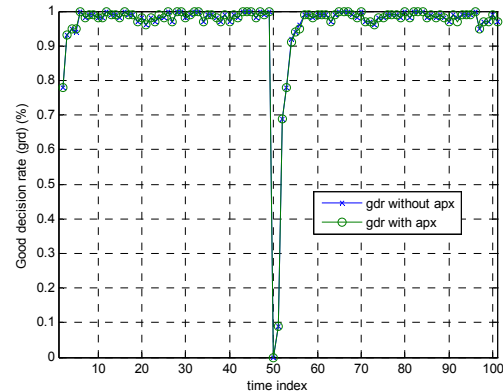


Figure 5 – Taux de bonnes décisions, pour la combinaison de Dempster-Shafer avec et sans l'approximation de Dencœux avec  $k=3$ .

Les figures 5-11 montrent pour différents cas de simulations les résultats des taux de bonnes décisions obtenus. Dans tous les cas simulés, nous obtenons des courbes de taux de bonnes décisions indiscernables entre les versions approximées et celles non approximées. Pour une même règle de combinaison, le taux de bonnes décisions pour les cas avec et sans approximation restent relativement similaires ( $DS \approx 98\%$ ,  $PCR6 \approx 94\%$ ,  $DP \approx 92\%$ ). Les oscillations dans les courbes sont présumées être dues à la nature aléatoire du processus de simulation. On peut voir cela par la dernière figure que l'on a faite pour une séquence de 400 Monte Carlo au lieu de 100, et déjà on y

perçoit un adoucissement au niveau des oscillations (voir figure 11 vs figure 10).

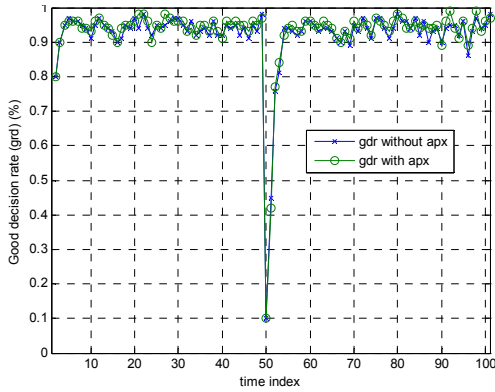


Figure 6 – Taux de bonnes décisions, pour la combinaison PCR6 sans et avec l’approximation de Denœux avec  $k=3$ .

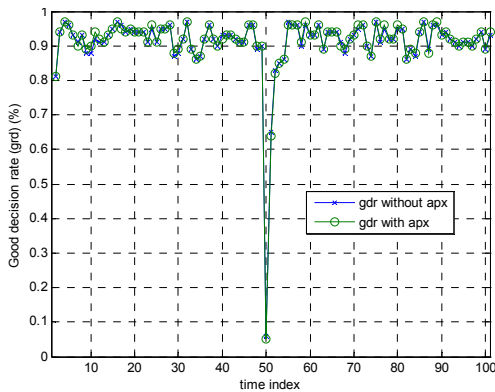


Figure 7 – Taux de bonnes décisions, pour la combinaison de Dubois-Prade sans et avec l’approximation de Denœux avec  $k=3$ .

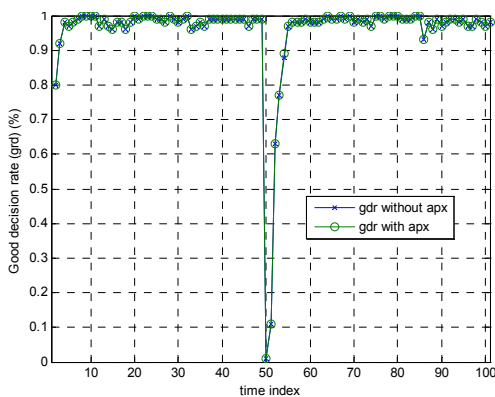


Figure 8 – Taux de bonnes décisions, pour la combinaison de Dempster-Shafer sans et avec l’approximation de Tessem avec  $klx=(5,10,0.2)$ .

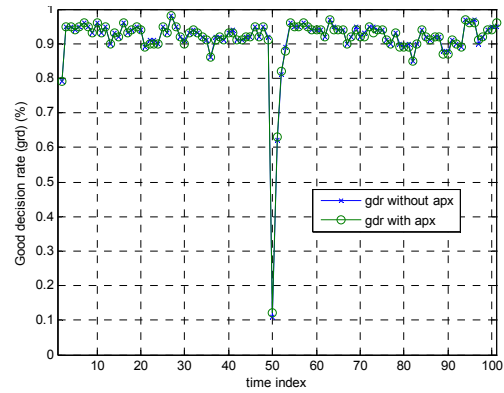


Figure 9 – Taux de bonnes décisions, pour la combinaison de Dubois-Prade sans et avec l’approximation de Tessem avec  $klx=(5,10,0.2)$ .

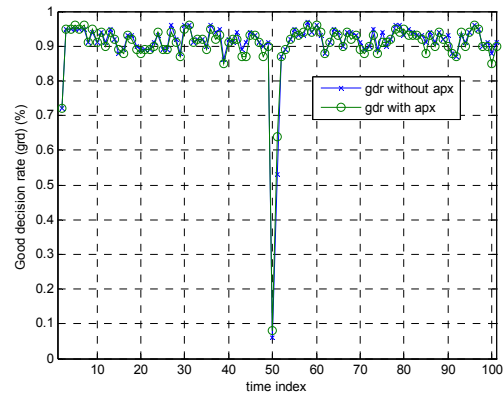


Figure 10 – Taux de bonnes décisions, pour la combinaison de Dubois-Prade sans et avec l’approximation de Tessem avec  $klx=(3,6,0.2)$ .

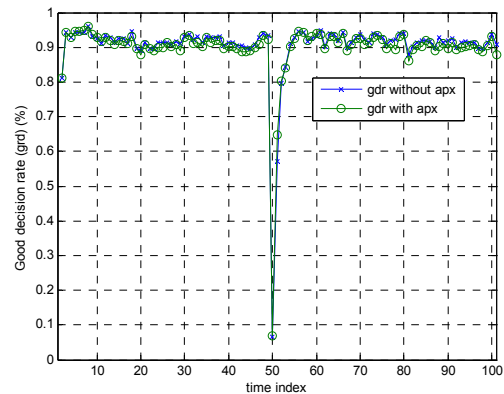


Figure 11 – Taux de bonnes décisions, pour la combinaison de Dubois-Prade sans et avec l’approximation de Tessem avec  $klx=(3,6,0.2)$  pour une Monte-Carlo de 400 simulations.

#### 4.4 Temps d'exécution

Le temps d'exécution moyen est évalué pour chaque simulation pour ensuite calculer le gain en pourcentage sur le temps d'exécution. Une valeur positive de gain représente un ralentissement dans le traitement pour la durée de la simulation, tandis qu'une valeur négative représente une baisse de la durée de la simulation et donc une amélioration en termes de vitesse d'exécution.

Le tableau 1 nous montre l'impact sur le temps d'exécution lors de l'utilisation de la méthode d'approximation de Denœux pour divers paramètres d'approximation pour les trois règles de combinaison explorées, soit DS, PCR6, puis DP. On y remarque que deux situations où l'on a un gain souhaité, c'est-à-dire pour un cas d'approximation de Denœux avec  $k=3$  et l'utilisation de la règle de combinaison DS et PCR6, autrement l'application de la méthode d'approximation de Denœux augmente le temps d'exécution.

Tableau 1 - Impact sur le temps d'exécution lors de l'approximation de Denœux.

Denœux	DS	PCR6	DP
$k=3$	-6.3%	-0.9%	+13.0%
$k=4$	+12.9%	+8.4%	+64.4%
$k=6$	+29.5%	+26.9%	+66.8%
$k=8$	+28.9%	+27.7%	+51.5%

Le tableau 2 présente l'impact sur le temps d'exécution lors de l'application de l'approximation de Tessem pour une variété de paramètres d'approximation pour la règle de combinaison DP. On y remarque peu d'impact lors des variations étudiées pour les paramètres d'approximation  $l$  et  $x$ . Par contre, lorsque  $k$  est modifié, un impact important se fait sentir. Pour une valeur de  $k=3$ , on a une réduction du temps d'exécution d'approximativement 50%. Lorsque  $k$  passe à 5, on a une réduction du temps d'exécution tournant autour de 28%. Cependant, pour une valeur de  $k=8$ , on a une augmentation du temps d'exécution de l'ordre de 10%. On prendra donc soin d'éviter une telle valeur de  $k$  pour une telle application.

Tableau 2 - Impact sur le temps d'exécution lors de l'approximation de Tessem.

DP	$x=0.2$	$x=0.2$	$x=0.4$	$x=0.4$
$k=3, l=6$	-50.5%			
$k=3, l=8$	-49.5%			
$k=3, l=10$	-49.8%			
$k=5, l=8$		-28.4%	-28.5%	
$k=5, l=10$		-27.5%	-28.3%	
$k=5, l=12$		-28.3%	-27.5%	
$k=8, l=11$				+9.4%
$k=8, l=13$				+10.0%
$k=8, l=15$				+10.5%

Le tableau 3 étudie l'impact sur le gain en termes de temps d'exécution pour les cas où l'on augmenterait le nombre de cycles de Monte-Carlo. Cela ne fait référence à rien en pratique, mais a pour but d'évaluer si nos résultats de temps sont affectés par changement dans les paramètres de simulation.

Tableau 3 - Impact sur le temps d'exécution lors de l'approximation de Tessem en fonction du nombre de cycles Monte-Carlo.

Nb. Cycles Monte-Carlo	$k/x=(3,6,0.2)$
100	-50.5%
400	-50.0%

On peut expliquer la différence de temps d'exécution de la méthode d'approximation de Denœux par rapport à celle de Tessem, du moins pour notre cas d'application, par la complexité de l'algorithme d'approximation, mais aussi par le fait qu'elle n'a pas été conçue pour cette application. L'approche de Denœux requière l'exécution d'une boucle 'while' incorporant deux doubles boucles (*For(For)*, et puis *While(While)*), alors que Tessem n'a qu'un seul 'while' implémenté. Il faudra donc un problème bien plus complexe que notre cas pour obtenir un gain par l'utilisation de l'approximation de Denœux.

## 5 Conclusions

Notre étude nous permet de tirer quelques conclusions :

- La méthode d'approximation de Dencœux permet de maintenir un bon taux de bonnes décisions. Par contre, elle accroît considérablement le temps d'exécution pour le cas à l'étude.
- La méthode d'approximation de Dencœux s'exécute plus lentement que la méthode d'approximation de Tessem.
- La méthode d'approximation de Tessem permet d'améliorer de 50% le temps d'exécution pour le cas à l'essai parmi les paramètres d'approximation testés avec de bons taux de bonnes décisions.
- Les taux de bonnes décisions obtenus avec l'utilisation d'algorithmes d'approximation sont demeurés équivalents à ceux obtenus sans approximation.
- Les courbes de taux de bonnes décisions démontrent une certaine instabilité, visible de par les oscillations dans les valeurs, qui sont dues au nombre de cycles de simulations Monte-Carlo.
- Pour l'algorithme d'approximation de Tessem, les variations dans le temps d'exécution et des taux de bonnes décisions sont minimales, lorsque les paramètres  $l$  et  $x$  sont changés.

Des questions demeurent toutefois non résolues, ce qui pourrait faire l'objet de recherches futures. Par exemple, existe-t-il une méthode d'approximation, ou de combinaison de valeur de paramètres d'approximation qui permettrait d'obtenir un meilleur taux de bonnes décisions que la combinaison sans approximation. L'exploration de la possibilité d'une implémentation plus efficace, ou plus rapide de l'algorithme d'approximation de Dencœux constitue aussi une autre voie de recherches futures.

## Références

- [1] M. Bauer, Approximation algorithms and decision making in the Dempster-Shafer theory of evidence, *IJAR*, 17, pp. 217-237, 1997.
- [2] P. Djiknavorian, and D. Grenier, *Reducing DSMT hybrid rule complexity through optimisation of the calculation algorithm*, in [12].
- [3] P. Djiknavorian, P. Valin, and D. Grenier, *Dezert-Smarandache theory applied to highly conflicting reports for identification and recognition – Illustrative example of ESM associations in dense environments*, DRDC Valcartier TR 2008-537.
- [4] P. Djiknavorian, P. Valin, and D. Grenier, *Fusion of ESM allegiance reports using DSMT*, in [13].
- [5] P. Djiknavorian, D. Grenier, P. Valin, *Klx approximation technique over Dsm Theory*, International Workshop on Belief functions 2010, Brest, France, April 2010.
- [6] A.P. Dempster, Upper and Lower probabilities induced by a multivalued mapping, *Annals of Mathematical Statistics*, 38, pp. 325-339, 1967.
- [7] T. Dencœux, Inner and outer approximation of belief structures using a hierarchical clustering approach, *Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(4), pp. 437-460, 2001.
- [8] A. Martin, *Implementing general belief function framework with a practical codification for low complexity*, in [13].
- [9] G. Shafer, *A mathematical theory of evidence*, Princeton University Press, 1976.
- [10] Simard M.A., Valin P. and Shahbazian E. *Fusion of ESM, Radar, IFF and other Attribute Information for Target Identity Estimation and a Potential Application to the Canadian Patrol Frigate*, AGARD 66th Symposium on Challenge of Future EW System Design, 18-21 October 1993, Ankara (Turkey), AGARD-CP-546, pp. 14.1-14.18.
- [11] Smarandache and J. Dezert (Editors), *Advances and Applications of DSMT for Information Fusion*, Collected works, Vol. 1, Rehoboth, U.S.A., 2004.
- [12] F. Smarandache and J. Dezert (Editors), *Advances and Applications of DSMT for Information Fusion*, Collected works, Vol. 2, Rehoboth, U.S.A., 2006.
- [13] F. Smarandache and J. Dezert (Editors), *Advances and Applications of DSMT for Information Fusion*, Collected works, Vol. 3, Rehoboth, U.S.A., 2009.
- [14] B. Tessem, Approximations for efficient computation in the theory of evidence, *Artificial Intelligence*, 61, pp. 315-329, 1993.