

Direction des bibliothèques

AVIS

Ce document a été numérisé par la Division de la gestion des documents et des archives de l'Université de Montréal.

L'auteur a autorisé l'Université de Montréal à reproduire et diffuser, en totalité ou en partie, par quelque moyen que ce soit et sur quelque support que ce soit, et exclusivement à des fins non lucratives d'enseignement et de recherche, des copies de ce mémoire ou de cette thèse.

L'auteur et les coauteurs le cas échéant conservent la propriété du droit d'auteur et des droits moraux qui protègent ce document. Ni la thèse ou le mémoire, ni des extraits substantiels de ce document, ne doivent être imprimés ou autrement reproduits sans l'autorisation de l'auteur.

Afin de se conformer à la Loi canadienne sur la protection des renseignements personnels, quelques formulaires secondaires, coordonnées ou signatures intégrées au texte ont pu être enlevés de ce document. Bien que cela ait pu affecter la pagination, il n'y a aucun contenu manquant.

NOTICE

This document was digitized by the Records Management & Archives Division of Université de Montréal.

The author of this thesis or dissertation has granted a nonexclusive license allowing Université de Montréal to reproduce and publish the document, in part or in whole, and in any format, solely for noncommercial educational and research purposes.

The author and co-authors if applicable retain copyright ownership and moral rights in this document. Neither the whole thesis or dissertation, nor substantial extracts from it, may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms, contact information or signatures may have been removed from the document. While this may affect the document page count, it does not represent any loss of content from the document.

Université de Montréal

Choix d'un associateur 2-D pour le balayage
multiple et optimisation de l'estimation des pistes

par

Francis Moreau

Département de mathématiques et de statistique

Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures

en vue de l'obtention du grade de

Maître ès sciences (M.Sc.)
en Mathématiques

janvier 2009

© Francis Moreau, 2008



0A
3
U54
2009
V-009

Université de Montréal

Faculté des études supérieures

Ce mémoire intitulé

Choix d'un associateur 2-D pour le balayage multiple et optimisation de l'estimation des pistes

présenté par

Francis Moreau

a été évalué par un jury composé des personnes suivantes :

WINTERNITZ, Pavel

(président-rapporteur)

PATERA, Jiri

(directeur de recherche)

DEMERS, Hugues

(co-directeur)

GRUNDLAND, Alfred Michel

(membre du jury)

Mémoire accepté le:

19 Janvier 2009



SOMMAIRE

Dans ce mémoire sont comparées les performances de deux algorithmes d'association linéaire à deux dimensions, l'algorithme d'enchère de Bertsekas et un algorithme de plus court chemin développé par R. Junker, T. Volgenant et D.A. Castañon. Les résultats montrent qu'il y a des avantages à utiliser l'algorithme d'enchère pour résoudre rapidement l'association sur des matrices creuses. Les tests sont ensuite étendus au problème d'association à balayage multiple. Les résultats montrent que les avantages de l'algorithme d'enchère sur les scénarios à matrices creuses sont perdus en balayage multiple. Finalement, deux approches à balayage multiple et senseur multiple qui se base sur la méthode de profondeur temporelle maximale de Kirubarajan et al. sont testées. Les deux approches diffèrent au niveau de l'estimation des pistes. L'une applique la méthode de profondeur temporelle maximale à des dimensions décroissantes, tandis que l'autre utilise la dernière association de dimension S de chaque senseur gardée en mémoire. Selon les résultats des tests, la première est plus précise alors que la seconde est plus rapide.

Mots clés : Fusion de données, Pistage, Association de données, Relaxation lagrangienne, Programmation linéaire, Association à S dimension, Algorithme d'enchère, JVC, Optimisation, Senseurs multiples.

SUMMARY

In this thesis are compared the performances of two algorithms that solve a two-dimensional linear assignment problem, the auction algorithm of Bertsekas and a shortest augmenting path algorithm developed by R. Junker, T. Volgant and D.A. Castañon. Results show that there are some advantages in using the auction algorithm to solve rapidly the two dimensional association on sparse matrices. Tests are extended to the multi-scan association problem. Results show that the advantages of the auction algorithm on sparse scenarios are lost in multi-scan. Finally, two multi-sensor/multi-scan approaches based on the maximum time depth method of Kirubarajan et al. are tested. The two approaches have a different track estimation. One applies the maximum time depth method to decreasing dimensions, while the other uses the last S -dimensional association of each sensor kept in memory. In accordance with tests results, the first is more accurate while the second is faster.

Keywords : Data Fusion, Tracking, Data association, Lagrange relaxation, Linear programming, S dimensional association, Auction algorithm, JVC, Optimization, Multi-Sensor.

TABLE DES MATIÈRES

Sommaire	iii
Summary	iv
Liste des figures	viii
Liste des tableaux	x
Glossaire	xii
Remerciements	1
Introduction	2
Chapitre 1. Problème d'association à deux dimensions	9
1.1. Présentation du problème.....	9
1.2. Revue littéraire.....	10
1.3. Algorithme d'enchère.....	14
1.3.1. Description de l'algorithme.....	14
1.3.2. L'algorithme d'enchère dans le contexte de pistage	16
1.3.2.1. Minimisation.....	16
1.3.2.2. Valeurs réelles.....	19
1.3.2.3. Terminaison de l'algorithme.....	19
1.3.2.4. Assignations impossibles.....	20
1.3.2.5. Plusieurs solutions optimales.....	23
1.4. Résultats.....	23
1.4.1. Méthodologie.....	23

1.4.2. Simulations	25
1.4.2.1. Scénario 1	25
1.4.2.2. Scénario 2	25
1.4.2.3. Scénario 3	25
1.4.3. Analyse.....	25
1.4.3.1. Tests sur matrices individuelles	25
1.4.3.2. Tests sur scénarios	26
1.5. Conclusion.....	30
Chapitre 2. Problème d’association à plusieurs dimensions	31
2.1. Présentation du problème	31
2.1.1. Revue littéraire	31
2.1.2. MHT orienté-piste.....	33
2.2. Relaxation lagrangienne.....	36
2.2.1. Théorie.....	36
2.2.1.1. Principe de base	36
2.2.1.2. Contrainte non nulle.....	37
2.2.2. Application au problème d’association à plusieurs dimensions ..	38
2.2.2.1. Relaxation des contraintes	38
2.2.2.2. Application des contraintes, mise à jour des multiplicateurs de Lagrange	39
2.3. Algorithme à balayages multiples.....	40
2.4. Expérimentations	41
2.4.1. Temps d’association.....	42
2.4.2. Qualité d’association.....	43
2.5. Conclusion.....	46
Chapitre 3. Algorithme à balayages multiples et multiple senseurs	48

3.1. Profondeur temporelle maximale	48
3.2. Estimation provisoire	51
3.2.1. Estimation par profondeur temporelle maximale.....	51
3.2.2. Estimation par l'association de dimension S	51
3.3. Résultats	54
3.3.1. Méthodologie	54
3.3.2. Analyse.....	54
3.4. Conclusion.....	55
Conclusion.....	58
Bibliographie	60
Annexe A. Résumé des recherches en balayages simples.....	A-i

LISTE DES FIGURES

0.1	Exemple d'une suite de balayages	3
0.2	Modèle récursif du pistage	4
0.3	Pistage des contacts de la figure 0.1	5
1.1	Trajectoires décrites par les cibles et la plateforme dans le scénario 1 .	26
1.2	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 5×5	27
1.3	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 10×10	27
1.4	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 50×50	28
1.5	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 100×100	28
1.6	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices ayant 10% des assignations possibles	29
2.1	Exemple d'association en $S = 4$ dimensions	31
2.2	Hypothèses générées par deux contacts après deux balayages	32
2.3	Schéma du MHT orienté-piste. Pour l'exemple, $S = 4$	34
2.4	Schéma du principe des fenêtres glissantes. Pour l'exemple, $S = 5$	35
3.1	Association à balayages multiples et à senseurs multiples utilisant la profondeur temporelle maximale	50

3.2	Estimation provisoire utilisant la dernière association à S dimensions calculée.....	52
3.3	Trajectoires décrites par les cibles et les plateformes dans le scénario 4	55

LISTE DES TABLEAUX

1.1	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur le scénario 1	29
1.2	Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur le scénario 2	30
2.1	Temps moyen en millisecondes de résolution de l'association en 3 et 4 dimensions pour le scénario 1	42
2.2	Comparaison du temps moyen de résolution de l'association entre JVC et l'algorithme d'enchère en 2-D et 3-D pour le scénario 2	42
2.3	Qualité de l'association en 2-D, 3-D et 4-D pour le scénario 1	44
2.4	Qualité de l'association en 2-D, 3-D et 4-D pour le scénario 3	46
3.1	Temps du calcul de l'estimation en 3-D et 4-D pour le scénario 1 et le scénario 4	56
3.2	Qualité de l'estimation, 3-D et 4-D pour le scénario 1	56
3.3	Qualité de l'estimation, 3-D et 4-D pour le scénario 4	56
3.4	Temps du calcul de l'estimation en senseurs multiples, en 3-D et 4-D pour le scénario 1 et le scénario 4	56
3.5	Qualité de l'estimation en senseurs multiples, en 3-D et 4-D pour le scénario 1	57
3.6	Qualité de l'estimation en senseurs multiples, en 3-D et 4-D pour le scénario 4	57

A.1 Tableau résumant quelques recherches en ordre chronologique sur le
balayage simple.....A-ii

GLOSSAIRE

Senseur : Dispositif optoélectronique de télédétection assurant le repérage de l'orientation dans l'espace d'un mobile.

RADAR : **R**adio **D**etecting **A**nd **R**anging. Système ou appareil de détection, qui émet un faisceau d'ondes électromagnétiques très courtes et reçoit l'écho, permettant ainsi de déterminer la direction et la distance d'un objet.

Balayage : Pour cet ouvrage, le terme balayage réfère à un ensemble de mesures correspondant en un tour complet du RADAR.

Balayage multiple: Ensemble composé de plus d'un balayage.

Cible : Entité physique pouvant être détectée par un senseur. Une cible peut correspondre à un avion, un navire ou tout autre véhicule.

Contact : Mesure rapportée par un senseur représentant la position d'une cible à un moment précis.

Piste : ensemble de contacts provenant de balayages différents correspondant hypothétiquement à la même cible.

Assignment : Une assignation de dimension S (ou un S -tuple) est un vecteur constitué d'un numéro de piste et de numéros de contacts. Le premier élément du vecteur est le numéro de la piste et les $S - 1$ autres sont les numéros des contacts disposés dans l'ordre chronologique de leur apparition. Dans une assignation, deux contacts ne peuvent pas provenir du même balayage. Un contact ne peut pas se trouver dans deux assignations.

Association : Une association est l'ensemble des assignations selon lesquelles les pistes et les contacts seront fusionnés.

Matrice creuse : Matrice qui contient beaucoup de zéros.

REMERCIEMENTS

Je remercie Jiri Patera, mon directeur de recherche à l'Université de Montréal et Hugues Demers, mon codirecteur de recherche chez Lockheed Martin Canada pour leur aide et leurs conseils. Je tiens à remercier particulièrement Hugues Demers pour son aide en informatique et en programmation et pour son implication dans mon intégration à Lockheed Martin Canada.

Je remercie Lockheed Martin Canada de m'avoir offert la possibilité de travailler avec eux dans le cadre de ma maîtrise. Je remercie le CRSNG, MITACS et Lockheed Martin Canada pour le financement de mon projet.

Je remercie mes collègues de travail chez Lockheed Martin pour leur conseils et leur soutien. Je tiens à remercier également parents, famille et amis pour leur soutien, leur aide et leurs encouragements.

INTRODUCTION

La fusion de données est la combinaison d'informations provenant d'une ou plusieurs sources afin d'obtenir une meilleure compréhension de la situation [1]. Les espèces vivantes les plus développées, comme les mammifères, ont en général des organes sensibles agissant comme senseurs pouvant collecter l'information provenant de cinq sources : la lumière, le son, la pression, l'odeur et le goût. Ces capacités amènent une grande possibilité de réactions autant réflexes que décisionnelles chez les plus évoluées. La perception des stimuli est identifiée à l'intégration de données tandis que les processus de sélection, de filtration, d'interprétation et de prise de décision consistent en la fusion de données.

Le but principal de la fusion de données est d'aider son utilisateur à accomplir son travail d'interprétation. Un humain utilise la machine pour augmenter sa capacité sensorielle et pour réduire la complexité des données et l'incertitude. En somme la fusion de données est utilisée dans la prise de décisions et représente les modèles, les processus et les techniques servant à consolider l'information [2].

La fusion de données intègre l'information pour la transformer, par des processus d'associations, de corrélations et de combinaisons, en un produit fusionné plus complet et précis que la simple accumulation des informations.

La fusion de données se retrouve dans plusieurs domaines tels la médecine, la robotique, le contrôle du trafic aérien, les processus manufacturiers, la modélisation environnementale, etc.

Le Joint Directors of Laboratories (JDL) a développé un modèle pour catégoriser les fonctions reliées à la fusion de données. Ce modèle décrit la fusion de

données en cinq niveaux. Ce n'est pas un modèle hiérarchique où les niveaux inférieurs alimentent les niveaux supérieurs. Les niveaux vont de 0 à 4 et se décrivent comme suit :

Niveau 0 – Évaluation des sous-objets : Estimation et prédiction des états d'un signal/objet observable sur la base d'une association de données de niveau pixel/signal ;

Niveau 1 – Évaluation des Objets : Estimation et prédiction des états d'une entité sur la base d'association, d'estimation d'états discrets et d'estimation d'états continus ;

Niveau 2 – Évaluation de la Situation : Estimation et prédiction des relations entre les entités ;

Niveau 3 – Évaluation des Impacts : Estimation et prédiction des effets sur une situation d'une action planifiée ou estimée des participants pour inclure les interactions des plans d'actions de multiple joueurs ;

Niveau 4 – Raffinement des Processus : Acquisition de données et processus adaptatif pour supporter les objectifs de la mission.

Le problème traité dans cette étude se situe au niveau 1 et se rapporte à la surveillance. Un ou plusieurs radars indiquent la position de cibles dans un voisinage.

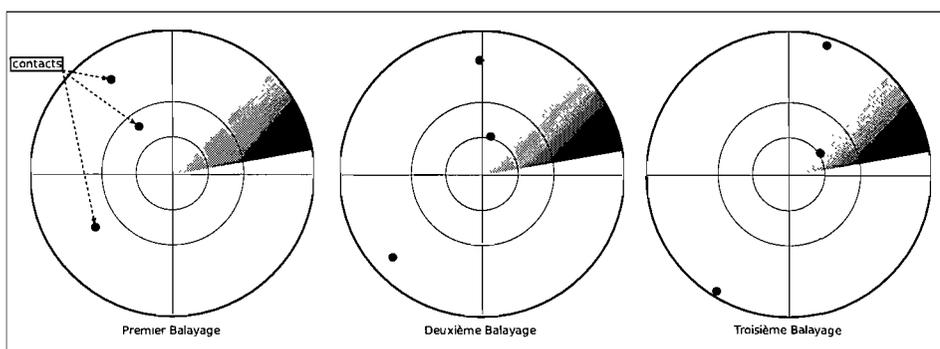


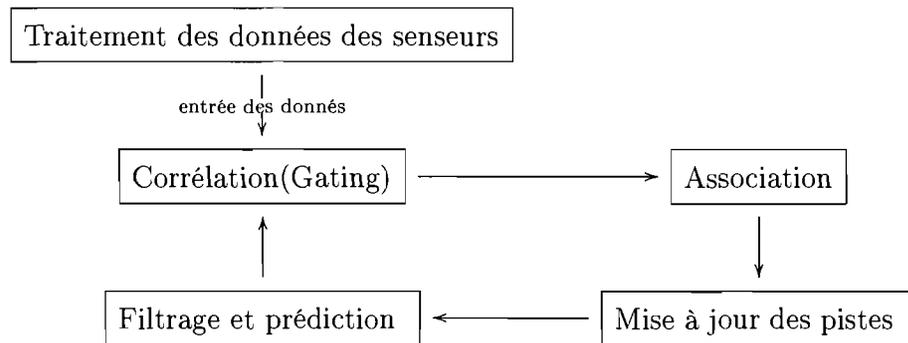
FIG. 0.1. Exemple d'une suite de balayages

La figure 0.1 montre un exemple d'une suite d'images que peut produire un radar. À partir de ces informations, la fusion de données permet d'aller au-delà du

simple positionnement des cibles. Grâce à elle, il est possible de faire le lien entre un balayage et son précédent. Autrement dit, le processus de fusion consiste à utiliser l'information positionnelle disponible pour déterminer l'association entre des objets nouvellement rapportés, appelés contacts, et des objets déjà existants, appelés pistes.

Ce type de fusion qui se situe au niveau 1 du modèle du JDL s'appelle pistage. C'est un processus récursif qui peut se représenter en cinq étapes (FIG. 0.2) :

FIG. 0.2. Modèle récursif du pistage



Traitement des données des senseurs : Dans cette étape, les données des senseurs sont transformées en mesures interprétables, par exemple : la transformation des coordonnées, le temps d'apparition d'un contact, uniformisation des unités, etc.

Corrélation : Cette étape consiste à déterminer le niveau de corrélation entre chaque paire contact-piste. Pour ce faire, le coût d'assignation d'un contact à une piste est calculé. Ce dernier se calcule par une distance statistique entre le contact et la position prédite de la piste. Certains contextes feront en sorte qu'une assignation doit être considérée impossible. Par exemple, une assignation qui requière un déplacement physiquement impossible de la cible. En général, une distance statistique maximale est déterminée pour ces cas. Si le coût d'une assignation dépasse cette limite, l'assignation en question est considérée impossible. Une piste et un contact sont dits associables quand l'assignation entre eux est possible.

Association : Cette étape utilise les informations données par la corrélation pour produire l'association contacts-pistes finale.

Mise à jour des pistes : Dans cette étape les contacts sont fusionnés aux pistes selon le résultat de l'association. Les contacts qui n'ont pas été associés à une piste en créeront de nouvelles. Si une piste n'est associée à aucun contact suite à plusieurs balayages consécutifs, elle est éliminée.

Filtrage et prédiction : Méthode pour estimer les mesures cinématiques présentes et futures d'une cible telle que la position, la vitesse et l'accélération. Cette méthode consiste à produire une courbe de régression à partir des mesures passées. Plusieurs méthodes existent : discrètes ou continues, avec ou sans contraintes, etc. La plus utilisée est le filtre de Kalman qui a l'avantage de produire une matrice de covariance pouvant être utilisée durant la corrélation [3].

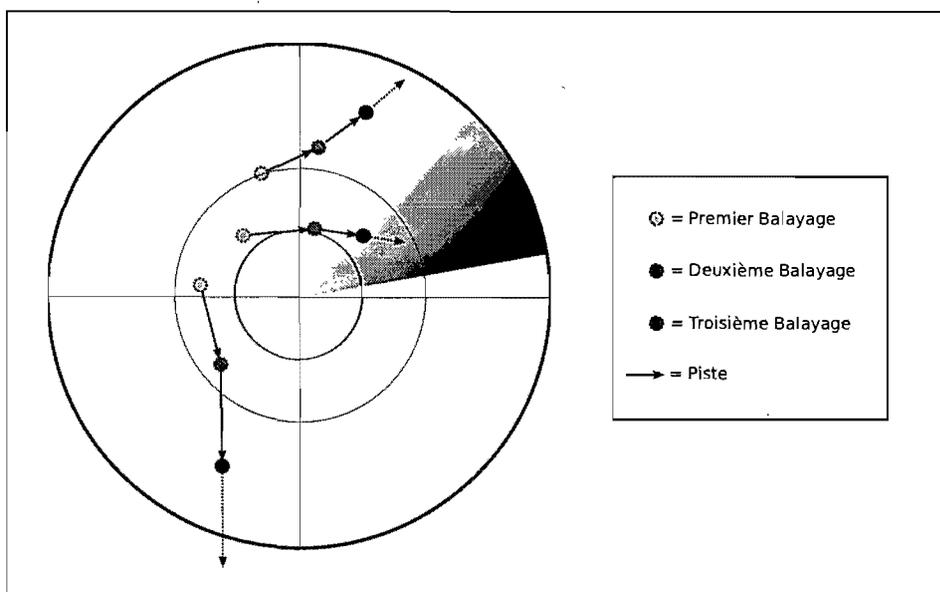


FIG. 0.3. Pistage des contacts de la figure 0.1

Le coeur du pistage se trouve à l'étape de l'association. C'est à cette étape que la décision de fusionner les contacts aux pistes se prend. Une mauvaise décision à cette étape peut influencer, voire fausser les décisions subséquentes. C'est également à cette étape que les calculs sont potentiellement les plus élaborés risquant ainsi d'augmenter le temps de calcul de tout un algorithme de pistage.

Les recherches dans ce domaine ont pour but de trouver des méthodes pouvant améliorer la fiabilité de l'association, de réduire le nombre de calculs nécessaires et de trouver le meilleur compromis entre les deux.

Plusieurs techniques d'associations pour pistage existent et sont généralement classées en deux catégories :

- (1) balayages simples et
- (2) balayages multiples.

Les algorithmes à balayages simples déterminent l'association entre les pistes et les contacts rapportés par un nouveau balayage.

D'un autre côté, les algorithmes à balayages multiples accumulent plusieurs balayages avant de déterminer l'association la plus appropriée. Une fois que l'association la plus appropriée a été déterminée, les algorithmes à balayages multiples doivent estimer provisoirement toutes les pistes. Ceci se fait généralement en considérant les balayages récents sans toutefois conclure définitivement sur l'association finale. En d'autres mots, il s'agit d'utiliser toute l'information disponible pour formuler la meilleure estimation provisoire des pistes. En principe, ces algorithmes sont plus robustes et ont un moins haut risque de mauvaises associations que ceux à balayages simples car plus d'informations sont considérées sur une plus grande période de temps.

L'association en balayage simple se fait à l'aide d'algorithmes d'associations à deux dimensions. En balayage multiple, des algorithmes d'associations à plusieurs dimensions sont utilisés. L'algorithme à plusieurs dimensions utilisé dans cette recherche transforme le problème à plusieurs dimensions en une série de problèmes à deux dimensions.

Le but de cette recherche est donc d'optimiser un algorithme à balayages multiples et à senseurs multiples par deux moyens :

- (1) En choisissant le meilleur algorithme d'association linéaire à deux dimensions.
- (2) En utilisant une méthode d'estimation sans calcul pour produire une estimation provisoire des pistes se basant sur l'association à plusieurs dimensions calculée précédemment.

La recherche est présentée en trois chapitres :

– *Chapitre 1*

Le premier chapitre porte sur les algorithmes d'association à deux dimensions et leur application au pistage à balayages simples. L'algorithme d'enchère de Bertsekas et JVC, un algorithme de plus court chemin développé par R. Junker, T. Volgenant et D.A. Castañon sont les principaux à l'étude. L'intérêt est porté sur la rapidité d'exécution des algorithmes et les résultats montrent que l'algorithme d'enchère est plus rapide avec des matrices creuses de dimensions variants de 25×25 à 200×200 .

– *Chapitre 2*

Le deuxième chapitre traite de l'association à balayages multiples. La méthode utilisée, proposé par Deb et al. utilise la relaxation lagrangienne pour convertir un problème de dimension S en une série de problèmes à deux dimensions. Les pistes et les contacts sont fusionnés selon le principe des fenêtres glissantes. Les tests portent sur la rapidité d'exécution entre l'algorithme d'enchère et l'algorithme de Junker, Volgenant et Castañon appliqués à l'algorithme de Deb et al. Les résultats montrent que l'algorithme d'enchère n'est pas avantageux dans ce contexte comparativement à l'algorithme de Junker, Volgenant et Castañon.

– *Chapitre 3*

Dans le troisième chapitre, le balayage multiple est testé dans un contexte où plusieurs radars collectent l'information. Avec le principe des fenêtres glissantes, seulement le premier balayage est fusionné à la piste. Donc, une estimation doit être faite pour les balayages restants. Deux techniques d'association à balayages multiples et senseurs multiples sont testées. Plutôt que de se baser sur l'approche standard qui considère les $S - 1$ derniers balayages parmi tous les senseurs, les deux techniques se basent sur la méthode de profondeur temporelle maximale de Kirubarajan et al. qui exécute l'association sur les balayages du senseur qui en a accumulés $S - 1$. Pour l'estimation, l'une applique la méthode de profondeur temporelle maximale

à des dimensions décroissantes, tandis que l'autre utilise la dernière association de dimension S de chaque senseur gardé en mémoire. Les résultats montrent que le temps d'exécution de la deuxième méthode est très inférieur à la première sans qu'il y ait une importante différence de la qualité de l'association.

Chapitre 1

PROBLÈME D'ASSOCIATION À DEUX DIMENSIONS

1.1. PRÉSENTATION DU PROBLÈME

Le problème à deux dimensions consiste en une association de contacts d'un balayage à des pistes. À chaque contact j , est attribué une probabilité d'appartenir à une piste i qui se traduit comme le coût d'assignation c_{ij} . Dans le déroulement normal d'un scénario, un balayage donne un ensemble de contacts qui doivent être simultanément associés aux pistes déjà existantes. Le but est de déterminer l'association contacts-pistes de telle sorte que la somme des coûts d'assignations soit minimale. Un contact ne doit être associé qu'à une seule piste et une piste ne doit être associée qu'à un seul contact. Mathématiquement, le problème d'association à deux dimensions dans un contexte où il y a n pistes et m contacts se formule comme suit :

$$\min_{p_{ij}} \sum_{i=0}^n \sum_{j=0}^m c_{ij} p_{ij} \quad (1.1.1)$$

soumis aux contraintes :

$$\begin{aligned} \sum_{j=0}^m p_{ij} &= 1; & i &= 1, 2, \dots, n \\ \sum_{i=0}^n p_{ij} &= 1; & j &= 1, 2, \dots, m \end{aligned} \quad (1.1.2)$$

où p_{ij} est une variable binaire valant 1 si le couple ij fait partie de la solution et 0 sinon.

Il s'agit d'un problème d'association linéaire à deux dimensions avec coûts, dans lequel une matrice est utilisée pour représenter les assignations. Chaque entrée de la matrice correspond au coût de l'assignation ij (piste i au contact j ; $i = 1, \dots, n$; $j = 1, \dots, m$).

Exemple 1.1.1. *Supposons un problème où il y a $n = 2$ pistes et $m = 3$ contacts. La matrice d'association est :*

$$\begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{pmatrix}$$

Il existe six solutions qui respectent la contrainte (1.1.2) :

$$\{(1, 1), (2, 2)\}, \{(1, 1), (2, 3)\}, \{(1, 2), (2, 1)\}, \\ \{(1, 2), (2, 3)\}, \{(1, 3), (2, 1)\}, \{(1, 3), (2, 2)\}$$

Parmi ces solutions, il y en a une qui minimise le coût d'association exprimé par :

$$c = c_{ij} + c_{kl} \quad \text{où } i \neq k, j \neq l$$

1.2. REVUE LITTÉRAIRE

La façon intuitive de résoudre ce problème est de vérifier le coût de tous les ensembles de couples et de choisir l'ensemble offrant le coût minimal. L'algorithme fondé sur ce principe, qui sera nommé l'algorithme de force brute, est de complexité $O(m!/(m-n)!)$ pour une association de m objets à n autres objets où $m \geq n$. Cette méthode est computationnellement peu intéressante car beaucoup trop complexe. Depuis le milieu du 20^{ième} siècle, plusieurs algorithmes ont été développés dans le but de résoudre ce problème de manière moins complexe et toujours optimale. Ces méthodes s'avèrent efficaces par le fait qu'il n'est nul besoin de calculer la valeur de toutes les associations possibles pour trouver la meilleure. Un tableau en annexe résume quelques études importantes dans ce domaine. Le premier algorithme fut la méthode du simplexe créée par George Dantzig en 1947 [4]. Par la suite est apparu en 1955 l'algorithme hongrois de Kuhn [5]. Avec les années, plusieurs algorithmes ont été développés s'inspirant de ces deux méthodes, séparant ainsi les algorithmes en deux branches principales :

- (1) méthodes duales et

(2) les descendants de la méthode du simplex.

Plusieurs méthodes duales sont des formes dérivées de l'algorithme hongrois de Kuhn. Conséquemment, leur complexité est inférieure à $O(n^4)$ pour une association $n \times n$, la complexité de l'algorithme hongrois original. Des améliorations et expériences répétées sur la méthode hongroise ont donné naissance à de nouveaux algorithmes capables de trouver une solution à l'intérieur de $O(n^3)$ opérations [6]. Klee et Minty ont réussi à montrer que la méthode du simplex est d'ordre exponentiel [7]. Pour un certain n , une méthode d'ordre exponentiel prend plus d'opérations pour s'exécuter qu'une méthode d'ordre polynômial. Ceci est appuyé par des études [8, 9] qui révèlent que les méthodes duales sont plus efficaces que celles basées sur la méthode du simplex. Les recherches dans ce domaine se sont faites en particulier pour tester les différents algorithmes afin de vérifier lesquels sont les plus efficaces en termes de rapidité d'exécution, d'acuité du résultat et d'utilisation de mémoire.

Le principal algorithme d'intérêt pour cette recherche apparaît en 1981 dans un article écrit par D.P. Bertsekas [10]. Celui-ci propose de le nommer algorithme d'enchère pour sa similitude avec les enchères réelles. Dans son article, Bertsekas applique son algorithme sur des matrices non creuses et compare ses performances avec l'algorithme hongrois et une version hybride de l'algorithme d'enchère et de l'algorithme hongrois. Ses résultats, portant sur le temps d'exécution, montrent que la méthode hybride surclasse les deux autres et que la forme pure d'enchère est supérieure à l'algorithme hongrois.

Le second algorithme d'intérêt a été créé par R.Jonker et A.Volgenant publié en 1987 [6]. Leur algorithme, qui sera représenté par JV, s'inspire de la méthode hybride hongrois-enchère proposée par Bertsekas. Il agit en deux phases. La première est une version de l'algorithme d'enchère et la seconde est une version modifiée de l'algorithme de Munkres[11]. L'algorithme de Munkres, présenté en 1957 par l'auteur du même nom, est une version améliorée de l'algorithme hongrois. Cette méthode a tendance à commencer lentement mais accélère lorsque des possibilités sont éliminées. D'un autre côté, l'algorithme d'enchère commence rapidement mais prend du temps à établir une solution optimale [12]. La stratégie

de JV est donc de faire quelques itérations au début avec l'algorithme d'enchère et de terminer avec l'algorithme de Munkres. Ceci donne finalement un algorithme de complexité $O(n^3)$, qui en fait une des méthodes les plus rapide pour résoudre le problème d'association linéaire à deux dimensions. Dans leur étude, JV a été appliqué sur des matrices denses et creuses, à valeurs aléatoires. Ses performances ont été comparées à l'algorithme d'enchère, l'algorithme hongrois et l'algorithme de Tomizawa [13], une version améliorée de l'algorithme hongrois. Ce sont quatre méthodes qui trouvent le coût d'association optimal. Leurs résultats montrent que pour des matrices non-creuses, leur algorithme est plus rapide que l'algorithme d'enchère, qui lui est plus rapide que la méthode de Tomizawa, qui elle est plus rapide que l'algorithme hongrois. En termes de temps utilisé, leurs résultats peuvent s'exprimer comme suit :

$$t_{JV} < t_{enchère} < t_{Tomizawa} < t_{Hongrois}$$

Pour des matrices creuses, JV est encore le meilleur algorithme.

En 1991, O.E. Drummond, D.A. Castanon et M.S. Bellovin revisitent l'algorithme de Jonker et Volgenant, mais décident de l'appliquer sur des matrices rectangulaires creuses [14]. Castanon a donc modifié l'algorithme pour lui offrir cette capacité. L'algorithme de Jonker, Volgenant et Castanon sera référé par JVC. Les performances du nouvel algorithme ont été comparées à une version de l'algorithme de Munkres [11] adaptée aux matrices creuses données par Stephens et Krupa[15] et à l'algorithme de Munkres compilé en FORTRAN et en Pascal. Les résultats, basés sur le temps d'exécution, sont en faveur de JVC. Il est également intéressant de constater dans leur étude que le choix du compilateur a une influence sur la rapidité.

Avec les années, Bertsekas écrit plusieurs articles donnant des version améliorées de l'algorithme d'enchère [10, 16, 17]. Une version définitive est donnée dans [18] avec plusieurs options d'implémentation dont, l'enchère inverse et l'échelonnage. Dans un article paru en 1993 [17], Bertsekas, Castanon et Tsaknakis confrontent plusieurs version d'enchère à JVC. Ils comparent leur vitesse par rapport au nombre d'assignations possibles. Dans la plupart des tests, la méthode la plus rapide est la combinaison normale-inverse non-échelonnée mais elle n'est

jamais comparée directement à l'algorithme d'enchère conventionnel. Les versions échelonnées sont meilleures seulement dans les cas asymétriques avec beaucoup d'assignations possibles. Les résultats montrent, en particulier, que l'algorithme d'enchère est plus efficace que JVC avec des matrices creuses.

Originellement, JV fut conçu pour le pistage de missiles balistiques intercontinentaux. Ces missiles voyagent en ligne droite, à vitesse constante et très éloignés les uns des autres, donc sont des cibles plutôt faciles à suivre. Donald B. Malkoff a vérifié en 1997 [12] les capacités de JVC avec des scénarios en temps réel où les cibles sont plus mobiles et plus rapprochées, pouvant s'apparenter à des cibles pilotées. La conclusion est que JVC est un algorithme efficace même dans des scénarii difficiles. Il considère l'erreur due aux limitations des composantes physiques de l'ordinateur et les limites de temps. Il compare les performances d'un algorithme qu'il a conçu lui-même qu'il réfère par l'algorithme de compétition, à JVC et à la méthode du plus proche voisin. L'algorithme de compétition et du plus proche voisin ne sont pas des méthodes qui donnent une solution optimale, mais comme le temps de calcul est limité, JVC ne se termine pas nécessairement, donc peut lui aussi donner une solution sous-optimale. Dans les expériences de Malkoff, les coûts d'assignations sont calculés de trois façons différentes :

- (1) par une distance spatiale ;
- (2) par une distance spatiale et la cinématique ;
- (3) par une distance spatiale, la cinématique et l'identification des cibles.

Ses résultats révèlent qu'en général l'erreur diminue avec la quantité d'informations. Ils révèlent également qu'il y a moins d'erreurs avec l'algorithme de compétition qu'avec JVC dans les scénarios difficiles, mais l'erreur est presque nulle dans les deux cas lorsqu'on ajoute l'identification des cibles.

En 2000, Levedahl écrit un article très exhaustif [19]. Il compare 19 versions de JVC, l'algorithme de force brute, du plus proche voisin, de Munkres et l'algorithme d'enchère en normal ou normal-inverse, avec ou sans échelonnage. Ils ont été compilés en C, Fortran ou Matlab. Les algorithmes sont testés sur des matrices creuses et non-creuses à entrées réelles ou entières. Les résultats montrent que JVC est plus rapide que l'algorithme d'enchère sauf pour des matrices creuses

quand les algorithmes sont conçus en Matlab. Les deux sont cependant beaucoup plus rapides que l'algorithme de Munkres et de force brute. L'enchère inverse n'a pas démontré une amélioration significative et l'échelonnage n'est efficace qu'avec des matrices très denses. Les résultats montrent également que le compilateur peut influencer la vitesse d'un algorithme.

1.3. ALGORITHME D'ENCHÈRE

Les scénarios de pistage réels montrent que les matrices d'associations produites sont généralement creuses. Un algorithme optimisé à traiter les matrices creuses est donc souhaitable. Dans ce domaine, l'algorithme d'enchère de Bertsekas [18] se présente comme un bon choix. Bertsekas a montré que son algorithme gagne en efficacité quand les matrices deviennent creuses, le rendant meilleur que JVC [17].

1.3.1. Description de l'algorithme

L'algorithme d'enchère, comme son nom l'indique, est une simulation d'un encan où des acheteurs se livrent à une compétition dans le but d'acquérir l'objet qui leur sera le plus rentable à cette différence près du cas réel que le choix de chacun est celui qui optimise le profit global. Autrement dit, représentant cette situation par une matrice où les lignes sont les acheteurs, les colonnes sont les objets et chaque entrée est la valeur d'un objet pour un acheteur, l'algorithme d'enchère ira trouver l'association acheteurs-objets de telle sorte que la somme de la valeur de chaque assignation soit maximale.

L'algorithme considère deux ensembles. D'un côté, un ensemble d'acheteurs et de l'autre un ensemble d'objets proposés. Les objets possèdent un prix initial mais, ils ont tous pour chaque acheteur une valeur particulière. Au premier tour, les acheteurs vérifient le profit net relatif à chaque objet. Si c_{ij} est la valeur de l'objet j pour l'acheteur i et p_j est le prix de l'objet j , le profit net est ;

$$\text{Profit}_{ij} = c_{ij} - p_j$$

et l'objet convoité est celui qui maximise ce profit. Une fois celui-ci sélectionné, l'acheteur doit le rendre moins intéressant aux yeux de ses compétiteurs donc,

l'enchère au maximum, c'est-à-dire de sorte que son profit net soit égal à celui qu'offre le deuxième objet le plus profitable. Si k est ce deuxième choix et que j est l'objet désiré, le prix proposé de j sera d'une enchère ;

$$\text{Enchère}_j = (c_{ij} - p_j) - (c_{ik} - p_k)$$

Une fois que tous les acheteurs ont choisi un objet et proposé un nouveau prix, les objets sont assignés aux acheteurs qui en ont proposé le prix le plus élevé. Le prix de ces objets est alors incrémenté par la valeur de l'enchère proposée par l'acheteur qui lui est assigné.

$$p_j \longrightarrow p_j + \text{Enchère}_j$$

Ceux-ci deviennent donc les acquéreurs potentiels de ces objets, ne miseront plus sur d'autres et perdront leur main mise seulement si un compétiteur en propose un prix plus élevé lors d'un tour subséquent. Le processus recommence pour les perdants. Chaque tour se déroule de la même façon excepté que ce ne sont que les acheteurs sans objet qui enchérissent. L'algorithme se termine lorsque chaque acheteur a un objet.

Le danger de cet algorithme est qu'il pourrait y avoir un grand nombre d'acheteurs qui se disputent un petit nombre d'objets mais les enchérissent chacun de 0, générant ainsi une boucle infinie.

Exemple 1.3.1. *Les matrices suivantes, où α et β sont des entiers, sont problématiques :*

$$\begin{pmatrix} \alpha & \alpha \\ \beta & \beta \end{pmatrix}, \begin{pmatrix} \alpha & \alpha \\ \alpha & \alpha \end{pmatrix}$$

Dans la première matrice, les deux acheteurs miseront sur le premier objet avec une enchère de $(\alpha - 0) - (\alpha - 0) = 0$ pour le premier acheteur et une enchère de $(\beta - 0) - (\beta - 0) = 0$ pour le deuxième acheteur. Le prix du premier objet n'augmente pas. L'acheteur libre misera donc toujours sur cet objet en prenant la place du propriétaire. Ce comportement se répétera à l'infini. L'algorithme d'enchère présente un comportement similaire avec la deuxième matrice.

La manière de palier à ce problème est d'imposer une mise minimale. Bertsekas propose qu'une association respecte la condition ϵ -complementary slackness (ϵ -CS) si

$$c_{ij_i} - p_{j_i} \geq \max_j \{c_{ij} - p_j\} + \epsilon$$

pour chaque pair (i, j_i) associée, où j_i est l'indice de l'objet associé à l'acheteur i et ϵ représente la mise minimale. Donc maintenant, posons ;

$$\text{Enchère}_j = (c_{ij} - p_j) - (c_{ik} - p_k) + \epsilon$$

Cette formule respecte la condition ϵ -CS. En général, plus ϵ est grand, plus l'issue de la vente se décide rapidement, mais avec une plus grande possibilité que l'association soit sous-optimale.

Il fut démontré [18] qu'il existe une valeur optimale pour ϵ .

Théorème 1.3.1. *Pour des valeurs de c_{ij} entières, l'association $N \times N$ sera optimale si*

$$\epsilon < \frac{1}{N}$$

1.3.2. L'algorithme d'enchère dans le contexte de pistage

Dans cette section il est question des modifications nécessaires à l'algorithme d'enchère lorsqu'appliqué au problème d'association contacts-pistes.

1.3.2.1. Minimisation

Durant la phase de corrélation, la probabilité d'assignation pour chaque couple est calculée. Supposant une distribution gaussienne, une fonction de vraisemblance proposée par Blackman[3] pour calculer le coût de l'assignation de la piste i au contact j est

$$g_{ij} = \frac{e^{-\frac{d_{ij}^2}{2}}}{(2\pi)^{M/2} \sqrt{|S_i|}} \quad (1.3.1)$$

où S_i est la matrice de covariance pour la piste i donnée par le filtre de Kalman, M est la dimension cartésienne dans laquelle le radar produit les mesures et

$$d_{ij}^2 = \tilde{y}_{ij}^T S_i^{-1} \tilde{y}_{ij}$$

\tilde{y}_{ij} = la différence entre la mesure et la prédiction.

De façon naïve, le but est de choisir l'assignation qui maximise g_{ij} . En prenant le logarithme de (1.3.1),

$$\begin{aligned} \log(g_{ij}) &= \log \left(\frac{e^{-\frac{d_{ij}^2}{2}}}{(2\pi)^{M/2} \sqrt{|S_i|}} \right) \\ &= \frac{-d_{ij}^2}{2} - \frac{M}{2} \log(2\pi) - \frac{1}{2} \log |S_i| \end{aligned}$$

la maximisation de g_{ij} est équivalent à la minimisation de

$$d_{g_{ij}}^2 = d_{ij}^2 + \log |S_i|$$

L'algorithme d'enchère est conçu à la base pour maximiser le coût. Pour minimiser le coût, il faut donc changer les calculs. D'abord, la formule du profit sera substituée par ;

$$\text{Profit}'_{ij} = c_{ij} + p_j$$

où le profit doit être minimal et si j est l'objet désiré et k le deuxième choix, l'enchère devient ;

$$\text{Enchère}'_j = (c_{ik} + p_k) - (c_{ij} + p_j) + \epsilon$$

où l'acheteur augmente le prix de l'objet j pour le rendre moins intéressant aux yeux des autres. La condition ϵ -CS est reformulée de la façon suivante ;

$$c_{ij_i} + p_{j_i} \leq \min \{c_{ij} + p_j\} + \epsilon$$

La solution reste optimale malgré ce changement. Ceci se voit par le fait que l'algorithme d'enchère fonctionne également dans le cas où la matrice contient des entrées positives et négatives. Pour une matrice entièrement négative, l'algorithme a le même comportement, à un signe près, que les nouvelles formules avec une

matrice entièrement positive. De manière plus formelle, la préservation de l'optimalité peut être démontrée en reformulant le théorème (1.3.1) pour l'association minimale avec les formules ci-haut.

Théorème 1.3.2. *Pour des valeurs de c_{ij} entières et $\epsilon > 0$, l'association $N \times N$ sera optimale si*

$$\epsilon < \frac{1}{N}$$

DÉMONSTRATION. Soit $\epsilon > 0$. i est l'indice sur les acheteurs, j est l'indice sur les objets et k_i est l'objet associé à l'acheteur i dans l'association.

$$\sum_{j=1}^N p_j = \sum_{i=1}^N p_{k_i} \quad (1.3.2)$$

car chaque objet est associé à un seul acheteur et il y a autant d'acheteurs que d'objets. Considérons le fait que :

$$\sum_{i=1}^N (c_{ik_i} + p_{k_i}) \geq \sum_{i=1}^N \min_j \{c_{ij} + p_j\} \quad (1.3.3)$$

En distribuant la somme, (1.3.3) devient :

$$\sum_{i=1}^N c_{ik_i} + \sum_{i=1}^N p_{k_i} \geq \sum_{i=1}^N \min_j \{c_{ij} + p_j\} \quad (1.3.4)$$

Par (1.3.2) :

$$\sum_{i=1}^N c_{ik_i} + \sum_{j=1}^N p_j \geq \sum_{i=1}^N \min_j \{c_{ij} + p_j\} \quad (1.3.5)$$

donc, le profit global pour n'importe quelle association $\{(i, k_i) | i = 1, \dots, N\}$ satisfait ;

$$\sum_{i=1}^N c_{ik_i} \geq \sum_{i=1}^N \min_j \{c_{ij} + p_j\} - \sum_{j=1}^N p_j \quad (1.3.6)$$

Posons maintenant ;

$$A^* = \min_{\substack{i=1, \dots, N \\ k_i \neq k_j \text{ si } i \neq j}} \sum_{i=1}^N c_{ik_i} \quad \text{et} \quad D^* = \max_{p_j} \left\{ \sum_{i=1}^N \min_j \{c_{ij} + p_j\} - \sum_{j=1}^N p_j \right\}.$$

A^* est la solution optimale du problème tandis que D^* peut être vue comme le problème dual calculé par l'algorithme d'enchère. D^* doit donc être égal à A^* .

Par (1.3.6) ;

$$A^* \geq D^* \quad (1.3.7)$$

Comme l'association donnée $\{(i, j_i) | i = 1, \dots, N\}$ satisfait la condition ϵ -CS avec l'ensemble de prix $\{\bar{p}_j | j = 1, \dots, N\}$ alors ;

$$\min_j \{c_{ij} + \bar{p}_j\} + \epsilon \geq c_{ij_i} + \bar{p}_{j_i}.$$

En étendant cette relation à tous les i , nous obtenons ;

$$\begin{aligned} D^* &\geq \sum_{i=1}^N \left(\min_j \{c_{ij} + \bar{p}_j\} - \bar{p}_{j_i} \right) \\ &\geq \sum_{i=1}^N \left((c_{ij_i} + \bar{p}_{j_i} - \epsilon) - \bar{p}_{j_i} \right) \\ &= \sum_{i=1}^N c_{ij_i} - \epsilon N \\ &\geq A^* - \epsilon N \end{aligned} \quad (1.3.8)$$

D^* et A^* sont entiers en combinant (1.3.7) et (1.3.8) :

$$A^* \geq D^* \geq A^* - \epsilon N,$$

donc en choisissant $\epsilon < 1/N$, $A^* = D^*$. □

1.3.2.2. Valeurs réelles

Les valeurs c_{ij} ne sont pas entières, donc le théorème (1.3.2) ne s'applique pas. Pour arriver à des entiers, il est possible de tronquer les valeurs de la matrice à partir de la $(k + 1)^{\text{ième}}$ décimale et les multiplier par 10^k ou diviser ϵ par 10^k . Il suffit de choisir un k pour toujours obtenir des associations optimales. Pour cela, avec $k = 3$ les associations finales étaient toujours optimales dans les tests.

1.3.2.3. Terminaison de l'algorithme

L'algorithme se termine lorsque tous les acheteurs ont un objet. Donc, s'il y a plus d'acheteurs que d'objets, l'algorithme ne se terminera jamais. Alors, pour régler le problème, s'il y a plus de pistes que de contacts, les contacts seront les acheteurs et s'il y a plus de contacts, les pistes seront les acheteurs.

1.3.2.4. Assignations impossibles

Lors de la phase de corrélation, si le coût d'une assignation dépasse un seuil ψ prédéterminé, cette assignation est considérée impossible. Une valeur α peut être utilisée pour représenter le coût de ces assignations. Cette valeur doit être choisie suffisamment grande pour ne pas influencer le choix de la solution optimale. Dans les principes de base de l'association, l'association finale ne doit contenir des assignations impossibles que si c'est le seul choix.

Exemple 1.3.2. Avec la matrice

$$\begin{pmatrix} \alpha & \alpha & \alpha \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

où les lignes sont les acheteurs et les colonnes sont les objets, il n'y a pas d'autres choix d'associer l'acheteur 1 à un objet où l'assignation est impossible.

La valeur α doit être plus grande que deux fois la valeur maximale pour une assignation possible.

$$\alpha > 2\psi$$

Pour comprendre pourquoi, considérons l'exemple suivant :

Exemple 1.3.3. Avec la matrice

$$\begin{pmatrix} \psi & \alpha \\ 0 & \psi \end{pmatrix}$$

l'association qui doit être choisie est $\{(1,1), (2,2)\}$, qui vaut 2ψ , car l'assignation $(1,2)$ est impossible. Si la valeur α est plus petite que 2ψ , l'association optimale donnée par l'algorithme sera $\{(1,2), (2,1)\}$ valant α .

À la base, l'algorithme d'enchère considère toutes les entrées de la matrice, il ne différencie pas les assignations possibles et impossibles. S'il y a beaucoup d'assignations impossibles, beaucoup de temps d'exécution pourra être gagné si celles-ci sont exclues des calculs.

Exemple 1.3.4. Soit les matrices suivantes :

$$\begin{pmatrix} \alpha & \alpha & \alpha \\ \alpha & \alpha & \alpha \\ \alpha & \alpha & x \end{pmatrix}, \begin{pmatrix} x & \alpha & \alpha \\ \alpha & y & \alpha \\ \alpha & \alpha & z \end{pmatrix}$$

où α est la valeur représentant l'impossibilité d'assignation et $x, y, z \in [0, 100] \cap \mathbb{R}$. L'association optimale dans la première matrice est $(3, 3)$ mais l'algorithme fait des calculs inutiles pour la déterminer. Les contacts un et deux ne s'associent à aucune piste mais il faut tout de même effectuer un processus d'enchère pour ces contacts. Dans la deuxième, l'association optimale est $(1, 1), (2, 2), (3, 3)$ mais dans le processus de l'algorithme, pour chaque acheteur, les coût d'assignations de trois objets doivent être comparés quand seulement un est possible.

Il serait donc intéressant de pouvoir transformer l'algorithme de manière à ce que chaque acheteur ne considère que les objets qui lui sont associables et que s'il lui est impossible d'en avoir un, qu'il se retire de la course. Tout cela est possible grâce à un comportement de l'algorithme d'enchère décrit dans la proposition qui suit. Pour bien comprendre cette proposition, rappelons que dès qu'une mise est proposée pour un objet, celui-ci sera toujours associé. Les prix initiaux sont nuls, la matrice est de dimension $N \times N$ à valeurs entières et que $\epsilon < 1/N$.

Théorème 1.3.3. Si, lors de l'exécution de l'algorithme, l'acheteur i mise sur l'objet j tel que $c_{ij} \geq c_{ik} \forall k$, alors il ne misera jamais plus sur un objet k tel que $c_{ij} > c_{ik}$.

DÉMONSTRATION. Considerons l'acheteur i . Nous faisons la preuve par induction où n sera le nombre de fois qu'apparaît un objet j tel que $c_{ij} = c_{ik} \forall k, p_j = 0$. Supposons que l'acheteur i mise sur un de ces objets.

Le prix doit être nul, sinon il y aurait un autre acheteur qui aurait misé sur cet objet auparavant. Donc, il existerait un objet k tel que $p_k = 0$ parce qu'il n'y a pas plus d'acheteurs que d'objets. $c_{ij} \geq c_{ik}$ mais comme l'acheteur i a misé sur l'objet j alors $c_{ik} + p_k \geq c_{ij} + p_j$ car l'acheteur veut minimiser le coût. On aurait donc une contradiction.

Si $n = 1$ et que l'acheteur i mise sur l'objet j tel que c_{ij} est maximal et $p_j = 0$ alors c'est parce que le prix de tous les autres objets a augmenté de telle sorte que celui-là offre un profit $c_{ij} + p_j$ minimal. Donc les autres objets ont tous reçu une mise, donc sont tous associés. L'acheteur en question est alors le seul non-associé parce qu'il y a autant d'objets que d'acheteurs. L'algorithme se termine avec cette association.

Si $n = 2$ et que l'acheteur i mise sur un objet j tel que c_{ij} est maximal, alors c'est que $n - 2$ acheteurs sont associés. Il y a alors deux possibilités. Soit l'acheteur garde son choix jusqu'à la fin, soit il le perd. S'il le garde, le théorème est vrai. S'il le perd, il devient le seul non-associé et il n'y aura qu'un seul c_{ij} maximal avec $p_j = 0$. On revient donc au cas $n = 1$.

Si $n = m$ et que l'acheteur i mise sur un objet j tel que c_{ij} est maximal, alors c'est que $n - m$ acheteurs sont associés. S'il perd son choix au cours de l'algorithme, il y aura alors au moins $n - (m - 1)$ acheteurs d'associés et on reviendra au cas $n = m - 1$ ou à un cas antérieur. \square

Grâce à ce théorème, il est maintenant possible de modifier l'algorithme comme suit. Les acheteurs ne considèrent que les objets qui leur sont associables, les autres étant ignorés. On désigne une valeur α qui joue le rôle de valeur d'impossibilité d'assignation. L'acheteur ne mise pas sur les objets qui lui sont associables qui offrent un profit $c_{ij} + p_j$ supérieur à α et si c'est le cas pour tous les objets qui lui sont associables, l'acheteur se retire de l'enchère. S'il n'y a qu'un seul objet avec lequel un acheteur est associable, l'enchère de cet acheteur pour cet objet est de ;

$$\alpha - (c_{ij} + p_j) + \epsilon$$

Et finalement si aucun objet ne peut être associé à un acheteur, cet acheteur est éliminé dès le début.

Quand un acheteur se retire, il n'enchère pas le prix. Donc, il y a peut-être un risque de changer l'association finale. Vérifions que l'optimalité est préservée. Considérons l'acheteur i . Soit n le nombre d'objets j tel que $c_{ij} \geq c_{ik} \forall k$ et $p_j = 0$. Si $n = 1$ et que l'acheteur mise sur cet objet, alors tous les autres objets sont associés et l'algorithme se termine avec cette association. L'optimalité n'est donc

pas affectée dans ce cas. Si $n \geq 2$ et que l'acheteur choisit un de ces objets, alors l'acheteur ne participera plus à l'enchère. Par conséquent, le prix de l'objet n'est pas incrémenté. Mais comme $n \geq 2$, l'incrément aurait été de ϵ . Dans ce cas l'association respecte la condition ϵ -CS. Alors un incrément de ϵ ou de zéro ne change pas la valeur de l'association finale.

1.3.2.5. Plusieurs solutions optimales

Il est possible que plusieurs associations minimisent le profit.

Exemple 1.3.5. Soit la matrice :

$$\begin{pmatrix} a & a \\ b & b \end{pmatrix}$$

Dans ce cas, les associations $((1, 1), (2, 2))$ et $((1, 2), (2, 1))$ valent $a + b$ et sont minimales. Il y a donc deux associations optimales mais seulement une des deux décrit la réalité.

C'est un cas qui est beaucoup plus probable lorsque les matrices sont à valeurs entières que lorsqu'elles ont valeurs rationnelles ou réelles. Donc en tronquant les valeurs calculées lors de la corrélation, il y a un risque, quoique minime, qu'il existe deux solutions optimales.

1.4. RÉSULTATS

1.4.1. Méthodologie

L'algorithme d'enchère et JVC ont été testés sur des matrices à valeurs aléatoires et dans des simulations de surveillance maritime, le tout compilé en C++.

Les simulations sont gérées par un banc d'essais. Le banc d'essais de Lockheed Martin Canada est une architecture de test pour algorithmes de fusion de données. Il est composé d'un simulateur, d'un module de fusion, d'une interface graphique d'analyse de données et autres applications de gestions d'une simulation. Le simulateur simule des données provenant de senseurs tels que radars, Electronic Support Measures (ESM) et Interrogation Friend or Foe (IFF). Le module de fusion associe les rapports provenant des senseurs aux pistes existantes

et les fusionnent. La fusion peut être positionnelle et identitaire. L'interface graphique permet de visualiser les pistes ainsi que les rapports brutes et d'évaluer les performances de la fusion.

À la base, le banc d'essais comprend, comme associateurs, l'algorithme de plus proche voisin, un algorithme simple et JVC.

L'algorithme simple associe chaque contact à la piste qui lui offre le meilleur coût. C'est un algorithme rapide mais n'assure pas d'association une-à-une, donc n'est vraiment pas adapté au problème d'association dont il est question.

L'algorithme de plus proche voisin est une approche non-optimale qui consiste en quelques étapes :

- (1) un contact est choisi arbitrairement
- (2) ce contact est associé à la piste non visité qui offre le meilleur coût
- (3) le contact et la piste sont marqués comme étant visités
- (4) on recommence avec un autre contact non visité
- (5) l'algorithme se termine lorsqu'il n'y a plus de piste ou de contact non visité.

Cette méthode assure des associations une-à-une mais comporte un haut risque d'erreur, en particulier dans les cas de matrices où le nombre de pistes n'est pas égal au nombre de contacts.

L'algorithme d'enchère a été ajouté pour que ses performances soient comparées à celles de JVC dans un contexte où la fusion est positionnelle seulement.

Une série d'expériences a été menée sur des matrices de dimension 5×5 , 10×10 , 50×50 et 100×100 . Pour chaque dimension, le pourcentage d'assignations possibles varie de 0% à 100%. Les assignations possibles sont distribuées de façon aléatoire dans les matrices. Dans le banc d'essais, le coût d'assignation est calculé par une distance statistique. Il a été convenu que si une assignation présente un coût supérieur à 100, alors elle est considérée impossible. Pour représenter les matrices produites lors des simulations du banc d'essais, les coûts d'assignations possibles sont des nombres rationnels variant de 0 à 100 et la valeur représentant les assignations impossibles $\alpha > 200$. Le but est de trouver le coût d'association minimal.

1.4.2. Simulations

Les scénarios utilisés lors des simulations sont paramétrés selon :

- (1) le nombre de cibles à pister ;
- (2) la vitesse des cibles ;
- (3) leur direction et leur parcours ;
- (4) la proximité des cibles ;
- (5) la précision des senseurs.

1.4.2.1. Scénario 1

Le scénario 1 est composé de dix cibles se déplaçant en trois groupes. Dans un des groupes, trois cibles se déplacent en ligne droite, dans un autre, deux cibles voyagent côte-à-côte décrivant un lacet et dans un dernier, cinq cibles s'éloignent les unes des autres ou se croisent (Figure 1.1). Une plateforme mobile contenant deux senseurs passe au centre des trois groupes.

1.4.2.2. Scénario 2

Le scénario 2 est composé de 25 cibles se déplaçant en formation 5 par 5 dans un carré de 25km^2 . Les cibles se déplacent toutes dans la même direction, à la même vitesse et ne font pas de manoeuvres spécifiques.

1.4.2.3. Scénario 3

Le scénario 3 est composé de 9 cibles se déplaçant à proximité les unes des autres, dans une même direction, à la même vitesse. Pour ce scénario, le radar utilisé a une probabilité de détection de 75%.

1.4.3. Analyse

1.4.3.1. Tests sur matrices individuelles

Les graphiques 1.2 et 1.3 montrent que pour des matrices 10×10 et moins, JVC est plus rapide que l'algorithme d'enchère dans tous les cas. Cependant, l'algorithme d'enchère devient plus rapide que JVC sur des matrices 50×50 ayant

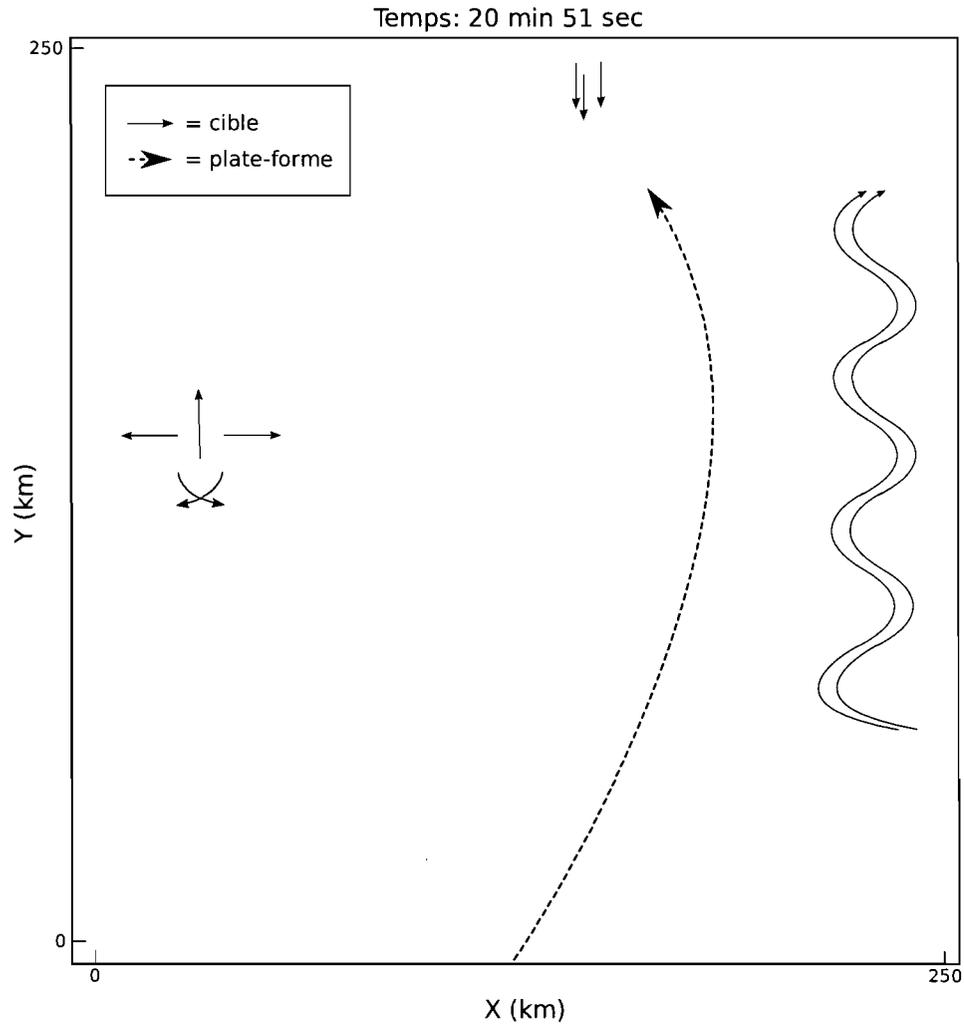


FIG. 1.1. Trajectoires décrites par les cibles et la plateforme dans le scénario 1

moins de 40% d'assignations possibles (Graphique 1.4). Le phénomène tend ensuite à s'amenuiser à mesure que la dimension des matrices augmente. Enchère est plus rapide que JVC lorsque moins de 20% des assignations sont possibles dans une matrice 100×100 (Graphique 1.5). Jusqu'à des matrices 200×200 , Enchère est plus efficace lorsque moins de 10% des assignations sont possibles (Graphique 1.6).

1.4.3.2. Tests sur scénarios

Le scénario 1 est composé de dix cibles séparées en trois groupes de moins de cinq cibles. En se référant aux tests sur les matrices aléatoires (graphique 1.3),

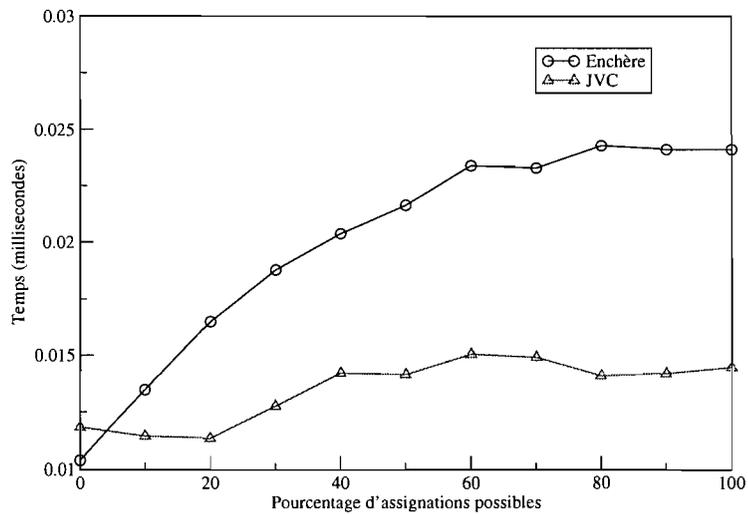


FIG. 1.2. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 5×5

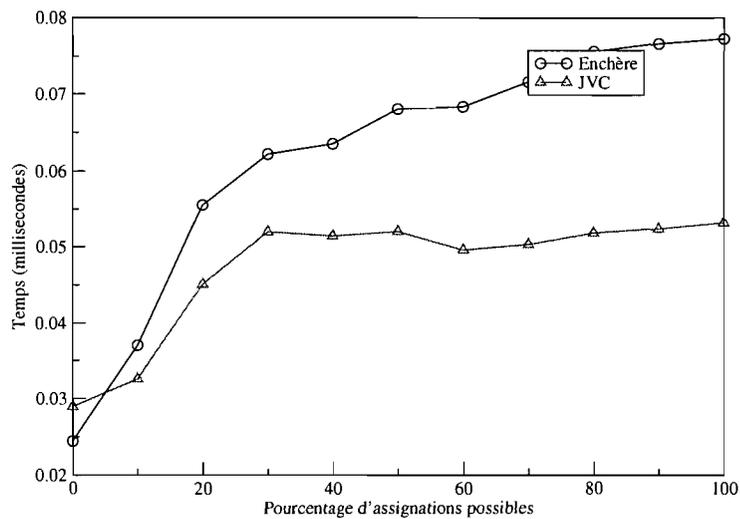


FIG. 1.3. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 10×10

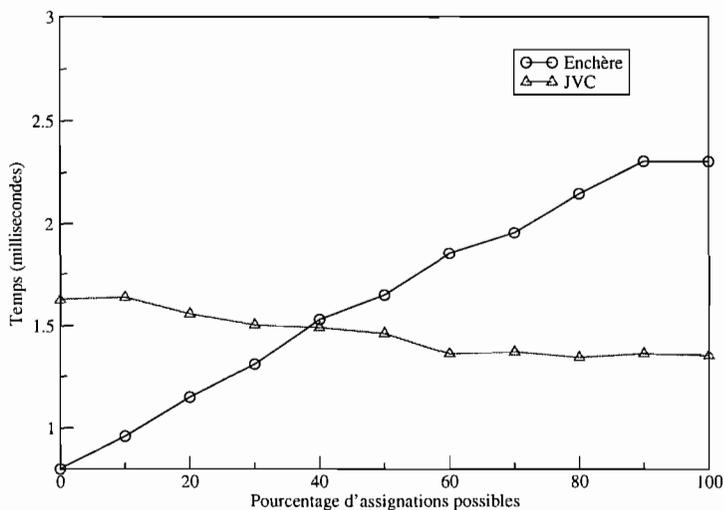


FIG. 1.4. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 50×50

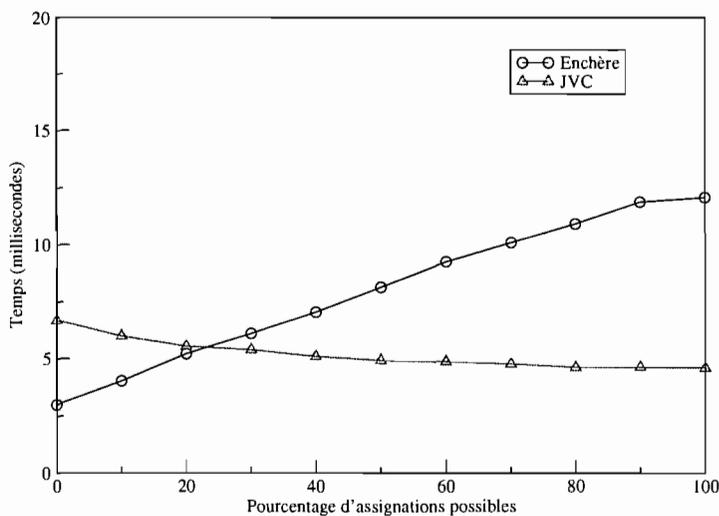


FIG. 1.5. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices 100×100

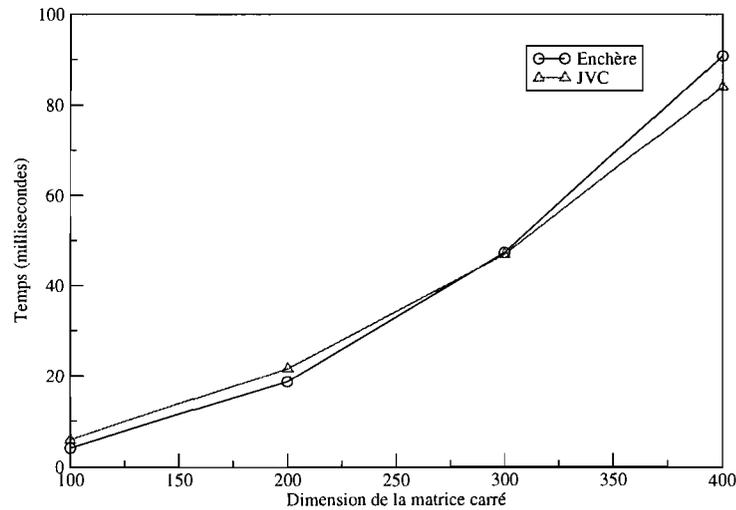


FIG. 1.6. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur matrices ayant 10% des assignments possibles

TAB. 1.1. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur le scénario 1

Algorithme	Temps (millisecondes)
JVC	0.019
Algorithme d'enchère	0.029

JVC devrait avoir l'avantage. C'est en effet ce qui est possible d'observer (tableau 1.1). JVC prend en moyenne 66,23% du temps d'exécution de l'algorithme d'enchère pour résoudre l'association.

Dans le scénario 2, les résultats (tableau 1.2) montrent que l'algorithme d'enchère est plus efficace. Le comportement peu élaboré des 25 cibles de ce scénario permet durant la corrélation d'éliminer plusieurs possibilités d'assignments. Les cibles ne changent pas de direction et elles voyagent à vitesse constante. Le filtre de Kalman a été configuré pour produire ses estimations sur une base de vitesse constante. L'estimation donnée par le filtre de Kalman est donc très fidèle aux données recueillies par le radar, ce qui facilite l'élimination d'assignments. Les

TAB. 1.2. Temps moyen de résolution de l'association de l'algorithme d'enchère et JVC sur le scénario 2

Algorithme	Temps (millisecondes)
JVC	2.003
Algorithme d'enchère	1.787

matrices sont creuses, avec beaucoup d'assignations impossibles et de dimension 25×25 .

1.5. CONCLUSION

La conclusion de ces tests est que l'algorithme d'enchère est conseillé dans un contexte où entre 25 et 200 cibles présentent un comportement favorisant l'élimination des possibilités, donc produisant des matrices creuses. Ceci signifie par exemple que les cibles ne se déplacent pas vers la même direction, qu'elles sont éloignées les unes des autres, qu'elles voyagent à vitesse constante, etc.

Les cas réels peuvent différer quelque peu des scénarios. Les avions de transport ou les bateaux de transport présentent en général des déplacements à trajectoires simples, avec peu de changements de direction et une vitesse constante. Pour ces types de véhicule, il serait préférable d'utiliser l'algorithme d'enchère pour pister de gros groupes. Les avions ou bateaux militaires peuvent présenter un comportement plus chaotique en situation de combat. Dans ce cas, JVC est indiqué. Autrement, en situation de patrouille, l'algorithme d'enchère est conseillé pour les gros groupes.

Il existe des algorithmes de partitionnement¹ qui séparent un gros problème en plusieurs petits problèmes. Il serait intéressant de vérifier si un tel algorithme jumelé à JVC ou à l'algorithme d'enchère améliorerait les performances de l'algorithme d'enchère pour des scénarios de plus de 200 cibles.

Quelques années après l'apparition des algorithmes à deux dimensions, plusieurs se sont intéressés à l'association à plusieurs dimensions dans le but de produire une association plus précise. Le prochain chapitre porte sur ce sujet.

¹En anglais, le terme *Clustering* est utilisé

Chapitre 2

PROBLÈME D'ASSOCIATION À PLUSIEURS DIMENSIONS

2.1. PRÉSENTATION DU PROBLÈME

Dans l'approche standard à deux dimensions (à balayage simple), le processus de fusion se déclenche dès qu'un nouvel ensemble de données (balayage) est reçu. Dans l'approche à balayages multiples, plusieurs balayages sont accumulés pour ensuite décider de l'association sur l'ensemble, augmentant ainsi la quantité d'informations pour réaliser une fusion améliorée. Le principe du pistage à balayages multiples veut que chaque contact de chaque balayage soit associé qu'à une seule piste. La figure 2.1 est un exemple d'association à 4 dimensions où il y a 3 contacts par balayage.

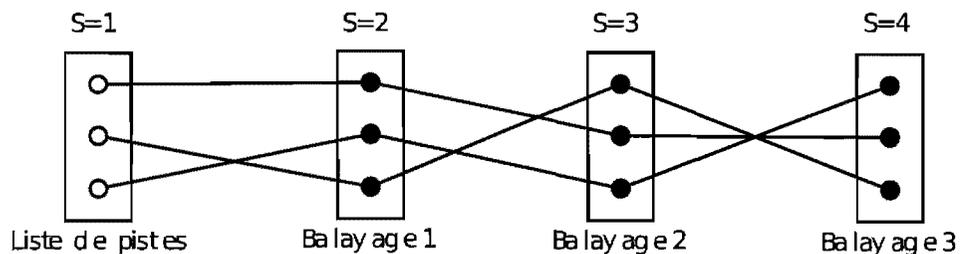


FIG. 2.1. Exemple d'association en $S = 4$ dimensions

2.1.1. Revue littéraire

Singer, Sea, Housewright sont les premiers en 1974 à proposer un algorithme à balayages multiples [20]. Leur méthode est conçue pour pister une seule cible

et ne traite pas les problèmes d'initiation de pistes ou leur suppression. Pour ce faire, ils utilisent une technique qui sera appelée le pistage à hypothèses multiples (MHT¹). À chaque ensemble de mesures reçu, un ensemble d'hypothèses est généré ayant chacune une probabilité *a posteriori* basée sur la règle de Bayes [3]. Plutôt que de choisir tout de suite après le premier balayage la meilleur association, les hypothèses sont propagées dans les temps futures en supposant que les balayages subséquents résoudre les incertitudes. Les hypothèses peuvent se représenter comme un arbre où chaque branche est une hypothèse. La figure 2.2 présente

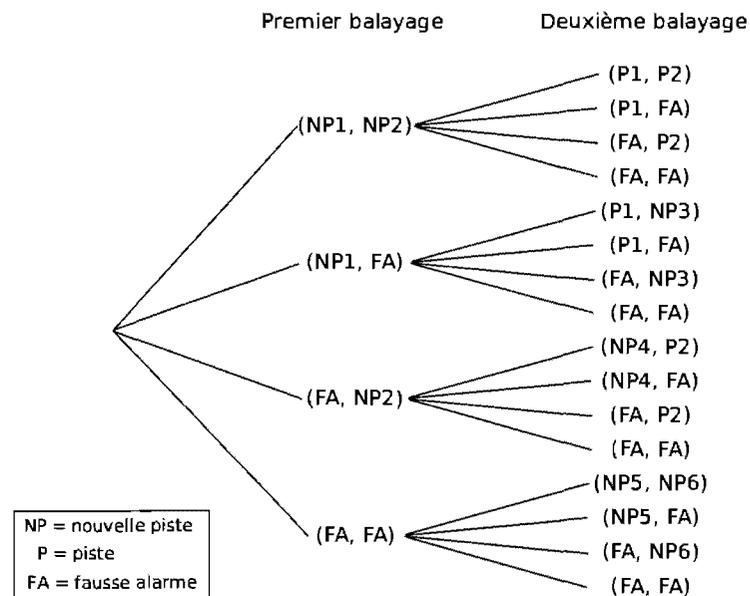


FIG. 2.2. Hypothèses générées par deux contacts après deux balayages

les hypothèses générées par deux contacts après deux balayages. En tout, 16 hypothèses sont générées. En pratique, le nombre d'hypothèses peut être très grand. Pour conserver le nombre d'hypothèses à une quantité raisonnable, les auteurs proposent de combiner les hypothèses qui ont N mesures en commun. Le point intéressant de la recherche de Singer, Sea et Housewright est qu'avec $N = 1$ et $N = 2$, les associations s'approchent de la situation réelle.

Deux ans plus tard, Reid s'inspire des travaux de Singer, Sea et Housewright pour proposer une méthode capable de pister plusieurs cibles [21]. De plus, ses

¹Multiple Hypothesis Tracking

travaux prennent en considération le problème d'initiation de pistes et de suppression. L'article de Reid devient une référence pour les recherches futures sur le MHT.

Les formes théoriques du MHT requièrent un volume de mémoires croissants à mesure que de nouvelles données sont compilées. Plusieurs techniques ont donc été apportées afin de limiter le nombre d'hypothèses pour qu'une version de l'algorithme adaptée aux cas réels soit exécutée. Ces techniques sont composées, entre autres, de partitionnements, d'élagages d'hypothèses et d'unions d'hypothèses.

- (1) Le partitionnement est une solution pour réduire le nombre d'hypothèses qui peuvent être générées. Une partition est un ensemble de pistes qui partage les mêmes observations. Le partage peut être indirect. Par exemple, si la piste 1 partage des observations avec la piste 2 et que la piste 2 partage des observations avec la piste 3, alors les trois sont dans la même partition. Le partitionnement a pour effet de réduire la taille du problème à un ensemble de problèmes plus petits qui sont tous traités indépendamment, ouvrant ainsi la porte à du traitement en parallèle.
- (2) L'élagage se fait généralement de trois façons. Une première consiste à choisir un nombre de balayages N et lorsque ce nombre de balayages est accumulé, la meilleure hypothèse pour le premier balayage est sélectionnée. Une deuxième façon de faire l'élagage consiste à conserver les M hypothèses, M étant prédéterminé, ayant la plus haute probabilité. Une troisième façon de procéder est d'éliminer les hypothèses dont la probabilité descend sous un seuil minimal prédéterminé.

2.1.2. MHT orienté-piste

Une autre approche, le MHT orienté-piste, reconstruit les hypothèses en utilisant les pistes nouvellement mises à jours après chaque balayage (Fig. 2.3). Contrairement au MHT standard, les hypothèses ne sont pas conservées. Les hypothèses sont construites à partir des pistes et des $S - 1$ derniers balayages et seulement les contacts du premier balayage sont fusionnés aux pistes. Deb[22].

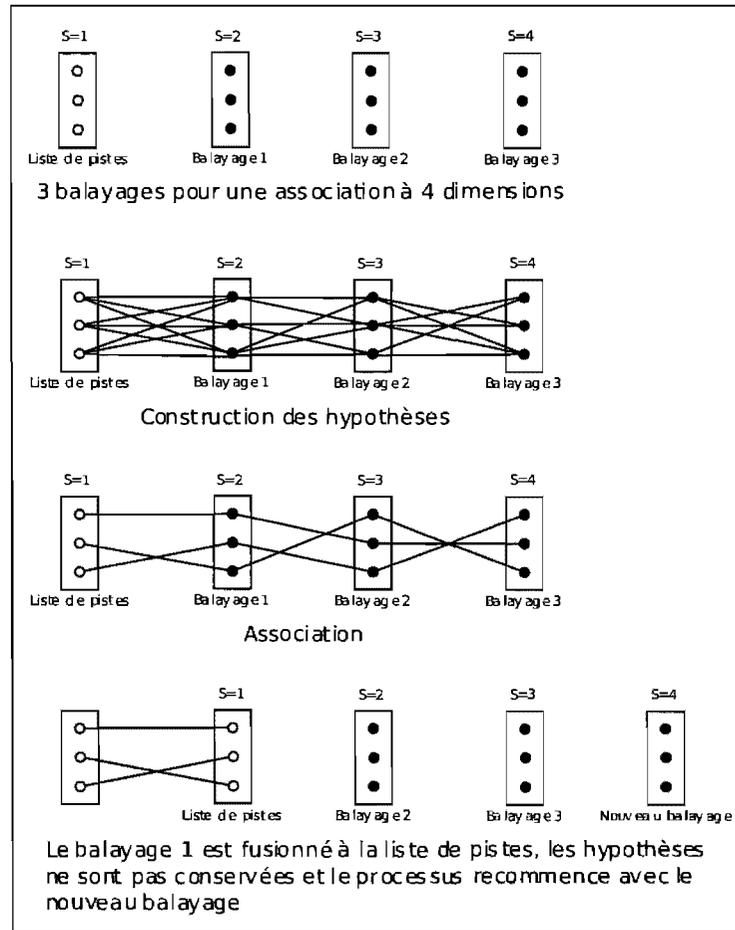


FIG. 2.3. Schéma du MHT orienté-piste. Pour l'exemple, $S = 4$.

et Poore[23] ont développé une méthode orientée-piste qui fait appel au principe des fenêtres glissantes (Fig. 2.4). Le processus reforme les pistes en hypothèses. Contrairement au MHT standard, dans une même hypothèse, les pistes ne partagent pas d'observation. Autrement dit, chaque piste est associée à un seul contact et chaque contact est associé à une seule piste. Le problème se formule comme un problème d'optimisation avec contraintes de dimension S où le but est de choisir l'hypothèse ayant un coût minimal.

Soit n_j , le nombre de contacts lors du $j^{\text{ième}}$ balayage. Le problème se traduit mathématiquement de la façon suivante :

$$\min_{x_{i_1 i_2 \dots i_S}} \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \dots \sum_{i_S=0}^{n_S} C_{i_1 i_2 \dots i_S} x_{i_1 i_2 \dots i_S} \quad (2.1.1)$$

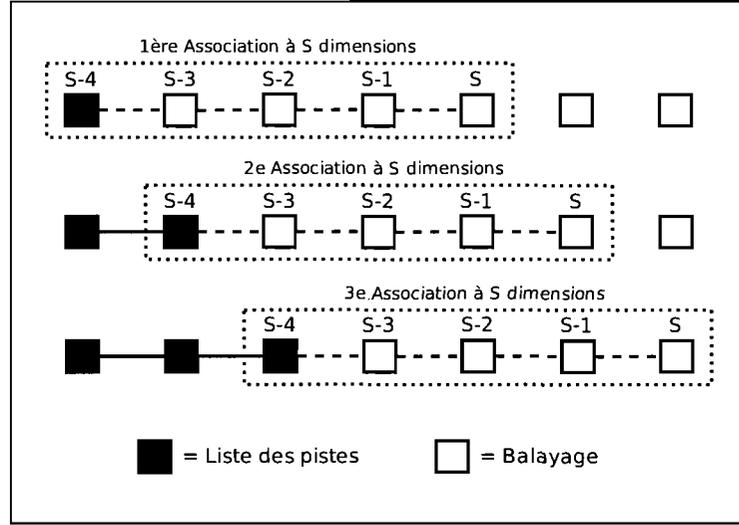


FIG. 2.4. Schéma du principe des fenêtres glissantes. Pour l'exemple, $S = 5$.

Sujet aux contraintes

$$\begin{aligned}
 \sum_{i_2=0}^{n_2} \cdots \sum_{i_S=0}^{n_S} x_{i_1 i_2 \dots i_S} &= 1; & i_1 &= 1, 2, \dots, n_1 \\
 \sum_{i_1=0}^{n_1} \sum_{i_3=0}^{n_3} \cdots \sum_{i_S=0}^{n_S} x_{i_1 i_2 \dots i_S} &= 1; & i_2 &= 1, 2, \dots, n_2 \\
 & \vdots & & \vdots \\
 \sum_{i_1=0}^{n_1} \cdots \sum_{i_{S-1}=0}^{n_{S-1}} x_{i_1 i_2 \dots i_S} &= 1; & i_S &= 1, 2, \dots, n_S
 \end{aligned} \tag{2.1.2}$$

La variable $c_{i_1 i_2 \dots i_S}$ est le coût d'assignation. $x_{i_1 i_2 \dots i_S}$ est une variable binaire qui prend la valeur 1 si $i_1 i_2 \dots i_S$ fait partie de la solution et prend la valeur 0 autrement.

Le problème d'association généralisé à S -dimension est NP-difficile. Pour résoudre ce problème, [22] suggère une méthode qui fait appel à la relaxation lagrangienne pour convertir le problème de dimension S en une série de problèmes de dimension 2, qui eux sont résolubles en temps polynômiaux. En l'occurrence, pour une association $n \times n$, $O(n^3)$ ou moins pour JVC et l'algorithme d'enchère.

Pour les expériences de cette étude, la méthode orientée-piste avec relaxation lagrangienne est utilisée pour résoudre le problème du balayage multiple. Il

convient donc de présenter la relaxation lagrangienne et de décrire son application au problème d'association à plusieurs dimensions pour ensuite proposer un algorithme à balayages multiples.

2.2. RELAXATION LAGRANGIENNE

La relaxation lagrangienne est un outil servant à trouver les bornes maximales d'un problème de maximisation[24].

2.2.1. Théorie

Soit $\Omega \subset \mathbb{R}^m$ et f une fonction linéaire ou quadratique. La relaxation lagrangienne est une technique universelle pour trouver une borne supérieure au problème :

$$\max f(x), \quad x \in \Omega$$

Ω peut s'écrire $\Omega = X \cap \{x : c(x) = 0\}$, où X est un ensemble où il est simple de maximiser f tandis que la contrainte $c = (c_1, \dots, c_m)$ rend le problème difficile à maximiser. La méthode produit un ensemble de solutions optimales pour le problème $\max_{x \in \bar{\Omega}} f(x)$, où $\bar{\Omega}$ est un agrandissement convexe de Ω .

2.2.1.1. Principe de base

Le mécanisme de la relaxation lagrangienne repose sur la relaxation des contraintes qui existe en théorie des graphes. La relaxation des contraintes est une méthode qui fait en sorte qu'un réseau permette plus de solutions. On permet à certaines contraintes d'être violées pour rendre le problème plus facile. Une fonction de coût optionnelle peut être utilisée pour mesurer le niveau de violation des contraintes. Cette technique de relaxation n'a rien à voir avec la méthode de relaxation utilisée pour résoudre une équation aux dérivées partielles elliptiques.

Soit le problème d'optimisation donné plus haut :

$$\max f(x), \quad x \in X, \quad c_j(x) = 0, j = 1, \dots, m \quad (2.2.1)$$

qui sera appelé le problème fondamental. À l'aide du lagrangien L , une fonction de la variable x et de la variable duale $u \in \mathbb{R}^m$, la contrainte $c(x)$ peut être

relaxée.

$$X \times \mathbb{R}^m \ni (x, u) \mapsto L(x, u) := f(x) - \sum_{j=1}^m u_j c_j(x) = f(x) - u^\top c(x). \quad (2.2.2)$$

La fonction duale associée à (2.2.1) et (2.2.2) est la fonction de u définie par :

$$\mathbb{R}^m \ni u \mapsto \theta(u) := \sup_{x \in X} L(x, u). \quad (2.2.3)$$

Le problème dual est donc :

$$\inf_u \theta(u), \quad u \in \mathbb{R}^m \quad (2.2.4)$$

Quel que soit X, f, c dans (2.2.1), la fonction θ est toujours convexe et semi-continue inférieurement[24]. Si $\exists u$ tel que (2.2.3) a une solution optimale en x_u , alors $g_u := -c(x_u)$ est un sous-gradient de θ en u .

2.2.1.2. Contrainte non nulle

Dans le cas du problème (2.1.1) les contraintes ne sont pas nulles. Avec une contrainte non-nulle :

$$\max f(x), \quad x \in X, \quad c_j(x) \leq 0, j = 1, \dots, m \quad (2.2.5)$$

une variable s est utilisée pour transformer (2.2.5) comme suit :

$$\max f(x), \quad x \in X, s \geq 0 \in \mathbb{R}^m, \quad c_j(x) + s = 0, j = 1, \dots, m. \quad (2.2.6)$$

Le lagrangien est :

$$L'(x, s, u) = f(x) - u^\top (c(x) + s) = L(x, u) - u^\top s \quad (2.2.7)$$

La fonction duale devient :

$$\theta'(u) = \begin{cases} +\infty & \text{si } \exists u_j < 0, \\ \theta(u) & \text{sinon} \end{cases} \quad (2.2.8)$$

Le problème dual devient donc :

$$\inf_{u \geq 0} \theta(u). \quad (2.2.9)$$

2.2.2. Application au problème d'association à plusieurs dimensions

Pour le problème (2.1.1), les S contraintes devront être relaxées les unes après les autres. L'algorithme est bâti autour de deux phases principales :

- (1) Relaxation des contraintes
- (2) Application des contraintes et mise à jour des multiplicateurs de Lagrange

2.2.2.1. Relaxation des contraintes

Premièrement, les multiplicateurs de Lagrange sont introduits :

$$\min_{x_{i_1 i_2 \dots i_S}} \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \dots \sum_{i_S=0}^{n_S} (c_{i_1 i_2 \dots i_S} - \lambda_{S i_S}) x_{i_1 i_2 \dots i_S} + \sum_{i_S=0}^{n_S} \lambda_{S i_S} \quad (2.2.10)$$

Les variables $\lambda_{S i_S}$ sont les multiplicateurs de Lagrange de la $S^{\text{ième}}$ contrainte. Ceux-ci vont éventuellement imposer les contraintes relaxées en pénalisant les solutions qui les violent. Deuxièmement, une nouvelle variable binaire est définie.

$$y_{i_1 i_2 \dots i_{S-1}}^{S-1} = \sum_{i_S=0}^{n_S} x_{i_1 i_2 \dots i_S} \quad (2.2.11)$$

Troisièmement, un nouveau coût est défini.

$$d_{i_1 i_2 \dots i_{S-1}}^{S-1} = \min_{i_S} (c_{i_1 i_2 \dots i_S} - \lambda_{S i_S}) \quad (2.2.12)$$

C'est en prenant le coût minimum ici, pour la $S^{\text{ième}}$ dimension qu'il y a risque de violation de la $S^{\text{ième}}$ contrainte. Finalement, le problème relaxé est :

$$\min_{y_{i_1 i_2 \dots i_{S-1}}^{S-1}} \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \dots \sum_{i_{S-1}=0}^{n_{S-1}} d_{i_1 i_2 \dots i_{S-1}}^{S-1} y_{i_1 i_2 \dots i_{S-1}}^{S-1} + \sum_{i_S=0}^{n_S} \lambda_{S i_S} \quad (2.2.13)$$

Sujet aux contraintes

$$\begin{aligned} \sum_{i_2=0}^{n_2} \dots \sum_{i_{S-1}=0}^{n_{S-1}} y_{i_1 i_2 \dots i_{S-1}}^{S-1} &= 1; & i_1 &= 1, 2, \dots, n_1 \\ \sum_{i_1=0}^{n_1} \sum_{i_3=0}^{n_3} \dots \sum_{i_{S-1}=0}^{n_{S-1}} y_{i_1 i_2 \dots i_{S-1}}^{S-1} &= 1; & i_2 &= 1, 2, \dots, n_2 \\ & \vdots & & \\ \sum_{i_1=0}^{n_1} \dots \sum_{i_{S-2}=0}^{n_{S-2}} y_{i_1 i_2 \dots i_{S-1}}^{S-1} &= 1; & i_{S-1} &= 1, 2, \dots, n_{S-1} \end{aligned} \quad (2.2.14)$$

De façon similaire les $(S - 1)$ ième, $(S - 2)$ ième, \dots , 3ième contraintes sont relaxées.

Une fois la 3ième contrainte relaxée, le problème est à 2 dimensions :

$$\min_{y_{i_1 i_2}^2} \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} d_{i_1 i_2}^2 y_{i_1 i_2}^2 + \sum_{i_3=0}^{n_3} \lambda_{3i_3} + \dots + \sum_{i_S=0}^{n_S} \lambda_{Si_S} \quad (2.2.15)$$

soumis aux contraintes :

$$\begin{aligned} \sum_{i_2=0}^{n_2} y_{i_1 i_2}^2 &= 1; & i_1 &= 1, 2, \dots, n_1 \\ \sum_{i_1=0}^{n_1} y_{i_1 i_2}^2 &= 1; & i_2 &= 1, 2, \dots, n_2 \end{aligned} \quad (2.2.16)$$

2.2.2.2. Application des contraintes, mise à jour des multiplicateurs de Lagrange

Une fois la relaxation effectuée, le problème à deux dimensions qui en résulte peut être résolu. La solution de ce problème est la solution duale J_2 . Soit $\{(j, k_j) : j = 0, \dots, m\}$, $m = \max\{n_1, n_2\}$ la solution du problème ;

$$\min_{y_{i_1 i_2}^2} \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} d_{i_1 i_2}^2 y_{i_1 i_2}^2 = \sum_{j=0}^m d_{j k_j}^2$$

Les contraintes sur $y_{i_1 i_2}^2$ ont été utilisées pour obtenir la solution. Cette solution s'exprime également ;

$$\sum_{j=0}^m d_{j k_j}^2 = \sum_{j=0}^m \min_{i_3} \left(d_{j k_j i_3}^3 - \lambda_{3i_3} \right)$$

La troisième contrainte est appliquée, le problème est donc :

$$\min_{y_{j k_j i_3}^3} \sum_{j=0}^m \sum_{i_3=0}^{n_3} \left(d_{j k_j i_3}^3 - \lambda_{3i_3} \right) y_{j k_j i_3}^3 \quad (2.2.17)$$

avec les contraintes :

$$\begin{aligned} \sum_{i_3=0}^{n_3} y_{j k_j i_3}^3 &= 1; & j k_j &= 1, 2, \dots, m \\ \sum_{j=0}^m y_{j k_j i_3}^3 &= 1; & i_3 &= 1, 2, \dots, n_3 \end{aligned} \quad (2.2.18)$$

Le but est maintenant de déterminer la meilleure association entre les couples (j, k_j) et les contacts i_3 . C'est un problème à deux dimensions. Une fois la solution

J_3 de ce problème calculée, les multiplicateurs de Lagrange sont mis à jours à l'aide de la méthode heuristique donnée par Deb et al.[22]. L'incrémentation se fait comme suit :

$$\begin{aligned}\lambda_{ri_r} &= \lambda_{ri_r} + \left[\frac{J_r^a - f_{\text{duale}}}{|g|_2^2} \right] \left[\frac{n_r \mu_{ri_r}}{\sum_{k=1}^{n_r} \mu_{rk}} \right] g_{ri_r} \\ i_r &= 1, 2, \dots, n_r; \quad r = 3, 4, \dots, S.\end{aligned}$$

où J_r^a est la meilleure solution pour le $r^{\text{ième}}$ sous-problème trouvée jusqu'à maintenant et f_{duale} est la meilleure solution duale. μ_{ri_r} est l'ensemble des prix générés par l'algorithme à deux dimensions pour obtenir la solution du $r^{\text{ième}}$ sous-problème J_r , g_{ri_r} est le sous gradient. Le processus est répété pour $r = 4, 5, \dots, S$, leur solution est trouvée à partir d'un problème à deux dimensions en appliquant le $r^{\text{ième}}$ ensemble de contraintes et en mettant à jours les multiplicateurs de Lagrange λ_{ri_r} .

Ces deux phases sont répétées jusqu'à ce que l'écart entre la solution duale J_2 et la solution fondamentale J_S n'est pas nulle ou n'est pas sous un seuil d'erreur maximale tolérable. Si J_2 et J_S sont égales, la solution est optimale.

2.3. ALGORITHME À BALAYAGES MULTIPLES

Dans cette section, les étapes principales de l'algorithme à balayages multiples sont décrites.

(1) *Étape 1 : Initialisation*

Lors de cette étape, les principales variables sont initialisées. Les solutions fondamentale et duale sont initialisées comme suit :

$$\begin{aligned}f_{\text{fondamentale}} &= \infty \\ f_{\text{duale}} &= -\infty\end{aligned}$$

Les variables d'arrêts de l'algorithme sont initialisées

- Seuil d'erreur maximale tolérable : ϵ_{max}
- Nombre d'itérations maximales : ι_{max}

(2) *Étape 2 : Calcul des coûts réduits pour $r = S - 1, S - 2, \dots, 2$.*

$$d_{i_1 i_2 \dots i_r}^r = \min_{i_{r+1}} \dots \min_{i_S} c_{i_1 \dots i_{r+1} \dots i_S} - \lambda_{r+1 i_{r+1}} \dots - \lambda_{S i_S}$$

(3) *Étape 3 : Application de l'ensemble de contraintes R .*

C'est un problème à deux dimension, soluble dans un temps d'ordre polynômial.

(4) *Étape 4 : Mises à jour des multiplicateurs de Lagrange.*

Si $R = S$ passer à l'étape 5. Sinon, les multiplicateurs de Lagrange sont mis à jours à l'aide de la méthode heuristique [22].

(5) *Étape 5 : Récursion*

Si la contrainte $S - 1$ n'a pas été appliquée, retourner à l'étape 3 et appliquer la contrainte suivante.

(6) *Étape 6 : Itération : amélioration de la solution.*

$$\begin{aligned} f_{\text{fondamentale}} &= \min (f_{\text{fondamentale}}, J_S) \\ f_{\text{duale}} &= \max (f_{\text{duale}}, J_2) \\ \epsilon &= (f_{\text{fondamentale}} - f_{\text{duale}}) / |f_{\text{fondamentale}}| \\ \text{iter} &= \text{iter} + 1 \end{aligned}$$

Si $\epsilon < \epsilon_{\max}$ ou $\text{iter} > \iota_{\max}$, passer à l'étape 7. Sinon retourner à l'étape 2.

(7) *Étape 7 : Résultat Final.*

Un exemple du fonctionnement de cet algorithme est donné dans l'article de Deb et al.[22].

2.4. EXPÉRIMENTATIONS

Les tests sont réalisés pour vérifier lequel entre l'algorithme d'enchère et JVC est le plus rapide pour calculer les solutions à tous les problèmes à deux dimensions que l'algorithme à balayages multiples génère. La qualité de l'association est également vérifié en deux, trois et quatre dimensions entre autres sur un scénario où le radar a une probabilité de détection plus faible. Les tests consistent

TAB. 2.1. Temps moyen en millisecondes de résolution de l'association en 3 et 4 dimensions pour le scénario 1

Associateur 2-D	3-D	4-D
JVC	9.763	18.038
Enchère	27.234	633.436

TAB. 2.2. Comparaison du temps moyen de résolution de l'association entre JVC et l'algorithme d'enchère en 2-D et 3-D pour le scénario 2

Associateur 2-D	2-D	3-D
JVC	2.003	11.166
Enchère	1.787	13.338

à appliquer l'algorithme à balayages multiples sur les scénarios 1, 2 et 3 (section 1.4.2).

2.4.1. Temps d'association

Théoriquement, l'algorithme à balayages multiples demande plus de calculs que l'algorithme à balayages simples pour trouver l'association optimale. Ceci toutefois au profit de la qualité de l'association. Dans cette section, le but de l'expérimentation est de comparer la rapidité de l'algorithme d'enchère à JVC lorsqu'ils sont appliqués à l'algorithme à balayages multiples.

Le tableau 2.1 présente les résultats des expérimentations sur le scénario 1 en utilisant le balayage multiple en 3-D et en 4-D. Comme dans le cas du balayage simple, l'algorithme à balayages multiples est plus rapide lorsque l'associateur à deux dimensions est JVC pour le scénario 1.

En balayage simple, l'algorithme d'enchère résout plus rapidement l'association pour le scénario 2. Les tests en balayages multiples à trois dimensions montrent que pour ce même scénario, JVC exécute les calculs plus rapidement en plusieurs dimensions (tableau 2.2). Cette perte d'efficacité de l'algorithme d'enchère s'explique par le fait que la création de pistes et les cibles non détectées doivent être considérer. La phase de corrélation n'élimine pas ces assignations,

qui sont de la forme $(0, 0, 0, 2, 6)$ ou $(1, 4, 0, 0, 3)$ par exemple, et elles peuvent être nombreuses en balayages multiples. Les matrices d'associations ne sont donc jamais très creuses.

Les tableaux 2.1 et 2.2 montrent que le scénario a une influence sur la vitesse de l'algorithme d'association à plusieurs dimensions. L'association en 3 dimensions pour le scénario 1 prend trois fois plus de temps avec l'algorithme d'enchère qu'avec JVC tandis que pour le scénario 2 l'association en 3 dimension est du même ordre de grandeur pour les deux méthodes à 2 dimensions.

2.4.2. Qualité d'association

Pour calculer la qualité d'association, deux métriques sont utilisées :

- (1) la Pureté de la Piste (PP) et
- (2) le Ratio d'associations correctes (RAC)

Ces métriques mesurent à quel point une piste dans un scénario est associée à une cible identifiée par son identité réelle, un nombre unique donné par le simulateur utilisé pour évaluer les performances d'un algorithme d'association. Soient les pistes t_1, \dots, t_b et un ensemble de cibles g_1, \dots, g_a ayant une identité réelle de $1, \dots, a$. La matrice suivante, qui est appelée matrice de confusion, est construite :

	g_1	g_2	g_3	\dots	g_a
t_1	C_{11}	C_{12}	C_{13}	\dots	C_{1a}
t_2	C_{21}	C_{22}	C_{23}	\dots	C_{2a}
t_3	C_{31}	C_{32}	C_{33}	\dots	C_{3a}
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
t_b	C_{b1}	C_{b2}	C_{b3}	\dots	C_{ba}

C_{ji} est le nombre de fois qu'une cible g_i ayant comme identité réelle i est associé à la piste t_j .

PP est définie comme la proportion d'associations correctes pour une piste donnée. La pureté d'une piste t_j est définie comme la valeur normalisée du plus grand $C_{ji}, i = 1, \dots, a$, i.e. la proportion d'association à une cible à laquelle une

TAB. 2.3. Qualité de l'association en 2-D, 3-D et 4-D pour le scénario 1

Métriques	2-D	3-D	4-D
MPPP(%)	87.5	88.3	94.2
MPRAC(%)	87.5	88.7	93.9

piste a été associé le plus souvent. Mathématiquement :

$$PP[t_j] = \frac{\max_i C_{ji}}{\sum_{i=1}^a C_{ji}}$$

Cette mesure est calculée pour chaque piste individuellement, mais avec un grand nombre de pistes, il est plus intéressant de calculer une statistique. Pour les tests de cette section, la statistique utilisée est la moyenne pondérée des PP (MPPP) prises sur toutes les pistes. La définition qui résulte de la MPPP est la suivante :

$$MPPP = \frac{\sum_{j=1}^b \max_i C_{ji}}{\sum_{j=1}^b \sum_{i=1}^a C_{ji}}$$

La métrique RAC mesure la qualité d'association pour une cible plutôt que pour une piste. Le RAC de la cible g_i ayant comme identité réelle i est défini comme la valeur normalisée du plus grand $C_{ji}, j = 1, \dots, b$, i.e. la proportion d'association à une piste à laquelle une cible a été associé le plus souvent. Mathématiquement :

$$RAC[g_i] = \frac{\max_j C_{ji}}{\sum_{j=1}^b C_{ji}}$$

Comme pour PP, une moyenne pondérée pour RAC peut être calculée :

$$MPRAC = \frac{\sum_{i=1}^a \max_j C_{ji}}{\sum_{i=1}^a \sum_{j=1}^b C_{ji}}$$

La qualité d'association du balayage multiple a été étudiée sur le scénario 1 et le scénario 3. Pour le scénario 1, l'association s'améliore lorsque la dimension augmente (tableau 2.3).

Il est important de souligner le fait qu'en dimension 2, MPPP et MPRAC sont égaux. En pratique, c'est une possibilité plutôt rare, car elle nécessite quelques conditions particulières.

- (1) La première condition est que la création accidentelle de pistes doit être évitée. La création accidentelle de pistes survient lorsque pour un balayage, un contact ne peut pas être associé à aucune piste existante, donc en crée une nouvelle. Pour un exemple où il y a 10 balayages, une matrice de confusion pour un scénario où il y a une création accidentelle de pistes pourrait avoir cette forme :

	g_1	g_2	g_3	total
t_1	8	1	1	10
t_2	2	6	2	10
t_3	0	2	7	9
t_4	0	1	0	1
total	10	10	10	30

La création accidentelle de pistes correspond à la cible g_2 qui crée la piste t_4 . Dans cet exemple, MPPP = 73.3% et MPRAC = 70%.

- (2) une deuxième condition est que deux cibles ne doivent pas produire un seul et même contact. Ceci peut se produire par exemple lorsque deux cibles sont très proches l'une de l'autre. Une matrice de confusion représentant cette situation pourrait avoir cette forme :

	g_1	g_2	g_3	total
t_1	8	1	1	10
t_2	2	6	6	14
t_3	0	3	3	6
total	10	10	10	30

Les cibles g_2 et g_3 ont produit 6 fois le même contact détecté par la piste t_2 . Dans cet exemple, MPPP = 56.7% et MPRAC = 66.7%.

- (3) Une troisième condition est que deux pistes ne doivent pas être associées à une même cible. Autrement dit, une cible ne doit pas produire deux

TAB. 2.4. Qualité de l'association en 2-D, 3-D et 4-D pour le scénario 3

Métriques	2-D	3-D	4-D
MPPP(%)	39.8	31.5	36.9
MPRAC(%)	42.9	32.0	31.1

contacts. Dans tous les scénarios de cette recherche, ce cas est évité. Dans la pratique, une matrice de confusion représentant cette situation pourrait ressembler à :

	g_1	g_2	g_3	total
t_1	3	6	1	10
t_2	2	6	2	10
t_3	0	3	7	10
total	5	15	10	30

La cible g_2 a produit deux contacts à six reprises qui sont détectés par les pistes t_1 et t_2 . Dans cet exemple, MPPP = 63.3% et MPRAC = 53.3%.

Lorsque ces trois conditions sont respectées il est possible que MPPP et MPRAC soient égaux. En pratique, que ces trois conditions s'observent simultanément n'est pas fréquent, mais dans le scénario 1 c'est le cas.

Le scénario 3 est composé de 9 cibles se déplaçant dans un même groupe dans une même direction. La difficulté dans ce scénario est que la probabilité de détection du radar est de 75%. Les résultats de la simulation en 2, 3 et 4 dimensions sont donnés dans le tableau 2.4. L'association ne s'améliore pas lorsque la dimension augmente, parce que la corrélation n'élimine pas assez de possibilités. Avec une distance statistique maximale plus courte l'association aurait probablement été meilleure.

2.5. CONCLUSION

Les tests pour le temps d'association montrent que JVC est plus rapide que l'algorithme d'enchère en balayages multiples pour les scénarios 1 et 2. Au premier chapitre, il est montré qu'en balayages simples, l'algorithme d'enchère est

plus rapide que JVC sur les scénarios avec beaucoup de cibles, où les matrices d'association sont creuses. Les tests sur le scénario 2 en balayages multiples montre que les avantages de l'algorithme d'enchère sur JVC, avec le scénario 2, sont perdus en balayages multiples. La rapidité de JVC comparativement à l'algorithme d'enchère s'explique par le fait que les matrices d'association ne sont pas aussi creuses en balayages multiples qu'en balayages simples, parce que les nombreuses possibilités de création de pistes ne sont pas rejetées. De plus, comme l'algorithme d'association à 2 dimensions est utilisé plusieurs fois pour résoudre un problème de dimension S , la différence de temps de calcul de l'association, lorsque l'algorithme d'association à 2 dimensions est JVC ou l'algorithme d'enchère, peut être très grande.

La qualité d'association s'améliore pour le scénario 1 lorsque le nombre de balayages est augmenté, ce qui représente les scénarios où quelques objets se déplacent en petits groupes. Cependant, la qualité d'association ne s'améliore pas pour le scénario 3, où le radar à une probabilité de détection de 75%, lorsque le nombre de balayages est augmenté. Une faible probabilité de détection peut être due à de mauvaises conditions météo, à des dommages aux systèmes électroniques de la plateforme ou à tout autre élément pouvant causer des interférences. Dans ces cas, il serait préférable d'utiliser une méthode à balayages simples pour effectuer le pistage.

Le prochain chapitre traite du balayage multiple dans le contexte où il y a plusieurs senseurs.

Chapitre 3

ALGORITHME À BALAYAGES MULTIPLES ET MULTIPLE SENSEURS

Le but de la fusion de données est de donner une interprétation de la réalité selon les informations recueillies. Si ces informations proviennent de plusieurs sources fiables, la fusion de données peut produire une image qui se rapproche beaucoup plus de la réalité. Dans ce chapitre, il est question de l'association à balayages multiples lorsque plusieurs plusieurs senseurs collectent l'information. Il est expliqué qu'en balayage multiple, lorsque la meilleure association est déterminée, les pistes doivent être estimées provisoirement. Deux méthodes d'estimation des pistes sont comparées : la méthode de profondeur temporelle maximale et la méthode de Opitz.

3.1. PROFONDEUR TEMPORELLE MAXIMALE

Dans le but d'éviter une explosion exponentielle des hypothèses, le MHT s'opère sur un nombre limité de balayages. La différence de temps qui sépare un balayage de son suivant détermine la profondeur de temps sur lequel les hypothèses sont évaluées. Dans l'approche standard à balayages multiples et multiple senseurs, à S -dimensions, l'association se fait sur les $S - 1$ derniers balayages sans égard à leur source. Ces $S - 1$ balayages peuvent provenir de plusieurs senseurs différents. Ceci peut entraîner une baisse significative de la profondeur de temps de l'association jusqu'à la réduire à zéro si il y a S senseurs ou plus pour rapporter l'information. Par exemple, supposons que l'association se fait en 3 dimensions

et qu'il y a trois radars. Si les trois radars envoient les informations de leurs balayages en même temps les cibles ne se seront pas déplacées. Les bénéfices de l'approche à balayages multiples seraient perdus, car son utilité se fonde sur la possibilité de recueillir plus d'informations provenant de temps différents.

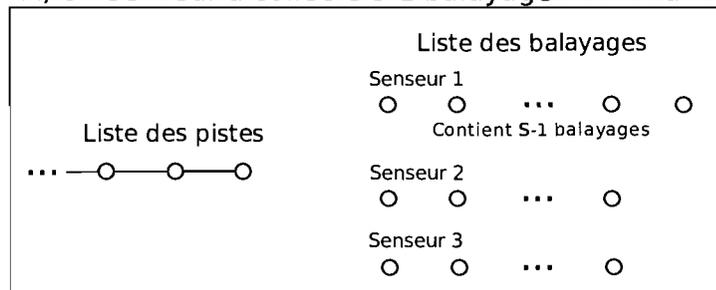
Dans la méthode de Profondeur temporelle maximale de Kirubarajan et al. [25], l'algorithme d'association alterne d'un senseur à l'autre mettant à jour les pistes utilisant une technique de fenêtres glissantes. Spécifiquement, pour chaque senseur, une liste des balayages qu'il a effectués est préservée. La dimension S étant fixée, chaque senseur accumule dans sa liste les balayages qu'il a effectués jusqu'à ce qu'elle en contienne $S - 1$ (Figure 3.1. A). À ce moment, une association de dimensions S est effectuée sur les balayages de la liste (Figure 3.1. B). Les pistes sont mises à jour avec le balayage $S - (S - 2)$, le plus vieux balayage de la liste (Figure 3.1. C). Le balayage $S - (S - 1)$ correspond à la liste des pistes. Ce balayage est, par la suite, éliminé de la liste du senseur.

La méthode de profondeur temporelle maximale a l'avantage d'augmenter l'intervalle de temps entre chaque balayage, laissant les cibles se déplacer. La méthode permet aussi de travailler avec de l'information provenant de temps plus anciens, la vraie raison d'être du balayage multiple. Avec la méthode à balayages multiples et senseurs multiples standard, pour travailler avec des balayages aussi anciens, la dimension doit être augmentée, réduisant considérablement les performances de l'algorithme à cause de sa nature très lourde en calculs.

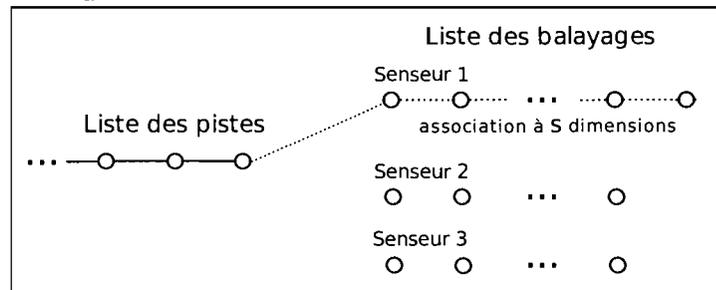
Une fois la solution calculée, la fusion est pratiquée entre les pistes et les contacts du plus vieux balayage de la liste. Cette décision est définitive et n'est pas sujette à changements pour le reste du scénario.

Toutes les méthodes à multiple senseurs ont une faiblesse en commun au niveau de la mise à jour des pistes. Les senseurs peuvent être éloignés les uns des autres et, dans ce cas, avoir une perspective différente de la position des cibles. Deux radars qui terminent un balayage en même temps n'ont pas nécessairement détecté chaque cible en même temps. Ils peuvent donc avoir détecté les mêmes cibles, mais à des positions différentes. Dans certain cas, si deux radars terminent

A) Un Senseur a collecté S-1 balayages dans sa liste



B) Une association à S dimensions est effectuée sur cette liste



C) Mise à jour des pistes

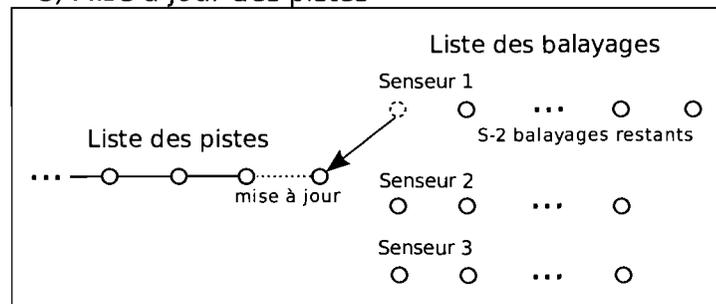


FIG. 3.1. Association à balayages multiples et à senseurs multiples utilisant la profondeur temporelle maximale

leur balayage presque en même temps, il se peut que le radar qui complète son balayage en deuxième ait détecté une cible en premier. Dans l'ordre des balayages, la cible va donc reculer. Pour ce problème, Wang, Kirubarajan et Bar-Shalom proposent une méthode de mise à jour à temps négatif dans leur article [26]. La méthode de profondeur temporelle maximale, quant à elle, présente un désavantage particulier qui vient du fait que la vitesse de rotation diffère selon le radar, donc tous les radars ne remplissent pas leur liste de balayages à la même vitesse. Par conséquent, le dernier balayage de la liste d'un radar peut provenir d'un temps plus ancien que la piste qui est mise à jour.

3.2. ESTIMATION PROVISOIRE

Comme seulement le dernier balayage est fusionné, une estimation de l'association doit être effectuée pour les, au plus, $S - 2$ balayages restants de chaque senseur. Autrement, l'information de la piste sera en retard sur la situation réelle. Contrairement à l'association à S dimensions décrite avant, cette estimation est sujette à des changements dans le futur.

3.2.1. Estimation par profondeur temporelle maximale

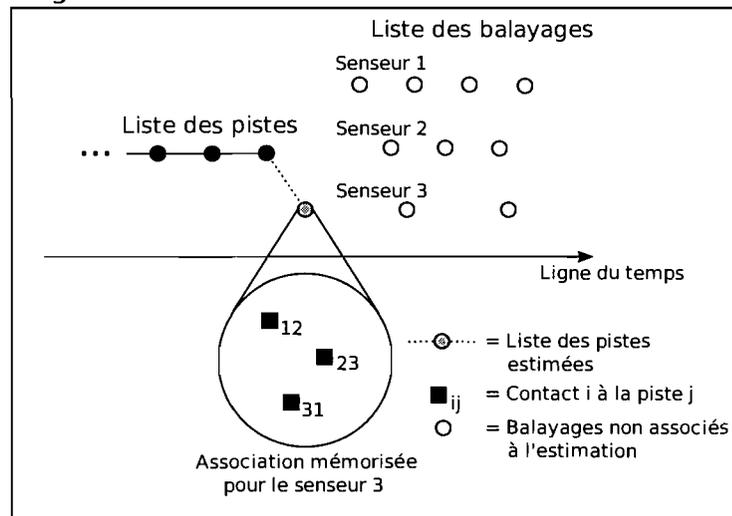
Pour conserver l'avantage donné par la méthode de profondeur temporelle maximale, l'estimation provisoire peut être calculée de la même façon [25]. Une liste contenant les pistes estimées est construite à partir des balayages restants de chaque senseur. La méthode commence avec les senseurs qui ont $S - 2$ balayages complets d'accumulés. En procédant par ordre chronologique, la solution du problème à $S - 1$ dimensions est calculée pour chacun des senseurs. La liste des pistes estimées est alors mise à jour de la même façon qu'avec le problème d'association de dimension S . Quand il ne reste plus de senseur ayant $S - 2$ balayages complets, le processus est répété pour les senseurs ayant $S - 3$ balayages complets. Ce processus continue jusqu'à ce qu'il n'y ait que des senseurs ayant un seul balayage complet. À ce moment, la liste des pistes estimées est mise à jour avec les balayages restants en utilisant une méthode à 2 dimensions. Cette technique demande beaucoup d'opération, donc peut ne pas être adaptée à des situations en temps réel où le nombre de cibles, le nombre de senseurs ou la dimension choisie pourraient augmenter le temps de calcul.

3.2.2. Estimation par l'association de dimension S

Une alternative proposée par Opitz[27] est d'utiliser les associations calculées précédemment. L'idée est de conserver en mémoire le résultat de la dernière association à S dimensions de chaque senseur et de se baser sur ces résultats pour calculer l'estimation. Pour cette expérience, l'algorithme d'estimation a été construit pour éviter le plus possible que les estimations soient hors séquence. Quand vient le temps de calculer l'estimation, les pistes estimées sont mises à

jour avec le plus vieux balayage de tous les senseurs. La liste des pistes estimées est mise à jour avec le balayage selon l'association S-D gardée en mémoire pour le senseur qui l'a générée (Figure 3.2 A). Après la mise à jour, le balayage est éliminé et le processus est repris avec le prochain balayage qui correspond au plus vieux (Figure 3.2 B).

A) Mise à jour selon l'association à S dimensions gardée en mémoire



B) mise à jour avec les plus vieux balayages restants

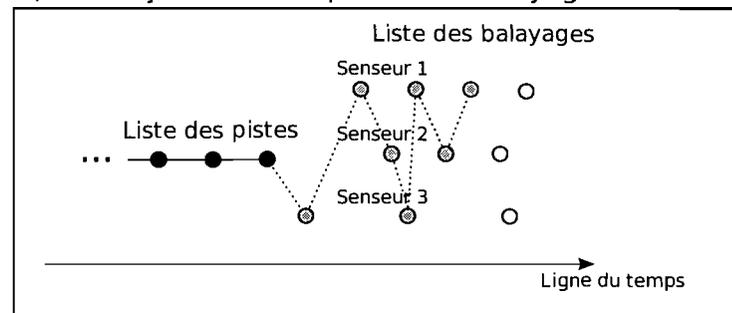


FIG. 3.2. Estimation provisoire utilisant la dernière association à S dimensions calculée

Exemple 3.2.1. *Supposons que l'association se fait en 5 dimensions avec 2 senseurs. L'association à S dimensions vient d'être trouvée, l'estimation doit à présent être calculée. Le plus vieux balayage appartient au senseur 1. La dernière*

association à S dimensions gardée en mémoire est :

(1, 3, 1, 4, 1)

(2, 1, 3, 3, 3)

(3, 2, 2, 1, 2)

(4, 4, 4, 2, 4)

La première entrée de chaque vecteur est le numéro de piste, les autres sont, dans l'ordre, un contact des balayages $S - 4, S - 3, S - 2$ et $S - 1$. Le balayage $S - 4$ a déjà participé à l'association $S-D$, donc a été retiré de ces vecteurs. En réalité, l'association en mémoire est :

(1, 1, 4, 1)

(2, 3, 3, 3)

(3, 2, 1, 2)

(4, 4, 2, 4)

Les pistes sont mises à jour avec le plus vieux balayage de cette association. Donc l'association contacts-pistes est :

(1, 1), (2, 3), (3, 2), (4, 4)

Le plus vieux balayage est ensuite éliminé. L'association en mémoire est :

(1, 4, 1)

(2, 3, 3)

(3, 1, 2)

(4, 2, 4)

Le plus vieux balayage maintenant appartient au senseur 2. L'association qu'il contient en mémoire est :

(1, 3, 2, 1)

(2, 2, 4, 2)

(3, 4, 1, 4)

(4, 1, 3, 3)

L'association piste-contact est :

(1, 3), (2, 2), (3, 4), (4, 1)

Le dernier balayage est ensuite éliminé. L'association en mémoire est :

(1, 2, 1)

(2, 4, 2)

(3, 1, 4)

(4, 3, 3)

Le plus vieux balayage appartient maintenant au senseur 1. Et le processus continue jusqu'à ce qu'il n'y ait plus de balayage dans les listes.

La méthode de Opitz a l'avantage sur la méthode de profondeur temporelle maximale qu'elle réduit considérablement le nombre d'opérations nécessaires à l'estimation. La précision des calculs n'est cependant pas assurée. Un autre avantage est que la chronologie des balayages est conservée.

3.3. RÉSULTATS

3.3.1. Méthodologie

Les tests de cette section ont pour but de vérifier l'efficacité de la méthode de Opitz pour l'estimation provisoire en comparaison à la méthode de profondeur temporelle maximale sur des scénarios où les données sont collectées par un radar ou deux radars. Conséquemment aux résultats du chapitre précédent, JVC sera l'associateur à deux dimensions pour les calculs d'association à balayages multiples.

Les scénarios utilisés sont le scénario 1 et un nouveau, le scénario 4 (Figure 3.3). Le scénario 4 est composé de 7 cibles se déplaçant de façon à créer des conflits à certaines occasions. Le scénario contient 5 plateformes, mais seulement une plateforme fixe ayant deux radars rotatifs est utilisée pour les tests de cette section.

3.3.2. Analyse

Les tests avec un radar montrent que pour les scénarios 1 et 4 le temps de calcul de l'estimation est significativement moins élevé lorsqu'elle est effectuée avec la méthode de Opitz (Tableau 3.1). La qualité de l'association, quant à elle, ne semble pas être affectée par la méthode d'estimation lorsqu'il n'y a qu'un

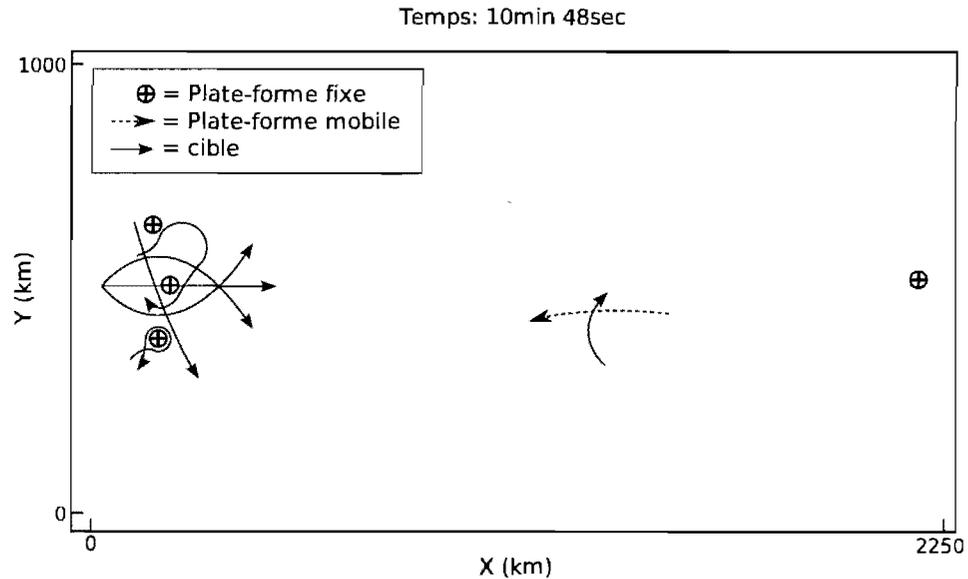


FIG. 3.3. Trajectoires décrites par les cibles et les plateformes dans le scénario 4

senneur (Tableaux 3.2, 3.3). Les résultats sont semblables pour les deux méthodes dans chaque scénario.

Comme pour les simulations avec un radar, celles avec deux radars montrent que la vitesse à laquelle est calculée l'estimation est beaucoup plus rapide avec la méthode de Opitz qu'avec celle de profondeur temporelle maximale (Tableau 3.4).

Dans certaines cases des tableaux de cette section, MPPP et MPRAC sont égaux. Les raisons qui expliquent ce phénomène sont données à la section 2.4.2.

3.4. CONCLUSION

Pour le balayage multiple avec un senseur la qualité d'association de l'estimation reste la même pour les deux méthodes mais avec la méthode de Opitz, le calcul de l'association est plus rapide. Donc, dans un contexte de surveillance où il n'y a qu'un senseur pour produire les balayages, il est préférable d'utiliser la méthode de Opitz pour calculer l'estimation provisoire.

Avec deux senseurs, la méthode de Opitz présente une qualité d'estimation inférieure à la méthode de profondeur temporelle maximale en dimension trois. La méthode de Opitz demeure cependant plus rapide. Le choix de la méthode

TAB. 3.1. Temps du calcul de l'estimation en 3-D et 4-D pour le scénario 1 et le scénario 4

Méthode	Temps Scénario 1 (millisecondes)		Temps Scénario 4 (millisecondes)	
	3-D	4-D	3-D	4-D
Profondeur temporelle maximale	10.635	84.944	21.270	138.560
Méthode de Opitz	5.265	11.857	10.665	17.217

TAB. 3.2. Qualité de l'estimation, 3-D et 4-D pour le scénario 1

Méthode	Métriques	3-D	4-D
Profondeur temporelle maximale	MPPP(%)	87.9	93.3
	MPRAC(%)	86.6	93.1
Méthode de Opitz	MPPP(%)	88.3	94.2
	MPRAC(%)	86.7	93.9

TAB. 3.3. Qualité de l'estimation, 3-D et 4-D pour le scénario 4

Méthode	Métriques	3-D	4-D
Profondeur temporelle maximale	MPPP(%)	88.8	89.0
	MPRAC(%)	83.4	83.1
Méthode de Opitz	MPPP(%)	88.8	88.9
	MPRAC(%)	83.4	83.2

TAB. 3.4. Temps du calcul de l'estimation en senseurs multiples, en 3-D et 4-D pour le scénario 1 et le scénario 4

Méthode	Temps Scénario 1 (millisecondes)		Temps Scénario 4 (millisecondes)	
	3-D	4-D	3-D	4-D
Profondeur temporelle maximale	14.232	153.302	28.979	338.797
Méthode de Opitz	8.206	17.582	16.236	49.784

TAB. 3.5. Qualité de l'estimation en senseurs multiples, en 3-D et 4-D pour le scénario 1

Méthode	Métriques	3-D	4-D
Profondeur temporelle maximale	MPPP(%)	87.1	90.4
	MPRAC(%)	87.1	89.3
Méthode de Opitz	MPPP(%)	76.3	91.1
	MPRAC(%)	76.3	89.4

TAB. 3.6. Qualité de l'estimation en senseurs multiples, en 3-D et 4-D pour le scénario 4

Méthode	Métriques	3-D	4-D
Profondeur temporelle maximale	MPPP(%)	84.7	85.0
	MPRAC(%)	80.4	79.9
Méthode de Opitz	MPPP(%)	82.0	89.4
	MPRAC(%)	79.4	89.4

en dimension trois dépend donc du compromis que l'utilisateur est prêt à faire et les limites du processeur de l'ordinateur qui exécute les calculs. Une grande quantité de contacts peut pousser le processeur à ses limites et forcer l'utilisateur à se servir d'une méthode plus rapide mais moins précise. En dimension quatre, la méthode de Opitz surclasse la méthode de profondeur temporelle maximale autant pour la qualité de l'estimation que pour la vitesse de calcul. Il est donc conseillé d'utiliser la méthode de Opitz pour le cas d'une estimation provisoire en dimension quatre.

CONCLUSION

Le but de cette recherche était d'optimiser un algorithme à balayages multiples et à senseurs multiples en choisissant le meilleur algorithme d'association linéaire à deux dimensions et en utilisant une méthode d'estimation sans calculs pour produire une estimation provisoire des pistes.

Le premier chapitre traitait des algorithmes d'association linéaire à deux dimensions appliqués au balayage simple. L'algorithme d'enchère et JVC ont été testés sur des matrices aléatoirement générées variant de creuses à denses et dans des simulations de pistage sur des scénarios en temps réel. Les résultats montrent qu'avec des matrices denses, l'algorithme de Junker et Volgenant est plus rapide que l'algorithme d'enchère, mais qu'avec des matrices creuses de dimensions entre 25×25 et 200×200 , l'algorithme d'enchère est plus rapide. En général, les matrices d'associations produites lors des simulation de pistage sur des scénarios en temps réel sont creuses. Donc, l'algorithme d'enchère est l'algorithme conseillé pour des situations de pistage à balayages simples où le nombre de contacts varie entre 25 et 200.

Le deuxième chapitre portait sur l'algorithme d'association à balayages multiples de Deb et al. qui utilise la relaxation lagrangienne pour convertir un problème de dimension S en une série de problèmes à deux dimensions. Les résultats du premier chapitre mènent à vérifier si l'algorithme d'enchère est aussi rapide avec le balayage multiple. Les résultats montrent que les avantages de l'algorithme d'enchère en balayage simple sont perdu en balayage multiple. JVC est donc le meilleur algorithme pour ce cas. Des tests montrent que de passer du balayage simple au balayage multiple améliore la qualité de l'association sauf si la probabilité de détection du radar est faible.

Dans le troisième chapitre, l'algorithme à balayages multiples de Deb et al. a été appliqué au balayage multiple. La méthode de profondeur temporelle maximale de Kirubarajan et al. a été comparée à la méthode de Opitz pour produire une estimation provisoire efficace. Les conclusions sont qu'en général, la méthode de Opitz est significativement plus rapide et que dans beaucoup de situations, la méthode de Opitz présente une qualité d'association aussi bonne que la méthode de profondeur temporelle maximale.

Les résultats montrent aussi que dans le cas du scénario 1, beaucoup plus de temps de calcul est économisé en choisissant le meilleur algorithme à deux dimensions qu'en choisissant la méthode d'estimation la plus rapide (Tableaux 2.1 et 3.1). Autrement dit, un utilisateur qui désire utiliser la méthode de profondeur temporelle maximale pour faire l'estimation va améliorer de beaucoup la vitesse de son algorithme en utilisant JVC comme algorithme d'association à deux dimensions.

Dans l'article où Opitz présente sa méthode [27], il n'y a pas de test pour permettre au lecteur de connaître l'ordre de grandeur de l'amélioration qu'apporte sa méthode comparativement à la méthode de profondeur temporelle maximale. Avec les résultats présentés dans ce mémoire, les avantages et désavantages d'utiliser la méthode de Opitz sont maintenant appuyés par des résultats provenant d'expériences sur des simulations de pistage.

Il serait intéressant dans un projet future d'augmenter la difficulté des tests : plus de senseurs, plus de types de senseurs, plus de balayages, plus de cibles. Avec de tels tests il serait possible de savoir quelles sont les limites de l'algorithme à balayages multiples et à senseurs multiples. Les recherches dans le domaine du pistage tentent de déterminer le meilleur compromis entre la qualité d'association et la rapidité d'exécution. Il serait donc particulièrement intéressant de savoir à quel moment l'intégration d'un nouveau senseur ou l'ajout d'un balayage n'apporte plus une amélioration significative à la qualité de l'association par rapport au temps de calculs supplémentaire.

BIBLIOGRAPHIE

- [1] S. M. Andrijich and L. Caccetta, "Solving the multisensor data association problem," *Nonlinear Analysis*, vol. 47, no. 8, pp. 5525–5536, 2001.
- [2] E. Blasch, "Information fusion for decision making," in *Proceedings of the NATO advanced study institute on data fusion for situation monitoring, incident detection, alert and response management*, Yerevan, Arménie, August 2003, pp. 3–22.
- [3] S. S. Blackman, *Multiple-Target Tracking with Radar Application*. Norwood, MA : Artech House, 1986.
- [4] J. C. Nash, "The (dantzig) simplex method for linear programming," *Computing in Science and Engineering*, vol. 2, no. 1, pp. 29–31, 2000.
- [5] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistic Quarterly*, vol. 2, pp. 83–97, 1955.
- [6] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, pp. 325–340, 1987.
- [7] V. Klee and G. J. Minty, "How good is the simplex algorithm?" in *Inequalities*, O. Shisha, Ed. New York : Academic Press, 1972, vol. 3, pp. 159–175.
- [8] M. S. Hung and W. Rom, "Solving the assignment problem by relaxation," *Operation Research*, vol. 28, pp. 969–982, 1980.
- [9] L. F. McGinnis, "Implementation and testing of prime-dual algorithm for the assignment problem," *Operation Research*, vol. 31, pp. 277–291, 1983.
- [10] B. P. Bertsekas, "A new algorithm for the assignment problem," *Mathematical programming*, vol. 21, pp. 152–171, 1981.
- [11] J. Munkres, "Algorithms for the assignment and transportation problems," *Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

- [12] D. B. Malkoff, "Evaluation of the jonker-volgenant-castanon (jvc) assignment algorithm for track association," in *SPIE proceedings series*, vol. 3068, Orlando, FL, USA, July 1997, pp. 228–239.
- [13] N. Tomizawa, "On some techniques useful for solution of transportation network problems," *Networks*, vol. 1, no. 2, pp. 173–194, 1971.
- [14] O. E. Drummond, D. A. Castañon, and M. Bellovin, "Comparison of 2-d assignment algorithms for sparse, rectangular floating point, cost matrices," in *Journal of the SDI panels on Tracking*, vol. 4, Alexandria, VA, December 1990, pp. 81–97.
- [15] P. A. Stephens and K. N. R., "A sparse matrix technique for the munkres algorithm," in *Summer computation simulation computation*, Toronto, Canada, July 1979, pp. 3–22.
- [16] B. P. Bertsekas, "The auction algorithm : A distribution relaxation method for the assignment problems," *Annals of Operations Research*, vol. 14, pp. 105–123, 1988.
- [17] B. P. Bertsekas, D. A. Castañon, and H. Tsaknakis, "Reverse auction and the solution of inequality constrained assignment problems," *SIAM Journal on Optimisation*, vol. 3, pp. 268–299, 1993.
- [18] B. P. Bertsekas, "Auction algorithms for network flow problems : A tutorial introduction," *Journal of Computation Optimization and its Applications*, vol. 1, no. 1, pp. 7–66, 1992.
- [19] M. Levedahl, "Performance comparison of 2-d assignment algorithms for truth objects to measured tracks," in *SPIE proceedings series*, vol. 4048, Orlando, FL, USA, April 2000, pp. 380–389.
- [20] R. A. Singer, R. G. Sea, and K. B. Housewright, "Derivation and evaluation of improved tracking filters for use in dense multi-target environments," *IEEE Transactions on Information Theory*, vol. 20, pp. 423–432, 1974.
- [21] D. B. Ried, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.
- [22] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom, "A generalized s-d assignment algorithm for multisensor-multitarget state estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 2, pp. 523–538, 1997.
- [23] A. B. Poore and A. J. Robertson, "A new lagrangian relaxation based algorithm for a class of multidimensional assignment problems." *Computational Optimisation and Applications*, vol. 8, no. 2, pp. 129–150, 1997.

- [24] C. Lemarechal, "Lagrangian relaxation," in *Computational Combinatorial Optimization, Optimal or Provably Near-Optimal Solutions [based on a Spring School]*. London, UK : Springer-Verlag, 2001, pp. 112–156.
- [25] T. Kirubarajan, H. Wang, Y. Bar-Shalom, and K. Pattipati, "Efficient multisensor fusion using multidimensional data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 2, pp. 386–398, 2001.
- [26] H. Wang, T. Kirubarajan, and Y. Bar-Shalom, "Large scale air traffic surveillance using imm estimators with assignment." *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. 255–266, 1999.
- [27] F. Opitz, "Fusion of active phased array radars within a quick reaction time using multidimensional data association," in *ISIF Proceeding 2003*, Uml, Germany, 2003, pp. 342–348.

Annexe A

RÉSUMÉ DES RECHERCHES EN BALAYAGES SIMPLES

TAB. A.1. Tableau résumant quelques recherches en ordre chronologique sur le balayage simple

Article	Algorithmes	Type de test	Résultats
[7]	Méthode du simplex	Analyse de complexité	La complexité de la méthode du simplex est d'ordre exponentielle
[10]	Enchère, Hongrois et méthode hybride Enchère-hongrois	Matrices non creuses aléatoires	hybride Enchère-hongrois plus rapide
[6]	JV, Enchère et Hongrois	Matrices creuses et non creuses aléatoires	JV est plus rapide dans les deux cas
[14]	JVC et Munkres	Matrices creuses	JVC est plus rapide
[17]	JVC et plusieurs versions d'enchère	Matrices creuses et non creuses aléatoires	JVC est plus rapide dans la plupart des cas. Enchère est une méthode généralement simple et efficace
[12]	JVC, plus proche voisin et l'algorithme de compétition de Malkoff	Simulation de pistage en temps réel	JVC est plus rapide, mais l'algorithme de compétition fait moins d'erreurs lorsque le temps de calcul est limité
[19]	JVC, enchère, Munkres, plus proche voisin et force brute	Matrices creuses et non creuses aléatoires à entrées réelles ou entières	JVC est plus rapide pour la plupart des tests