HAL
open science

# Une méthode de Branch and Bound par Intervalles appliquée à la résolution en vitesse de conflits aériens

Nicolas Durand, Jean-Marc Alliot

## To cite this version:

Nicolas Durand, Jean-Marc Alliot. Une méthode de Branch and Bound par Intervalles appliquée à la résolution en vitesse de conflits aériens. ROADEF 2011, 12ème congrès annuel de la Société française de Recherche Opérationnelle et d'Aide à la Décision, Mar 2011, St-Etienne, France. hal-00940896

**HAL Id: hal-00940896**

**https://enac.hal.science/hal-00940896**

Submitted on 24 Apr 2014

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

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

# Une méthode de Branch and Bound par Intervalles appliquée à la résolution en vitesse de conflits aériens

Nicolas Durand[1,2], Jean-Marc Alliot[1,2]

[1] DSNA/DTI R&D[‡], 7, av Edouard Belin, BP 54005 F-31055 Toulouse Cedex 4, France
durand,alliot@tls.cena.fr
[2] APO-IRIT[§], Université Paul Sabatier ; 118, route de Narbonne, F-31062 Toulouse Cedex 9, France

**Mots-Clés** : *IBBA, résolution de conflits*

## 1 Introduction

Deux avions en croisière à la même altitude séparés de moins de 5 miles nautiques sont dits en conflit. Le rôle du contrôleur aérien est d'éviter les situations de conflits en anticipant des manoeuvres de séparation (changement de cap ou de niveau de vol de l'un des deux avions). En modifiant légèrement les vitesses des avions, on peut résoudre les conflits aériens en amont, et ce à l'insu du contrôleur qui n'est pas perturbé par ce prétraitement. Le projet ERASMUS [BDG09] qui a introduit ce concept se base actuellement sur un algorithme évolutionnaire développé dans les années 90 sur le simulateur CATS [GDA01]. Le problème de résolution de conflits est un problème très combinatoire. La littérature ne propose que deux approches efficaces pour résoudre de façon centralisée des problèmes de grande taille (plus d'une vingtaine d'avions). L'approche de [PFB02] utilise la programmation linéaire mixte, mais requiert des hypothèses fortes sur les trajectoires (vitesses constantes, manoeuvres exécutées en même temps). L'approche par algorithme évolutionnaire [DA98] est plus ancienne et permet de prendre en compte des trajectoires issues d'un simulateur de trafic. Ce résumé a pour but de présenter un algorithme de Branch and Bound par Intervalle, tel que décrit par exemple par [Han92], adapté à un problème de résolution de conflits en vitesse. Un problème "jouet" est utilisé pour tester l'algorithme. $n$ avions sont disposés sur un cercle et se dirigent vers le centre du cercle.

## 2 Modélisation du problème

Les avions volent à vitesse et altitude constante, et on peut modifier à l'instant 0 leur vitesse en restant à l'intérieur d'un intervalle autour de la vitesse nominale de l'avion (par exemple dans une fenêtre allant de $-6\%$ à $+3\%$ si l'on reprend les hypothèses du projet ERASMUS). On veut minimiser le changement de vitesse de chaque avion, tout en respectant les contraintes de séparation entre les avions. Si l'on note pour chaque avion $i$, $v_i$ sa vitesse, et $m_i$ le pourcentage de modification de sa vitesse, on veut minimiser $f(m_1, m_2, ..m_n) = \sum_{i=1}^{n} m_i^2$, tout en respectant les contraintes de

[‡]Direction des Services de la Navigation Aérienne, Direction de la Technique et de l'Innovation, Domaine R & D
[§]Equipe Algorithmes Parallèles et Optimisation - Institut de recherche en Informatique de Toulouse

séparation des trajectoires. La fonction d'inclusion naturelle de $f$ utilisée pour l'algorithme de Branch an Bound peut s'exprimer : $F(M_1, M_2, ..M_n) = \sum_{i=1}^{n} M_i^2$ où $M_i$ est un intervalle de modification de vitesse possible pour l'avion $i$.

On veut garantir que pour tout couple d'avions $(i, j)$ et tout temps $t \geq 0$, les positions $\vec{x_i}(t)$ et $\vec{x_j}(t)$ sont séparées d'une distance $d$ appelée norme de séparation. Si on note $\vec{p_i}$ et $\vec{p_j}$ les positions des avions $i$ et $j$ à $t = 0$ alors $\vec{x_i}(t) = \vec{p_i} + t\, m_i \vec{v_i}$ et la contrainte s'exprime : $(\vec{x_i}(t) - \vec{x_j}(t))^2 - d^2 \geq 0$. Il ne reste alors plus qu'à résoudre une équation du second degré pour vérifier qu'il n'y a pas de racine positive. L'algorithme de Branch and Bound doit être capable de déterminer si un vecteur d'intervalles :

- satisfait les contraintes de séparation : il faut que pour tous les couples $(i, j)$ et les modifications de vitesses dans les intervalles $M_i$ et $M_j$, les avions soient séparés. On peut alors réduire $(M_1, M_2, ..., M_n)$ au point qui minimise $f$ dans $(M_1, M_2, ..., M_n)$.
- ne satisfait pas les contraintes de séparation : il suffit que pour un couple $(i, j)$ et toute modification de vitesse dans les intervalles $M_i$ et $M_j$, les avions $i$ et $j$ soient toujours en conflit. On peut supprimer le vecteur d'intervalles concerné.
- n'est pas dans l'un des deux précédents cas : il faut alors conserver le vecteur d'intervalles tout entier, il sera subdivisé en deux nouveaux vecteurs d'intervalles.

Vérifier la contrainte de séparation pour deux intervalles $M_i$ et $M_j$ est assez simple. Il s'agit de mesurer la distance entre deux segments dont la taille augmente linéairement avec le temps et qui se déplacent à vitesse constante sur deux droites représentant les trajectoires des avions. La comparaison des positions extrêmes des segments et un test de croisement permettent de calculer rapidement les distances de séparation minimales et maximales atteintes.

## 3 Résultats

L'algorithme de Branch and Bound par Intervalles permet de résoudre efficacement des instances de problème allant jusqu'à six ou sept avions. Au-delà, les temps de calcul deviennent très importants. Plusieurs stratégies ont été testées pour calculer l'estimateur du meilleur point atteint et aussi pour classer les vecteurs d'intervalles dans la liste des intervalles restant. Il reste de nombreuses pistes à explorer pour essayer d'atteindre des performances au moins égales à celles des autres approches citées en introduction, on peut notamment s'intéresser à des stratégies spécifiques de découpage et de classement des intervalles qui favorisent l'emergence de la solution optimale.

## Références

[BDG09] Deidre Bonini, Carole Dupré, and Géraud Granger. How erasmus can support an increase in capacity in 2020. In *Proceedings of the 7th International Conference on Computing, Communications and Control Technologies : CCCT 2009, Orlando, Florida*, 2009.

[DA98] Nicolas Durand and Jean-Marc Alliot. Genetic crossover operator for partially separable functions. In *Genetic Programming*, 1998.

[GDA01] Geraud Granger, Nicolas Durand, and Jean-Marc Alliot. Optimal resolution of en route conflicts. In *ATM 2001*, 2001.

[Han92] E. Hansen. *Global optimization using interval analysis.* Dekker, New-York, 1992.

[PFB02] L. Pallottino, E. Feron, and A. Bicchi. Conflict resolution problems for air traffic management systems solved with mixed integer programming. *IEEE Transactions on Intelligent Transportation Systems*, 3(1) :3–11, 2002.