



HAL
open science

Optimisation par fusion et fission, application au problème du découpage aérien européen

Charles-Edmond Bichot, Jean-Marc Alliot, Nicolas Durand, Pascal Brisset

► **To cite this version:**

Charles-Edmond Bichot, Jean-Marc Alliot, Nicolas Durand, Pascal Brisset. Optimisation par fusion et fission, application au problème du découpage aérien européen. *Journal Européen des Systèmes Automatisés (JESA)*, 2004, 38 (9-10), pp 1141-1173. 10.3166/jesa.38.1141-1173 . hal-00934545

HAL Id: hal-00934545

<https://hal-enac.archives-ouvertes.fr/hal-00934545>

Submitted on 25 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.

Optimisation par fusion et fission, application au problème du découpage aérien européen

Charles-Edmond BICHOT

Jean-Marc ALLIOT Nicolas DURAND Pascal BRISSET

*LOG (Laboratoire d'Optimisation Globale) CENA/ENAC
7, avenue Edouard Belin 31055 Toulouse cedex, FRANCE
bichot@recherche.enac.fr*

1 Introduction

Le **LOG** est un laboratoire d'informatique spécialisé dans l'optimisation et dont le domaine d'application principal est le trafic aérien. Une des applications actuelles du laboratoire est le projet **FABOP** qui consiste en l'étude du redécoupage du ciel aérien européen. Le contrôle aérien en Europe tel qu'il existe aujourd'hui est presque exclusivement national. Mais ce contrôle est appelé à évoluer vers une européanisation croissante. Les contrôleurs aériens surveillent un espace limité, appelé secteur de contrôle. Cependant, ils sont formés pour pouvoir prendre en charge différents secteurs de manière alternative. Nous appelons l'ensemble des secteurs sur lesquels ils sont habilités à travailler, une zone de qualification. Or ces zones de qualification respectent scrupuleusement les frontières nationales. Nous cherchons maintenant un redécoupage de ces zones de qualification faisant abstraction des frontières nationales et cherchant à augmenter la sûreté comme la fluidité de contrôle aérien.

Notre étude consiste à former des « paquets » de secteurs à partir de la connaissance des flux d'avions entre les secteurs. Le ciel européen est découpé en $nb_tot_secteurs = 762$ secteurs regroupés dans $nb_de_zones = 36$ zones de qualification. Ces données sont approximatives car elles fluctuent beaucoup dans le temps, et correspondent à des regroupements de zones pour les secteurs de niveaux inférieurs¹. Pour évaluer la pertinence d'un découpage du ciel, nous utilisons une fonction énergie que nous étudierons plus en détail à la section 6. Ainsi, notre problème consiste à trouver la configuration d'énergie minimale du découpage de l'espace aérien européen en zones de qualification. Notre espace de recherche est donc inclus dans l'ensemble des partitions en p parties (ou p zones de qualification) d'un ensemble à n éléments (ou encore n secteurs). Nous pouvons donc borner par valeur supérieure la taille de notre espace de recherche [BA05a] [Gia04]. Cette majoration du nombre de solutions est donnée par le nombre de Stirling de seconde espèce, qui vaut :

$$N_{n,p} = \sum_{k=0}^p (-1)^k \frac{(p-k)^n}{k! (p-k)!}$$

¹Les secteurs de niveaux inférieurs sont principalement les secteurs de contrôle voisins des aéroports

Ce qui représente pour le découpage du ciel aérien européen plus de 36^{762} possibilités. De plus, comme nous le verrons dans la section (7), ce problème de découpage de l'espace aérien, quand il utilise la fonction énergie définie en section (6), est NP-complet. Ce sont deux arguments qui nous poussent à utiliser des méthodes d'optimisation par métaheuristiques [P.S02], particulièrement adaptées aux problèmes NP-difficiles, et donc NP-complets, et aux espaces de recherche très grands.

Nous avons déjà appliqué les méthodes du recuit simulé et des colonies de fourmis à ce problème [BA05a] [Bic04]. Pour compléter notre étude, nous cherchons d'autres méthodes d'optimisation utilisant des heuristiques. Les similitudes entre notre problème et l'organisation structurale des atomes nous ont incités à rechercher un modèle d'optimisation inspiré de la physique nucléaire. Notre idée est de comparer les secteurs aux nucléons qui composent les atomes et d'assimiler ces derniers aux zones de qualification. Puis nous nous inspirons des interactions régissant une partie de la physique nucléaire pour assembler nos secteurs en zones de qualification, comme le font les nucléons et les atomes par les mécanismes de fusion et de fission nucléaires.

Nous allons vous présenter ce modèle d'optimisation par fusion et fission par étapes successives. Dans un premier temps (2), nous aborderons de manière très simple les éléments de physique nucléaire utilisés par notre modèle. Puis nous parlerons de la mise en place de cette méthode (3). Cette mise en place sera suivie par la structure de la méthode où nous détaillerons les algorithmes utilisés (4), et les fonctions utilisées par ces algorithmes (5). Nous reparlerons de manière plus détaillée dans la section (6), du problème du découpage de l'espace aérien. Puis nous démontrerons que ce problème est NP-complet dans la section suivante (7). Alors nous présenterons (8) la façon dont nous avons concrètement adapté notre méthode d'optimisation à notre problématique de découpage de l'espace aérien. Dans la suite de cette présentation, il nous faudra aborder de manière succincte le mécanisme de percolation (9) ainsi que la méthode d'optimisation par colonies de fourmis (10), qui nous ont servis dans notre adaptation du mécanisme de fission. Puis nous présenterons notre adaptation de la méthode d'optimisation par recuit simulé au découpage aérien (11). Enfin, nous présenterons les résultats comparant les méthodes d'optimisation par recuit simulé, colonies de fourmis et fusion/fission (12).

2 Notions de physique nucléaire

Le but de ce paragraphe est d'exposer avec un esprit simplificateur les notions de physique nucléaire qui nous ont servi à élaborer notre modèle. Nous ne prétendons en aucun cas être des spécialistes en physique nucléaire. Le livre du professeur Daniel Blanc sur la physique nucléaire [Bla03] et le [site internet](#)² du chargé de recherche [David Calvet](#), hébergé par l'IN2P3³, permettent pour ceux qui le souhaiteraient d'aborder ce domaine de manière simple et claire.

L'atome est la plus petite partie d'un corps pur. Il est formé d'électrons et de nucléons. Les nucléons forment le noyau atomique et les électrons gravitent autour. Il existe deux sortes de nucléons, les protons de charge positive et les neutrons de charge neutre. Un atome contient donc trois sortes de particules : des

² « *Voyage au cœur de la matière...* », <http://voyage.in2p3.fr/index.html>

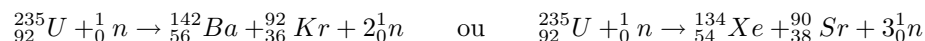
³Institut National de Physique Nucléaire et de Physique des Particules

électrons de charge négative, des protons et des neutrons. Le noyau atomique se situe au centre de l'atome, il est mille fois plus petit que ce dernier, mais contient presque la totalité de sa masse. La cohésion du noyau atomique est assurée par l'interaction forte, qui attire les nucléons entre eux et empêche ainsi les protons de se repousser.

Les atomes sont classés en fonction de leur nombre atomique Z dans le tableau périodique des éléments. Z est égal au nombre d'électrons lui-même égal au nombre de protons d'un atome. Les isotopes sont des atomes qui ont le même nombre de protons Z , mais un nombre de neutrons N différent. A est le nombre de masse, c'est le nombre de nucléons. Lorsqu'un noyau n'est pas radioactif, un isotope est dit stable, sinon il est dit instable. Les isotopes stables sont représentés en noir dans la figure 1. Et il y a aussi des valeurs « magiques » de N et de Z pour lesquelles la stabilité du noyau est maximale. Le symbole du noyau est A_ZM où M est le symbole de l'élément, A le nombre de masse et Z le nombre de protons. Précisons deux autres notations : un neutron seul s'écrit 1_0n , et un proton seul, 1_1p .

À l'heure actuelle, nous connaissons quatre interactions. Elles interviennent dans tous les phénomènes de l'univers. La gravitation est l'interaction la moins intense, c'est une force attractive entre plusieurs corps qui dépend de leur masse. L'interaction électromagnétique est une force répulsive ou attractive qui agit sur des éléments possédant une charge électrique. L'interaction faible agit sur toutes les particules. L'interaction forte est une force de cohésion attractive très intense entre les quarks⁴.

Certains atomes ont un noyau instable qui émet des rayonnements. On dit alors que ces atomes sont *radioactifs*. La radioactivité est la désintégration spontanée des noyaux. Contrairement à ce que certaines personnes croient, ce phénomène est tout à fait naturel. Les interactions fortes, faibles et électromagnétiques sont à l'origine de différentes formes de radioactivité, mais nous ne détaillerons ici que la radioactivité produite par l'interaction forte, et plus particulièrement la fission spontanée. Le phénomène de la fission spontanée ne devient important que lorsque l'on est en présence de noyaux lourds. Un noyau est dit lourd lorsqu'il comporte un grand nombre de nucléons. Lors de la *fission* d'un atome, son noyau se scinde en plusieurs fragments (généralement deux), en même temps que sont émis plusieurs neutrons rapides. Mais la fission peut aussi se produire lorsqu'un atome a été violemment heurté par un neutron. Exemple de fission : l'uranium 235 sous l'impact d'un neutron d'énergie suffisante se transforme en uranium 236 hautement instable, qui se scinde :



La *fusion* de deux atomes apparaît lorsque les noyaux de ces deux atomes sont suffisamment proches l'un de l'autre pour fusionner, c'est-à-dire pour former un unique noyau. Pour fusionner, les atomes doivent être dans un milieu très chaud appelé plasma thermique. Ils auront alors des vitesses suffisamment élevées pour pouvoir fusionner avant d'être séparés par la force électromagnétique répulsive. Cette répulsion qui empêche habituellement les noyaux de fusionner, est due aux charges électriques positives des noyaux. Quand aucun état stable n'existe, il n'est pas toujours possible de provoquer la fusion de deux noyaux, par exemple,

⁴En particulier, les nucléons sont constitués de trois quarks.

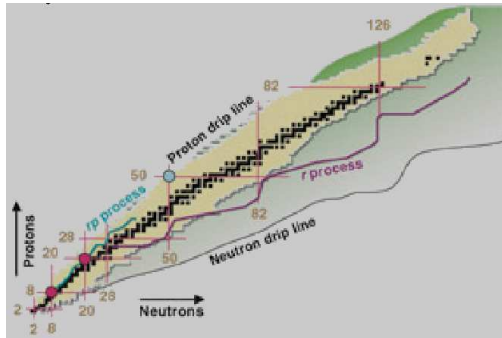


FIG. 1 – Diagramme de stabilité des noyaux en fonction de N et de Z. Carrés noirs : noyaux stables.

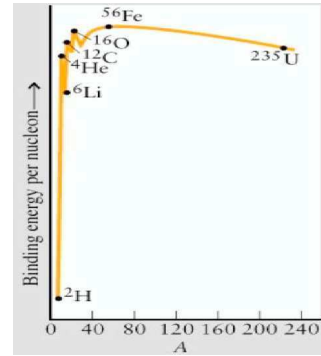
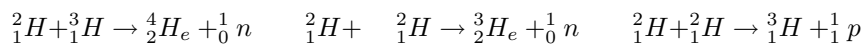


FIG. 2 – Énergie moyenne de liaison par nucléon

${}^4_2\text{He} + {}^4_2\text{He}$ ne peuvent pas fusionner ensemble. Mais voici quelques exemples de réactions de fusion existantes :



Nous pouvons voir à travers cet exemple que deux configurations identiques d'atomes peuvent engendrer des réactions de fusion différentes.

La masse d'un atome est inférieure à la somme des masses des nucléons isolés. Cette différence de masse correspond à la quantité d'énergie qu'il faudrait fournir au noyau pour le dissocier en ces constituants séparés. On appelle cette quantité l'énergie de liaison du noyau. La figure 2 donne la variation de l'énergie moyenne de liaison par nucléon en fonction du nombre de masse A.

3 Mise en place de la méthode

Comme nous l'avons vu dans l'introduction, nous cherchons des méthodes d'optimisation à appliquer à notre problématique de découpage du trafic aérien européen. Nous avons aussi vu que les caractéristiques de notre problème nous conduisent à utiliser des méthodes d'optimisation par métaheuristique. Nous entendons par métaheuristique, une méthode d'optimisation « difficile » tel que la décrit [Patrick Siarry](#) dans le préambule de son livre [\[DPST03\]](#), mais aussi [\[Hao99\]](#). Notre problème est un problème d'optimisation stochastique, NP-complet, possédant une ou plusieurs fonctions de coût à minimiser. Bien entendu, il ne s'agit pas à proprement parler de trouver *le* minimum global, mais plutôt un meilleur rapport entre temps de calcul et optimum atteint, que celui trouvé avec d'autres métaheuristicques classiques [\[Che99\]](#).

Pour répondre à notre problème, nous voulions trouver une méthode proche d'une métaheuristique permettant de regrouper des éléments dans des ensembles. Nous avons alors comparé les ensembles aux atomes et les éléments aux nucléons. Soit *nb_atomes* le nombre de ces ensembles que nous voulons obtenir. Alors que les méthodes classiques comme le recuit simulé agissent en déplaçant un élément dans un nombre fixe d'ensembles, nous voulions pouvoir déplacer un nombre quelconque d'éléments vers un nombre d'ensembles non fixe. En effet, nous nous sommes rendu compte que des aberrations pouvaient survenir pour

un nombre donné d'ensembles, mais être résolues avec un nombre proche de ce dernier. Il est vrai que nous aurions cependant pu étendre le voisinage d'une méthode classique à un ensemble d'éléments, mais cela ne correspond généralement pas aux principes de telles méthodes qui préconisent des modifications élémentaires plus que globales. À ce stade, il est important de faire remarquer que le nombre de variables élémentaires est fixé dès le départ pour les problèmes que nous cherchons à optimiser, soit $nb_tot_nucléons$ ce nombre.

Comme c'est souvent le cas pour les métaheuristiques, nous nous sommes donc tournés vers un exemple d'optimisation issu de la nature. Les mécanismes de fusion et fission nucléaire, de par leurs actions sur des ensembles d'éléments, les atomes, comme sur des particules, les nucléons, répondaient à notre problématique. Nous avons donc cherché à nous inspirer de manière très simple de ces mécanismes.

Nous avons vu précédemment en section 2, que la fusion comme la fission font interagir les atomes entre eux et plus particulièrement les nucléons entre eux. Le cœur des atomes est composé de nucléons, donc si nous faisons abstraction des électrons gravitant autour du noyau, nous pouvons considérer que les atomes sont juste des ensembles de nucléons. Pour simplifier encore le modèle, nous ne tiendrons pas compte de la différence entre neutrons et protons dans les réactions, seul leur qualité de nucléon va nous intéresser.

Nous avons vu que la température jouait un rôle important dans les mécanismes de fusion/fission. Lorsque la température diminue, l'état des particules devient plus stable, et l'on pourrait imaginer que les atomes convergent vers un état où leur énergie de liaison est maximale, c'est à dire vers l'atome de fer comme nous le montre la figure 2. Nous appelons état, une configuration des nucléons répartis en atome, qui soit solution du problème. À l'inverse, lorsque la température augmente, les atomes peuvent devenir très gros comme très petits selon qu'un processus de fusion ou de fission en chaîne a commencé.

Dans la nature, les atomes de fer possèdent l'énergie moyenne de liaison par nucléon la plus forte. Nous allons considérer dans notre modèle, qu'un état possédant nb_atomes atomes correspond à l'état où l'énergie moyenne de liaison par nucléon est la plus forte. En effet, puisque notre nombre de nucléons est fixé, seul le nombre d'atomes peut nous permettre de faire varier le nombre de nucléons par atomes. Ainsi, une configuration possédant nb_atomes atomes devra naturellement être de coût minimal. Ce sera donc vers ce nombre d'atome que l'algorithme convergera. Notre fonction de coût devra donc favoriser les configurations avec nb_atomes atomes.

Les réactions de fusion et de fission obéissent à des règles bien précises, mais facilement généralisables. Lors d'une fusion deux atomes peuvent s'assembler en libérant ou non un petit nombre de nucléons. Ces nucléons une fois libérés viennent se rattacher à d'autres atomes dont certains peuvent ainsi devenir radioactifs. Lors d'une fission un atome se scinde en deux parties en pouvant libérer ou non un petit nombre de nucléons. Si leur énergie est suffisante, ces nucléons peuvent à leur tour engendrer la fission d'autres atomes voisins, et ainsi engendrer une fission en chaîne. Sinon, ils se rattachent à d'autres atomes par fusion. Ils nous faut donc modéliser ces procédures de fusion/fission en formulant des règles auxquelles elles obéissent. Nous verrons à la section 4 ce mécanisme plus en détail.

Enfin, comme dans toute méthode d'optimisation par métaheuristiques, nous cherchons un extrémum global correspondant à un état donné du problème.

Plus particulièrement nous cherchons ici un minimum. Nous devons donc disposer d'une fonction coût qui permette de calculer l'énergie d'un état. Notre méthode doit retenir le meilleur état précédemment atteint, afin de pouvoir disposer constamment d'une comparaison avec le dernier état obtenu. C'est ce meilleur état atteint qui sera renvoyé à la fin de l'optimisation. Notre méthode de par sa nature va changer le nombre d'atomes des différents états, pourtant nous cherchons l'énergie minimale d'un état pour un nombre précis d'atomes, nb_atomes . Notre méthode va chercher un tel état en faisant varier le nombre d'atomes autour de cette valeur. Ainsi, l'un des avantages de cette méthode est qu'elle va donner des résultats satisfaisants pour des états voisins en nombre d'atomes.

Cependant la fonction de coût doit pénaliser les configurations d'état dans lesquelles le nombre d'atomes est éloigné du nombre nb_atomes d'atomes souhaités. Nous voulons que cette fonction de coût imite la courbe de la figure 2 et que la configuration de coût minimale (nous cherchons à minimiser l'énergie et non à la maximiser comme c'est le cas de l'énergie de liaison) corresponde à l'atome de fer, c'est-à-dire à un état possédant nb_atomes atomes. Pour adapter la fonction de coût, il suffit de la multiplier par une fonction proportionnelle à l'écart entre le nombre d'atomes d'un état et nb_atomes .

4 Structure de la méthode

Nous distinguons deux étapes dans notre méthode. Une première étape d'initialisation qui va partir d'un état qui n'est pas solution du problème. Dans cet état ne sont présents que les nucléons et pas encore les atomes. Cette initialisation a pour but de créer un état solution du problème, c'est-à-dire ne laissant plus aucun nucléon seul, et possédant au plus une certaine énergie. L'initialisation sera suivie d'une étape d'optimisation proprement dite. Celle-ci partira de l'ensemble d'atomes issus de l'initialisation et fera évoluer l'état vers un état de plus basse énergie. Pour cela, l'état de départ de l'optimisation sera considéré comme un plasma thermique à très haute température. La température descendra progressivement, en même temps que se réaliseront les fusions et fissions entre atomes. Quand la température sera trop basse, nous réitérerons cette optimisation à partir du meilleur état précédemment trouvé.

Mais avant de décrire ces processus, nous devons expliquer le mécanisme de règle dont nous avons parlé précédemment et qui sera utilisé par ces deux processus.

4.1 Le mécanisme de règle

Le mécanisme de règle va nous servir à décrire les fusions et fissions et particulièrement donne le nombre de nucléons éjectés par elles. Dans le mécanisme de règle, nous classifions les réactions différemment selon que ce sont des fissions ou des fusions. Dans le cas d'une fission, nous nous intéressons au nombre de nucléons présents dans l'atome, et en fonction de ce nombre nous utiliserons une règle différente. Pour une fusion, la règle sera indexée par les nombres de nucléons des deux atomes en présence. Dans les règles sont comprises des listes de probabilités. Pour chacune de ces listes, la somme des probabilités vaut un. Chaque probabilité est strictement comprise entre zéro et un. Chaque probabilité

de la liste correspond au nombre de nucléons indépendants qui vont apparaître en plus de l'atome fusionné ou des deux nouveaux atomes résultant de la fission. Nous avons inclus dans les règles un mécanisme d'apprentissage. Lorsqu'une des probabilités d'une règle s'est révélée satisfaisante, par exemple si le nouvel état atteint grâce à elle est moins énergétique, nous augmentons cette probabilité au détriment des autres probabilités de cette règle. Nous dirons que la règle est acceptée positivement. À l'inverse, si cette probabilité ne s'est pas montrée satisfaisante, nous la pénalisons au profit des autres probabilité de cette règle. Nous dirons que la règle est acceptée négativement. Il faut cependant que toutes les probabilités soient strictement positives, pour toujours laisser la possibilité d'avoir un nombre de nucléons différents. Nous faisons cependant remarquer que ce système de règle pourrait être utilisé différemment.

Prenons un exemple, soit la règle $\left\{ \begin{array}{l} \text{fusion} : \{p_{u0}; p_{u1}; p_{u2}; p_{u3}\} \\ \text{fission} : \{p_{i0}; p_{i1}; p_{i2}; p_{i3}\} \end{array} \right\}$ correspondant à $x > 3$ nucléons. Nous avons $\forall i, 0 < p_i < 3$ et $\sum_i p_i = 1$. Dans cette règle, la probabilité qu'une fusion ne se sépare d'aucun nucléon est p_{u0} , et p_{i0} lors d'une fission. Et la probabilité de se séparer de trois nucléons est p_{u3} pour une fusion et de p_{i3} pour une fission. Dans le cas où une fusion est réalisée et rejette un nucléon, et où l'état obtenu est de plus basse énergie que le précédent, la nouvelle règle pour la fusion sera $\left\{ p_{u0} - \frac{\text{pas_de_règle}}{3}; p_{u1} + \text{pas_de_règle}; p_{u2} - \frac{\text{pas_de_règle}}{3}; p_{u3} - \frac{\text{pas_de_règle}}{3} \right\}$, avec *pas_de_règle* un paramètre de variation de la règle pour que celle-ci apprenne au fur et à mesure, Dans le cas où l'énergie serait supérieure, la nouvelle règle pour la fusion serait le même ensemble que précédemment en remplaçant les additions par des divisions et inversement. La liste des probabilités pour la fission ne changeant pas.

4.2 Le processus d'initialisation

Le processus d'initialisation peut être vu comme une version simplifiée du processus d'optimisation. L'initialisation démarre sur un état E_{init} constitué uniquement de *nb_tot_nucléons* nucléons indépendants, chacun dans un atome. soit l'état E_i pendant le processus d'initialisation. Un atome ou un nucléon sont tirés au hasard. Une fonction de probabilités détermine la probabilité qu'il y ait fusion ou fission. Cette fonction a pour argument le nombre de nucléons de l'atome ou du nucléon sélectionné, l'énergie de l'état E_i , le nombre de nucléons restants, le nombre d'atomes de l'état et le nombre moyen de nucléons par atomes. À partir de ce choix, une fusion ou une fission va être réalisée. Si c'est une fusion, un atome ou un nucléon va être sélectionné en fonction de critères énergétiques et de distance au premier atome/nucléon. Nous entendons par distance un critère particulier au problème permettant de sélectionner les atomes. Par exemple, pour notre problématique il s'agira du nombre d'avions passant entre deux atomes (entre deux zones de qualification). La règle de fusion correspondant à la somme du nombre de nucléon des deux atomes est alors sélectionnée. Puis la fusion est réalisée. Elle peut libérer des nucléons qui sont alors fusionnés avec d'autres atomes ou nucléons. Si c'est une fission, la règle correspondant au nombre de nucléons de l'atome est utilisée. Des nucléons peuvent être éjectés. Dans ce cas d'initialisation, nous considérons que leur énergie est trop faible pour engendrer d'autres fissions. Ces nucléons vont donc fusionner avec d'autres atomes/nucléons. Dans tous les cas le nombre d'atomes et/ou de

nucléons va avoir changé. Nous obtenons alors l'état E_{i+1} . Puis nous réitérons cette initialisation jusqu'à l'état E_n solution du problème, c'est-à-dire ne possédant plus de nucléons isolés, un nombre d'atomes proche de nb_{atomes} et étant d'énergie inférieure à une énergie d'initialisation seuil. L'initialisation se rapproche d'un mécanisme de fusion en chaîne, contrôlé par la possibilité de fission lorsque des atomes deviennent disproportionnés par rapport aux autres atomes. L'algorithme 1 présente le pseudo-code de la méthode d'initialisation.

Algorithme 1 – Initialisation

Fonction récursive `init(état, règles) →`
`atome ← choix_atome_init(état, règles);`
Si `choix_init(atome, état)` est la fusion **Alors**
`(nouv_état, nouv_règles, liste_nucléons) ← fusion(atome, état, règles);`
Sinon
`(nouv_état, nouv_règles, liste_nucléons) ← fission(atome, état, règles);`
Tant que `liste_nucléons` non vide **Faire**
`(nucléon, liste_nucléons) ← prendre_nucléon(liste_nucléons);`
`(nouv_état, nouv_règles) ← fusion(nucléon, nouv_état, nouv_règles);`
Si `fin_initialisation(état)` **Alors**
`retourner (nouv_état, nouv_règles);`
Sinon
`init(nouv_état, nouv_règles);`

4.3 Le processus d'optimisation

Le fonctionnement du processus d'optimisation reprend, en le complétant, celui de l'initialisation. L'optimisation part d'un état E_j . Un atome est tiré au hasard. Une fonction de probabilités détermine la probabilité qu'il y ait fusion, fission ou aucun changement. Cette fonction a pour argument la température du plasma, l'énergie de l'état E_j , le nombre de nucléons de l'atome sélectionné, le nombre d'atomes de l'état et le nombre moyen de nucléons par atomes. Si l'atome est « performant » par rapport aux autres atomes, c'est-à-dire d'énergie propre basse par rapport à celle des autres atomes, alors il n'y a ni fusion, ni fission. Dans le cas d'une fusion, un atome va être sélectionné en fonction de son énergie propre, de la température et de sa distance avec l'atome choisi en premier ; nous gardons la même notion de distance que pour l'initialisation. La règle de fusion appropriée est alors sélectionnée. Puis la fusion est réalisée. Elle peut libérer des nucléons qui sont alors fusionnés avec d'autres atomes. Dans le cas d'une fission, la règle appropriée est utilisée. Des nucléons peuvent être éjectés. Mais contrairement à l'initialisation, nous considérons que ces nucléons peuvent avoir atteint une énergie suffisante pour à leur tour créer une fission. Pour les nucléons, le choix entre fission et fusion se fait en considérant la température et les caractéristiques de l'état comme la place de l'atome d'énergie la plus grande par rapport au reste de l'état. Une fission en chaîne peut donc commencer, mais elle se jugule rapidement grâce à la règle choisissant le nombre de nucléons éjectés par une fission. Lorsque toutes les fusions et/ou fissions sont finies, la température est diminuée. Si la température n'est pas trop basse, un calcul de

l'énergie du nouvel état atteint est effectué. Dans le cas où cette énergie est inférieure à la meilleure énergie précédemment trouvée, nous enregistrons ce nouvel état comme l'état le plus performant trouvé. Si cette énergie est supérieure à l'énergie de l'état précédent, la règle est acceptée positivement, sinon elle l'est négativement au sens défini plus haut. Dans tous les cas où la température n'est pas trop basse, l'optimisation est réitérée à partir du dernier état trouvé, l'état E_{j+1} . Mais si la température est trop froide, un nouveau cycle d'optimisation est lancé en partant du meilleur état trouvé, et en élevant la température à son maximum. Le meilleur état trouvé devient le plasma de départ de la nouvelle optimisation. Le pseudo-code est présenté sur l'algorithme 2. L'optimisation telle qu'elle est décrite est sans fin, mais il est facile d'y ajouter une fonction d'arrêt ; par exemple lorsqu'un nombre donné de températures basses aura été atteint, ou lorsqu'une certaine énergie aura été dépassée.

Algorithme 2 – Optimisation

```

température_max ; meilleur_état ;
Fonction récursive optimisation(état, règles, température) →
atome ← choix_atome(état, règles) ;
Si choix_aléatoire(atome, état) est la fusion Alors
  (nouv_état, liste_nucléons) ← fusion(atome, état, règles) ;
  Tant que liste_nucléons non vide Faire
    (nucléon, liste_nucléons) ← prendre_nucléon(liste_nucléons) ;
    nouv_état ← fusion(nucléon, nouv_état, règles) ;
  Si Si choix(atome état) est la fission Alors
    (nouv_état, liste_nucléons) ← fission(atome, état, règles) ;
    Tant que liste_nucléons non vide Faire
      (nucléon, liste_nucléons) ← prendre_nucléon(liste_nucléons) ;
      Si haute_énergie(nucléon) est vrai Alors
        nouv_état ← fission(nucléon, nouv_état, règles) ;
      Sinon
        nouv_état ← fusion(nucléon, nouv_état, règles) ;
  Sinon
    optimisation(état, règles, température) ;
Si énergie(nouv_état) < énergie(état) Alors
  nouv_règles ← accepter_positivement(règles) ;
  Si énergie(nouv_état) < énergie(meilleur_état) Alors
    meilleur_état ← nouv_état ;
Sinon
  nouv_règles ← accepter_négativement(règles) ;
nouv_température ← baisser_température(température) ;
Si température_basse(température) est vrai Alors
  optimisation(meilleur_état, nouv_règles, température_max) ;
Sinon
  optimisation(nouv_état, nouv_règles, nouv_température) ;

```

5 Fonctions de choix et de décroissance de la température

5.1 Fonctions de choix entre fusion et fission

Nous utilisons deux fonctions de choix pour déterminer si un atome a subira une fission ou une fusion. La première fonction concerne le processus d'initialisation, et la seconde, le processus d'optimisation proprement dit.

5.1.1 Fonction de choix pour l'initialisation

La fonction de probabilité pour l'initialisation utilise deux valeurs communes à toute implémentation de la méthode de fusion/fission, le nombre d'atomes de l'état courant et le nombre de nucléons de l'atome a . Si le nombre de nucléons d'un atome est grand proportionnellement au nombre d'atomes, nous voulons qu'il y ait une fission, et que si ce nombre est petit, il y ait une fusion, et que si ce nombre est médian, les probabilités soient égales. Pour cela nous avons construit une fonction $h : [-1; 1]^2 \rightarrow \text{Img}(h) \in R$ qui répond à cette problématique :

$$h(x, y) = \frac{\exp^{\gamma(x+y)} - 1}{\exp^{\gamma(x+y)} + 1}$$

Cette fonction possède des points d'inflexion pour $h(x, -x) = 0, \forall x$. Le paramètre γ régule la pente de la courbe autour de ses points d'inflexion.

Mais pour que cette fonction n'est pas une fonction de probabilité, puisque $\text{Img}(h) \neq [0; 1]$. Nous avons donc construit une fonction $t_z : \text{Img}(h) \rightarrow [0; 1]$, telle que $t_z(z) = \frac{z+k}{2k}$ où $k = h(t_x(\text{nb_tot_nucléons}), t_y(\text{nb_tot_nucléons}))$. Pour les points d'inflexion, les probabilités sont égales puisque $\forall x, h(x, -x) = \frac{1}{2}$. De plus, lorsque le nombre d'atomes est négligeable devant celui de nucléons de a , $t_z(h(x, y)) \rightarrow 1$, et lorsque le nombre de nucléons de a est négligeable devant celui d'atomes, $t_z(h(x, y)) \rightarrow 0$.

Nous connaissons le nombre maximal d'atomes pour l'initialisation, qui est égal au nombre de nucléons nb_tot_nucléons , et le nombre final d'atomes nb_atomes que l'on veut obtenir. Le nombre minimal d'atomes est égal à 1. De même, nous connaissons le nombre maximal de nucléons par atome, et le nombre moyen de nucléons par atome, $n = \frac{\text{nb_tot_nucléons}}{\text{nb_atomes}}$. Le nombre minimal de nucléons dans un atome est 1. Pour pouvoir utiliser la fonction h , nous devons créer deux fonctions adaptant le nombre d'atomes pour t_x , et le nombre de nucléons pour t_y , au domaine de définition de h . Ces fonctions doivent respecter les contraintes suivant : $t_x(\text{nb_atomes}) = 0$ et $t_y(n) = 0$ afin que $h(t_x(\text{nb_atomes}), t_y(n)) = \frac{1}{2}$, $t_x(\text{nb_tot_nucléons}) = t_y(\text{nb_tot_nucléons}) = 1$ et $t_x(1) = t_y(1) = -1$ pour ne pas sortir du domaine de définition de h . La figure 5 en section 8 représente ces fonctions dans leur adaptation à notre problème de découpage de l'espace aérien.

Soit $t(x, v) = \frac{\log((m-2v+1)x+v^2-m) - \log((v-1)(m-v))}{\log(m-v) - \log(v-1)}$ où $m = \text{nb_tot_nucléons}$.

Nous proposons : $t_x(x) = -t(x, \text{nb_de_zones})$ et $t_y(y) = t(x, n)$.

Pour finir, nous avons défini la fonction de probabilité choix_2 représentée en section 8, figure 6 dans son adaptation à notre problème

$$\text{choix}_2(x, y) = t_z(h(t_x(x), t_y(y)))$$

5.1.2 Fonction de choix pour l'optimisation

Pour le processus d'optimisation, nous avons voulu une fonction de choix très simple et qui soit facilement généralisable. Nous avons donc abandonné les courbes avec points d'inflexion, et nous nous sommes concentré sur un seul argument, le nombre de nucléons de l'atome. Nous avons laissé de côté le nombre d'atomes. Pourtant durant l'optimisation, le nombre d'atome devra osciller autour de nb_atomes . Mais ce n'est pas à la fonction de choix d'influer directement sur le nombre d'atomes, mais à la fonction de coût, comme nous l'avons vu en section 3.

Nous avons défini la fonction de probabilité $choix_n$:

$$choix_n(x) = \begin{cases} 1 & \text{si } x > n + \frac{1}{2\alpha} \\ 0 & \text{si } x < n - \frac{1}{2\alpha} \\ \alpha(x - \frac{nb_tot_nucléons}{nb_atomes}) + \frac{1}{2} & \text{sinon} \end{cases}$$

Et α peut être fonction de la température, $\alpha(t) = k \frac{t_{max}-t}{t_{max}-t_{min}} + r$. Avec t_{max} et t_{min} les températures maximales et minimales du plasma.

Cette fonction de choix est une simple droite de pente α entre $n - \frac{1}{2\alpha}$ et $n + \frac{1}{2\alpha}$, et telle que $choix_n(\frac{nb_tot_nucléons}{nb_atomes}) = \frac{1}{2}$. Ainsi, la probabilité d'avoir une fusion ou une fission est égale si le nombre de nucléons de l'atome a est égal au nombre moyen de nucléons. Quand le nombre de nucléons de a est trop grand, $x > n + \frac{1}{2\alpha}$, il y a fission, et quand il est trop petit, $x < n - \frac{1}{2\alpha}$, il y a fusion.

S'il est intéressant de se concentrer sur le nombre de nucléons d'un atome, et de le faire varier, il peut être encore plus intéressant de mettre l'énergie d'un atome en paramètre de la fonction de choix. En effet, les nucléons peuvent être très différents les uns de autres et deux atomes constitué du même nombre de nucléons peuvent avoir une énergie propre bien différente. Nous verrons dans le cadre de notre adaptation au problème du découpage aérien en section 8, une fonction de probabilité qui utilise l'énergie.

5.2 Fonction de décroissance de la température

Nous avons défini trois fonctions de décroissance de la température. Chacune prend la forme d'une suite strictement décroissante. Une seule de ces suite sera utilisée lors de l'optimisation. Une représentation graphique de ces trois suites est donnée par la figure 3.

$$\begin{aligned} u_1(n) &= \frac{u_1(0)}{n+1} \\ u_2(n) &= \frac{1 - \exp^{p(n-n_{max})}}{1 + \exp^{p(n-n_{max})}} u_2(n-1) \\ u_3(n) &= \sqrt{\frac{u_3(n-1) - t_{min}}{t_{max} - t_{min} + 1}} u_3(n-1) \end{aligned}$$

Pour chaque suite u_i , la valeur $u_i(0)$ est égale à la température maximale t_{max} . La suite u_1 a une décroissance très rapide et limitera rapidement les fission en chaîne. La suite u_2 possède un point d'inflexion. Dans cette suite, n_{max} représente le nombre maximum d'itérations de la température que nous cherchons, et p sert à faire varier la pente de la courbe autour de son point d'inflexion. La suite u_3 , a une courbe de décroissance d'abord lente, puis très rapide.

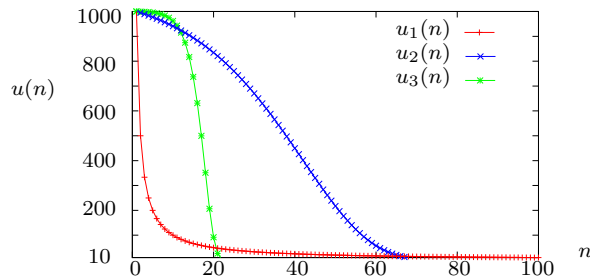


FIG. 3 – Représentations graphiques des suites u_1 , u_2 et u_3

Nous verrons en section 8 que la suite u_3 correspond à la fonction de décroissance la plus performante que nous proposons.

6 Le problème du découpage de l'espace aérien

L'espace aérien est décomposé en secteurs géographiques élémentaires nommés secteurs de contrôle. Ces secteurs sont regroupés en zones de qualification. Les contrôleurs aériens sont qualifiés pour une zone spécifique. Les contrôleurs, lorsqu'ils contrôlent un ou plusieurs secteurs, ont une charge de travail qui dépend de trois facteurs, la surveillance du trafic, la résolution des conflits entre avions ou conflits potentiels, ceux qui sont susceptibles de se produire, et la coordination entre secteurs [All96] [Del95]. Dans ce dernier cas, un stress plus particulier s'impose au contrôleur lorsqu'il doit gérer des échanges de trafic important entre un secteur de sa zone de qualification et celui d'une autre zone. Ceci est dû principalement au fait qu'il ne connaît pas aussi bien les secteurs hors de sa zone que ceux qui y appartiennent. Parallèlement, une bonne gestion des coordinations entre zones peut diminuer le risque de conflits et de conflits potentiels que le contrôleur doit gérer.

Dans cette étude nous travaillerons sur les données de trafic européen de la journée du 8 février 2002. Nous utiliserons la modélisation des flux⁵ entre secteurs introduite dans le rapport [Bic04] pour calculer les masses aux frontières et les masses internes des zones de qualification. La masse interne est la somme des flux échangés entre secteurs d'une même zone, alors que la masse aux frontières est la somme des flux entre secteurs d'une zone et secteurs connexes⁶ hors de cette zone. Il est important de souligner qu'une zone de qualification comporte toujours plus de deux secteurs, et que la somme des flux entre ces secteurs est non nulle. De plus, il existe toujours un chemin pour aller d'un secteur vers un autre secteur.

À titre illustratif, voici quelques valeurs pour nous permettre de mieux cerner le problème. Plus de 20 000 avions sont contrôlés quotidiennement par les 11 pays de la « country core area »⁷ tel que défini dans [BA05b]. Dans la journée du 8 février 2002, il y a plus de 150 000 mouvements d'avions passant d'un

⁵ Nous appelons flux le nombre d'avions passant d'un secteur à un autre, ou encore du sol à un secteur et inversement, par unité de temps

⁶ Deux secteurs sont connexes si et seulement si ils ont une frontière commune

⁷ L'Allemagne, le Royaume-Uni, la France, l'Espagne, l'Italie, la Suisse, le Luxembourg, le Belgique, les Pays-Bas, le Danemark et l'Autriche

secteur à un autre. Pour $nb_de_zones = 36$ zones, la masse interne d'une zone est en moyenne de 2 375 passages, alors que la masse aux frontières est de 3 365 passages. Il existe toujours un chemin entre 2 secteurs quelconques. De plus, une zone de qualification est au moins constituée de deux secteurs reliés entre eux par un flux non nul. Ainsi, aucune des deux masses ne peut être nulle. Ces deux masses varient entre 1 et 25 000 passages.

Nous allons appliquer des méthodes d'optimisation par métaheuristiques au découpage de l'espace aérien en nouvelles zones de qualification. Mais pour que les résultats soient exploitables, il faut que les critères à optimiser soient judicieusement choisis [AGP02]. Pour cela nous avons défini trois contraintes que doit satisfaire le ou les critères que nous cherchons :

1. Toutes les zones doivent avoir le moins de flux aux frontières et les plus petits flux aux frontières possibles.
2. La concentration des flux importants doit se situer à l'intérieur des zones.
3. Les différentes zones doivent être homogènes entre elles.

Ces contraintes nous ont amenés à définir trois critères. Le premier de ces critères cherche à satisfaire les deux premières contraintes. Nous l'appellerons critère total puisque c'est une somme sur le total des zones de qualification, à la différence du critère qui va suivre.

$$C_{total} = \sum_{\{zones\}} \frac{masse_aux_frontieres(zone)}{masse_interne(zone)}$$

Ce critère est toujours défini puisque la masse interne pour chaque zone est non nulle d'après ce que l'on a vu plus haut.

Nous avons aussi défini le critère partiel qui satisfait la troisième contrainte et qui lui n'est au final que le rapport d'au plus deux zones différentes, d'où son nom de partiel.

$$C_{partiel} = \frac{\min_{\{zones\}} masse_aux_frontieres(zone)}{\max_{\{zones\}} masse_interne(zone)}$$

De même que pour le critère total, ce critère est toujours défini puisque comme nous l'avons vu plus haut, la masse interne pour chaque zone est non nulle. Nous avons vérifié expérimentalement que ce critère tend à homogénéiser la taille des zones entre elles, tout en contribuant aux eux premières contraintes énoncées précédemment.

Enfin, nous avons introduit le critère somme afin de concilier toutes nos contraintes en un seul et même critère.

$$C_{somme} = C_{partiel} + C_{total}$$

Le critère somme nous donne l'énergie du découpage de l'espace aérien. Nous chercherons dans nos différentes optimisations à minimiser cette fonction énergie.

7 Un problème NP-complet

Nous allons montrer dans cette section que le problème II de découpage du ciel aérien européen utilisant le critère somme défini section 6 est NP-complet.

Pour montrer que Π est NP-complet, d'après [GJ79], il nous suffit de suivre les étapes suivantes :

1. Montrer que Π est dans NP
2. Trouver un problème Π' NP-complet proche de Π
3. Réduire Π à Π' : trouver un sous-problème de Π équivalent à Π'

Dans ce problème, nous pouvons considérer chaque secteur comme un nœud, chaque flux entre secteurs comme un arc. Nous cherchons alors à diviser le graphe $G = (V, E)$ formé par ces nœuds et ces arcs en n sous-ensembles tels que la somme des poids des arcs entre chaque sous-ensemble soit minimale et que la somme des poids des arcs internes à chaque sous-ensemble soit maximale. Le graphe G est connexe, fini et connu. Comme nous l'avons vu à la section 6, pour qu'un état soit solution du problème, chaque sous-ensemble doit comporter au moins deux sommets dont le poids de l'arc les reliant est non nul : il doit au moins posséder deux secteurs reliés par un flux non nul.

7.1 Formalisation du problème

Selon la théorie des graphes, nous pouvons formaliser le problème Π de la manière suivante :

Instance : Soit un graphe pondéré $G = (V, E)$, une fonction de poids $w : E \rightarrow \mathbb{Z}^+$ et un entier positif K , un entier positif $B \leq |V|$, et n sommets dans V .

Question : Existe-t'il une partition P de V en n sous-ensembles $V_1 \cdots V_n$ disjoints, non vides, et vérifiant pour tout i , $B \geq |V_i|$, et tel que la somme pour chaque V_i , de la somme des poids des arcs reliant V_i aux autres sous-ensembles divisé par la somme des poids des arcs possédant leurs deux sommets dans V_i , plus le minimum sur les V_i de la somme des poids des arcs reliant V_i aux autres sous-ensembles divisé par le maximum sur les V_i de la somme des poids des arcs possédant leurs deux sommets dans V_i , est inférieure ou égale à K ?

Soit $poids(i, k) = \sum_{e \in (V_i, V_k)} w(e)$. Nous cherchons une partition P telle que :

$$\sum_{V_k \in P} \frac{\sum_{V_i \neq V_k \in P} poids(i, k)}{poids(k, k)} + \frac{\min_{V_k \in P} \sum_{V_i \neq V_k \in P} poids(i, k)}{\max_{V_k \in P} poids(k, k)} \leq K \quad (1)$$

Comme chaque sous-ensemble doit comporter au moins deux sommets dont le poids de l'arc entre eux est non nul, nous avons $\forall i, poids(i, i) \geq 1$.

7.2 Π dans NP

Pour montrer que Π est dans NP, il suffit de montrer que la vérification d'une réponse au problème Π s'effectue en un temps polynomial.

Soit une réponse au problème Π . Nous avons n ensembles V_i , qui sont autant de zones de qualification. Pour vérifier la validité de cette réponse, il suffit de calculer la somme des poids des arcs, entre chaque ensemble de sommets V_i ; ainsi que pour chaque V_i , calculer la somme des poids des arcs dont les deux sommets sont dans V_i . Comme le graphe G est connu, il est évident que le calcul de l'équation (1) se fait en temps polynomial.

7.3 Le problème du « MIN CUT »

Il nous faut chercher un problème NP-complet connu se rapprochant de Π . Nous proposons le problème du « minimum cut into bounded sets », NP-complet [GJS76], qui s'énonce ainsi : Soit un graphe pondéré $G = (V, E)$, une fonction de poids $w : E \rightarrow Z^+$, deux sommets $s, t \in V$, un entier positif $B \leq |V|$ et un entier positif K' . Existe-t'il une partition de V en deux ensembles disjoints V_1 et V_2 vérifiant $s \in V_1$, $t \in V_2$, $|V_1| \leq B$, $|V_2| \leq B$, et telle que : $poids(1, 2) \leq K'$

Nous observons que le premier terme de cette inégalité se retrouve dans l'équation (1).

7.4 Restriction du problème

Il existe plusieurs méthodes pour prouver la NP-complétude [GJ79]. Nous allons utiliser la méthode la plus commune, qui consiste à restreindre le problème de départ pour montrer que ce dernier contient pour un cas particulier un problème connu comme étant NP-complet.

Posons $n = 2$ et supposons que : $poids(2, 2) \leq poids(1, 1)$. L'équation (1) devient :

$$2 \frac{poids(1, 2)}{poids(1, 1)} + \frac{poids(1, 2)}{poids(2, 2)} \leq K \quad (2)$$

7.5 Un sous problème de Π équivalent à Π'

Or comme $\forall i, poids(i, i) \geq 1$, $\frac{1}{poids(i, i)} \leq 1$, si l'on utilise l'inégalité précédente pour majorer le premier terme de l'inégalité (2), nous obtenons :

$$2 \frac{poids(1, 2)}{poids(1, 1)} + \frac{poids(1, 2)}{poids(2, 2)} \leq 3 poids(1, 2) \quad (3)$$

Nous retrouvons en second terme de l'équation (3), le problème du « MIN CUT ». Nous pouvons donc majorer le second terme de l'équation (3) par $3K'$. Nous avons donc trouvé le terme K de l'équation (1), il vaut $3K'$. Ainsi le problème Π est réductible au problème Π' . Π est donc NP-complet.

8 Adaptation à notre problème

8.1 Préambule

Nous pouvons facilement assimiler les secteurs aux nucléons et les zones de qualification aux atomes. Ainsi, tout état est associé de manière unique à un découpage aérien du ciel et réciproquement. Concernant le mécanisme des règles, nous avons fixés à trois le nombre maximal de secteurs pouvant être éjectés simultanément lors d'une fusion ou d'une fission. Les zones de qualification, qui fusionnent ou fissionnent, libèrent au plus trois secteurs. Nous voulons diviser l'espace aérien européen en un nombre précis de zones de qualification, $nb_de_zones = nb_atomes$ que nous avons défini plus haut. De même, le nombre de nucléons $nb_tot_nucléons = nb_tot_secteurs$, le nombre total de secteurs de l'espace aérien européen.

Nous allons voir le mécanisme de fusion, puis celui de fission que nous avons choisis. Dans la fusion, à partir de la zone de qualification choisie au début de l’algorithme, une autre zone est sélectionnée. Cette sélection est déterministe, et fonction de l’importance des flux entre la nouvelle zone et la première. Puis les deux zones sont fusionnées en une troisième, et disparaissent. Grâce au mécanisme de règle, un nombre aléatoire de secteurs sont retirés de la nouvelle zone résultant de la fusion. Le mécanisme est exactement le même si l’on part d’un secteur, pour le fusionner avec une zone de qualification.

Concernant la fission, la première étape est d’appliquer le mécanisme de règle. Un nombre aléatoire de secteurs sont ainsi détachés de la zone initiale. Nous voulons après cette étape répartir les secteurs restants en deux zones. Nous disposons de méthodes performantes pour répartir les secteurs en zones de qualification. Ces méthodes sont présentées en section 9 pour la percolation et 10 pour les colonies de fourmis. Le découpage de cette ensemble de secteurs en deux zones va ainsi se faire par la méthode d’optimisation par colonies de fourmis, précédé d’une étape d’initialisation par percolation. Les deux secteurs de départ de la percolation sont choisis aléatoirement parmi l’ensemble des secteurs. Puis vient l’optimisation par colonies de fourmis. Le nombre d’itérations de l’optimisation par colonies de fourmis est un nouveau paramètre. À la fin de ce nombre d’itérations, l’ensemble de secteurs est séparé en deux zones et la fission est réalisée.

Le mécanisme de règle, nous écarte du problème de connexité forcée entre les secteurs. En effet, deux zones et/ou secteurs ne peuvent fusionner que si un flux les lie. La fusion fonctionne donc sur le mécanisme de connexité entre les secteurs. Mais le fait d’exclure au moins un secteur choisi aléatoirement, rompt ce mécanisme. En effet, l’ensemble résultant peut ne plus du tout être connexe comme précédemment. Ceci montre bien l’utilité d’interdire qu’une probabilité devienne nulle dans le mécanisme de règle (nous avons vu en section 3 que chaque probabilité doit être strictement comprise entre 0 et 1).

8.2 Le processus d’initialisation

Comme nous l’avons vu en section 4 notre méthode se divise en deux étapes. Nous allons donc commencer par l’étape d’initialisation. Pour cette étape, il nous faut définir trois fonctions utilisées dans l’algorithme 1. La première, la fonction de choix de la zone de qualification initiale, *choix_atome_init*, choisit aléatoirement une zone ou un secteur parmi ceux de l’état. La fin de l’initialisation se produit si l’énergie de l’état obtenu est inférieure à une valeur donnée en paramètre de l’algorithme. Enfin la fonction *choix_init* nous donne la probabilité de choisir une fusion ou une fission. Nous avons créé deux fonctions de probabilité, *choix₁(x, y)* et *choix₂(x, y)* qui peuvent servir pour cette fonction.

Nous avons choisi la première fonction de probabilité pour répondre à la contrainte suivante : si le nombre de secteurs de la zone choisie est supérieur au nombre moyen de secteurs des zones de cet état, la probabilité de fission soit plus importante. Et réciproquement, si ce nombre est inférieur à la moyenne, la probabilité de fusion doit être plus importante. Cette première fonction de probabilité est spécifique à notre problème de découpage de l’espace aérien, mais peut servir d’inspiration pour d’autres problèmes. Soit *nb_zones* est le nombre de zones de qualification de l’état *E* sur lequel nous allons appliquer notre fonction de probabilité. Cette fonction *choix₁(x, y)* prend en arguments

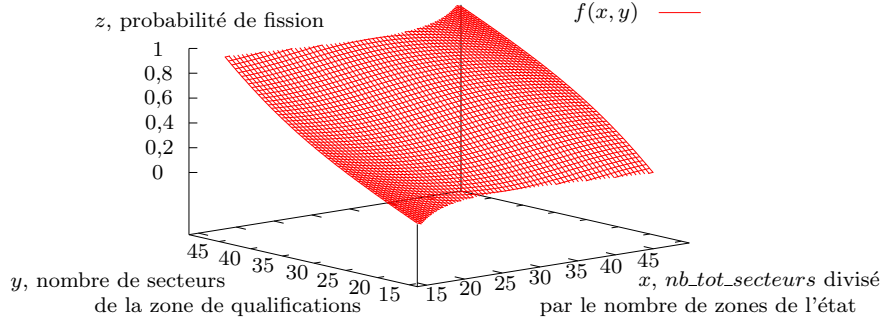


FIG. 4 – Courbe de la fonction f dans l'intervall $[0;1]$

$x = \frac{nb_tot_secteurs}{nb_zones}$ le nombre moyen de secteurs dans une zone pour notre état E , et y le nombre de secteurs présents dans une zone choisie par $choix_atome_init$, et s'écrit :

$$f(x, y) = \frac{\exp^{\alpha(x+y-2n)}}{1 + \exp^{\alpha(x+y-2n)}} + \frac{\exp^{\beta(y-n)}}{\exp^{\beta(x-n)} - \exp^{\beta(y-n)}}$$

$$choix_1(x, y) = \begin{cases} 1 & \text{si } f(x, y) > 1 \\ 0 & \text{si } f(x, y) < 0 \\ f(x, y) & \text{sinon} \end{cases}$$

Pour que notre méthode converge bien vers le nombre de zones nb_de_zones , c'est-à-dire que $choix_1(n, n) = \frac{1}{2}$, nous avons introduit $n = \frac{nb_tot_secteurs}{nb_de_zones}$. La courbe possède donc un point d'inflexion en $x = y = n$, pour lequel la fonction vaut $\frac{1}{2}$. Le coefficient α permet de faire varier la pente de la courbe autour de son point d'inflexion et le coefficient β permet de faire varier la pente verticale de la courbe sur toute sa largeur. Si la valeur retournée par $choix_1(x, y)$ est supérieure à $\frac{1}{2}$, alors il y aura plus de chance d'avoir une fission, et inversement. Soient $p_1 = f(\frac{1}{2}n, \frac{1}{2}n)$ et $p_2 = f(\frac{1}{2}n, \frac{3}{2}n)$. Alors : $\alpha = \frac{1}{n} \log(\frac{1}{p_1} - 1)$ et $\beta = \frac{1}{n} \log\left(\sqrt{\left(\frac{p_2}{2} - \frac{1}{4}\right)^2 + 1} + \frac{p_2}{2} - \frac{1}{4}\right)$.

La figure 4 représente $f(x, y)$ où $nb_de_zones = 24$, $nb_tot_secteurs = 762$, et donc $n = 31,75$. Si nous prenons $p_1 = 0,02$ et $p_2 = 1,2$ alors nous trouvons $\alpha \simeq 0,12$ et $\beta \simeq 0,02$. Nous obtenons des résultats assez performants en rapidité de convergence avec ces valeurs.

La seconde fonction de probabilité est celle présenté en section 5 pour le processus d'initialisation. Il s'agit de la fonction de probabilité $choix_2$ représentée figure 6 pour $\gamma = 3$. Nous rappelons que $choix_2(x, y) = t_z(h(t_x(x), t_y(y)))$. Les courbes des fonctions t_x et t_y sont représentée figure 5. Comme nous l'avons vu, $nb_tot_nucléons = nb_tot_secteurs$ et $nb_atomes = nb_de_zones$.

8.3 Le processus d'optimisation

Maintenant que l'étape d'initialisation a été décrite, nous pouvons passer à celle d'optimisation proprement dite. Nous démarrons l'optimisation avec les résultats obtenus après l'initialisation et la règle retournée. Le choix de la zone de qualification initiale est le même que pour l'initialisation, c'est-à-dire aléatoire.

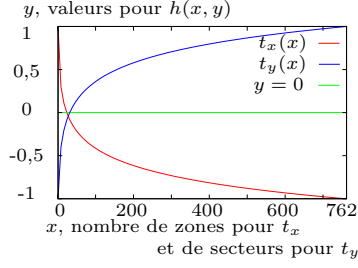


FIG. 5 – Courbes des fonctions $t_x(x)$ et $t_y(x)$

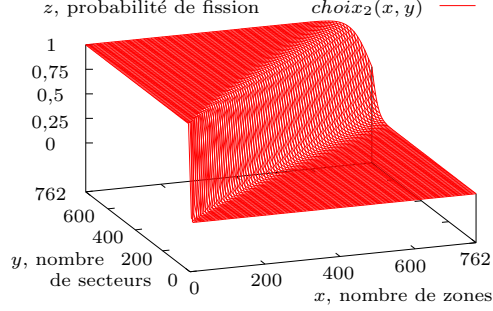


FIG. 6 – Courbe de $choix_2(x, y)$

La méthode d'optimisation par fusion/fission doit faire varier le nombre de zones de qualification des états solutions du problème autour de nb_de_zones . Nous utilisons comme fonction de coût le critère somme défini en section 6. Mais comme nous l'avons vu à la section 3, notre fonction de coût doit tenir compte de l'écart entre le nombre de zone de qualification différentes de l'état et le nombre de zones que l'on veut obtenir, nb_de_zones . Nous avons donc créé la fonction d'énergie proportionnelle qui permet expérimentalement de toujours rester proche de nb_de_zones zones de qualification :

$$E_{prop}(x) = \left(\left(\frac{nb_de_zones - x}{3} \right)^2 + 1 \right) C_{somme}$$

E_{prop} est fonction du nombre de zones de qualification de l'état. Plus nous nous écartons de nb_de_zones , plus le facteur multiplié au critère somme est grand.

La fonction, $temperature_basse$, indiquant si la température est basse effectue simplement une comparaison avec un paramètre de température minimale.

La fonction de choix, $choix_aleatoire$, qui détermine si une fusion ou une fission sera appliquée à une zone de qualification ressemble à la fonction $choix_init$ utilisée lors de l'initialisation. Nous pouvons utiliser les fonctions $choix_1$ et $choix_2$ pour cette étape d'optimisation, à condition de faire varier respectivement les paramètres α , β et γ en fonction de la température. Pour cela, il nous a suffi de créer une fonction d dont les valeurs sont comprises entre 0 et 1, telle que : $d(t) = \frac{t-t_{min}}{t_{max}-t_{min}}$. Chacun de ces paramètres peut donc être réutilisé sous la forme $p \times d(t) + r$. Ainsi, nous utilisons pour notre problème $\alpha(t) = 0.8 \times d(t) + 0.5$, $\beta(t) = \gamma(t) = 3 \times d(t) + 2$. Mais nous avons créé une autre fonction de probabilité qui prend aussi en compte l'énergie d'une zone grâce au critère total défini section 6, et l'énergie de l'état. Mais le critère total C_{total} peut retourner une valeur infinie si une des zones n'est constituée que d'un secteur. En effet dans ce cas la masse interne de la zone devient nulle, ce qui implique que C_{total} tende vers l'infini. Cette nouvelle fonction $choix_e$ vaut $choix_2$ lorsque le critère total C_{total} prend une valeur infinie, et $e(x, y)$ sinon :

$$e(x, y) = \frac{\exp^{\alpha k_x + \beta k_y} - \exp^{\alpha(x - nb_de_zones) + \beta(y - \frac{e_{tot}}{x})}}{(\exp^{\alpha k_x + \beta k_y} - 1)(1 + \exp^{\alpha(x - nb_de_zones) + \beta(y - \frac{e_{tot}}{x})})}$$

$$choix_e(x, y, z) = \begin{cases} choix_2(x, z) & \text{si } C_{total} \rightarrow \infty \\ e(x, y) & \text{sinon} \end{cases}$$

Avec x le nombre de zones de qualification de l'état, y l'énergie de cette zone et z le nombre de secteurs de la zone. L'énergie d'une zone de qualification vaut $y = \frac{\text{masse_aux_frontieres}(zone)}{\text{masse_interne}(zone)}$ avec les notations utilisées section 6. C'est l'énergie d'une zone de qualification dont nous nous servons pour calculer C_{total} . Les coefficients k_x et k_y sont calculés ainsi :

- Les probabilités d'avoir une fission ou une fusion sont égales lorsque le nombre de zones de l'état vaut nb_de_zones et que le nombre de secteurs de la zone de qualification vaut $\frac{e_{etat}}{nb_de_zones}$, c'est-à-dire : $e(nb_de_zones, \frac{e_{etat}}{nb_de_zones}) = \frac{1}{2}$
- Il y aura toujours fusion pour $nb_de_zones - k_x$ et $\frac{e_{etat}}{nb_de_zones} - k_y$, c'est-à-dire : $e(nb_de_zones - k_x, \frac{e_{etat}}{nb_de_zones} - k_y) = 0$
- Il y aura toujours fission pour $nb_de_zones + k_x$ et $\frac{e_{etat}}{nb_de_zones} + k_y$, c'est-à-dire : $e(nb_de_zones + k_x, \frac{e_{etat}}{nb_de_zones} + k_y) = 1$

La fonction de probabilité $choix_e$, en plus d'être complexe, est difficile à réutiliser dans le cadre d'autres problèmes. Nous pouvons utiliser la fonction $choix_n$ présentée en section 5. Nous rappelons que cette fonction agit sur le nombre de secteurs d'une zone de qualification et non plus sur son énergie.

Les fonctions de décroissance de la température sont décrites en section 5. Nous avons remarqué d'une manière générale qu'il n'était pas profitable de diminuer trop rapidement la température, et que la diminuer lentement aux températures basses ne servait à rien. Ainsi, nous avons constaté de manière empirique que le schéma de décroissance de la température ne suivait absolument pas celui du recuit simulé tel que nous le conseille Lester Ingber dans [Ing89]. En effet, pour le *Fast Annealing*, la décroissance n'est autre que u_1 , et pour le *Very Fast Annealing*, la décroissance est en exponentielle inverse, dont la décroissance est lente pour les petites valeurs de température. Ainsi, avec la suite u_3 donne des résultats plus satisfaisants en terme de rapidité et de solutions que les autres suites.

Nous avons décrit les mécanismes qui nous ont permis d'adapter la méthode d'optimisation par fusion/fission à notre problème. Il nous reste à présenter le mécanisme de percolation, ainsi que l'optimisation par colonies de fourmis et recuit simulé.

9 Percolation

Au sens littéral, la percolation est l'écoulement d'un fluide à travers un solide poreux, tel qu'un filtre. L'étude de la percolation a été faite par MM. Broadbent et Hammersley qui ont introduit le terme de percolation dans [BH57]. La percolation est un processus de propagation aléatoire d'un fluide à travers un milieu. Le modèle de percolation se définit par le mouvement déterministe d'un fluide à travers un milieu dont la structure est aléatoire. L'aléa ne se trouve pas dans le mouvement du fluide mais dans le milieu à travers lequel il évolue [Paj01] [Cou03].

Nous avons cherché à adapter le principe de percolation à notre problème. La percolation fonctionne par écoulement d'un fluide de proche en proche, le fluide part d'un endroit initial pour se répandre aux alentours. Il nous faut donc définir autant de secteurs d'écoulement initiaux, que de zones de qualification. Au départ, ces secteurs initiaux constituent les zones. Puis un secteur connexe à une zone sera rattaché à cette zone en fonction d'un critère de choix déter-

ministe reposant sur des informations variables entre deux secteurs connexes. Ces informations entre deux secteurs reposent sur la connaissance du flux entre ces secteurs, mais aussi sur les flux entre un secteur initial et les secteurs relais jusqu'à ces deux secteurs. Ces flux étant pondérés par la distance avec le site de départ. Cette notion de distance a un caractère aléatoire, puisque la distance entre deux secteurs est fonction du nombre de secteurs traversés pour passer de l'un à l'autre, or le chemin suivi pour déterminer cette distance est celui de la percolation, ce n'est donc pas forcément le plus court et il est changeant.

Le choix d'un secteur à intégrer dans une zone déjà formée se fait en considérant pour un secteur donné tous ses secteurs connexes, puis en créant les liens qui unissent tous ces secteurs entre eux et en choisissant le lien le plus fort parmi les secteurs appartenant à des zones existantes. Le secteur sera alors intégré à cette zone. Nous calculons le lien d'un secteur S à une zone CEN de secteur initial C par la formule suivante :

$$lien(S, CEN) = \sum_{\text{Chemin de } C \text{ vers } S} \frac{flux(A \rightarrow B)}{2^d}$$

On divise par 2^d pour tenir compte de l'éloignement au secteur origine de la zone.

Cette méthode procède par incorporation successive des secteurs. À chaque itération, tous les secteurs d'une zone se voient appliquer cette méthode, ainsi un secteur appartenant déjà à une zone peut malgré tout changer de zone. Nous arrêtons la récursion lorsque tous les secteurs sont répartis dans les différentes zones et qu'aucun déplacement de secteur vers une autre zone n'est envisageable.

Ce mécanisme est très rapide pour créer une solution initiale connexe, c'est-à-dire une solution dont chaque graphe représentant une zone de qualification est connexe. Et comme nous l'avons déjà constaté [Bic04], les solutions connexes sont des solutions de basse énergie. La percolation est donc pour notre problème une méthode qui possède un bon rapport rapidité sur performance de la solution. C'est pourquoi nous l'utilisons pour initialiser nos méthodes d'optimisation par métaheuristiques [BA05a], comme les colonies de fourmis que nous allons voir maintenant.

10 Colonies de fourmis

10.1 Introduction aux colonies de fourmis

L'optimisation par colonies de fourmis, [DMC96] [DCG99] s'inspire du comportement des fourmis lorsque celles-ci sont à la recherche de nourriture. Les fourmis en se déplaçant déposent des phéromones, substances olfactives et volatiles. Chaque fourmi se dirige en tenant compte des phéromones qui sont déposées par les autres membres de la colonie. Les fourmis choisissent leur chemin de manière probabiliste, en tenant compte du nombre de phéromones déposées précédemment par les autres fourmis au même endroit. Plus le nombre de phéromones indiquant un chemin sera grand, plus la fourmi aura de chances de suivre ce chemin. Comme les phéromones s'évaporent progressivement, le choix probabiliste que prend une fourmi pour choisir son chemin évolue continuellement. Ce mécanisme de choix des chemins peut être illustré par l'exemple figure 7.

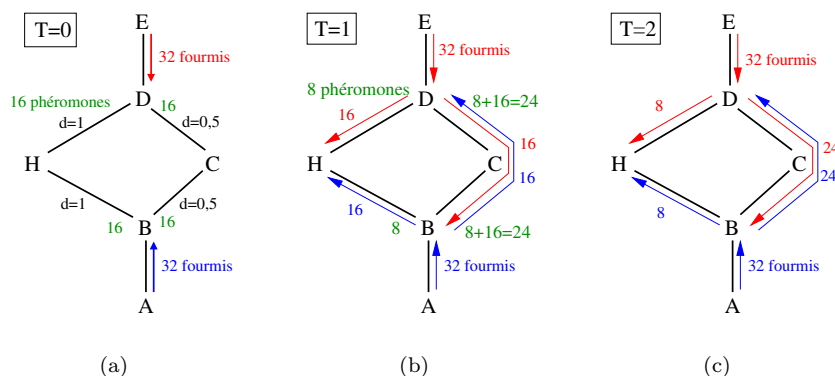


FIG. 7 – Un exemple de la stratégie des colonies de fourmis, le contournement d’obstacles

- (a) Le schéma initial avec les distances, 32 fourmis arrivent en D et en B. Pas de phéromones, les fourmis choisissent une direction de manière aléatoire.
- (b) À $t=1$, évaporation des phéromones, arrivée en B des fourmis venant de C.
- (c) Les fourmis sont allées majoritairement dans la plus courte branche.

À $T=0$, 32 fourmis sont en D et 32 autres en B. Le choix du chemin à suivre entre H et C est complètement aléatoire puisqu’il n’y a encore aucune phéromone sur ce chemin. Chaque groupe de fourmis se divise donc en deux parties égales qui déposent chacune 16 phéromones sur leur chemin. À $T=1$, 32 nouvelles fourmis se présentent en D et 32 autres en B. Mais pendant ce temps, les 16 fourmis parties de D en passant par C sont arrivées en B, de même les 16 autres parties de B en passant par C sont en D. Il y a donc $16 + \frac{16}{2} = 24$ phéromones qui indiquent le chemin en passant par C. Ces $\frac{16}{2} = 8$ phéromones sont dues à l’évaporation. De l’autre côté, les 16 fourmis parties de D et les 16 autres parties de B se retrouvent en H. Il n’y a donc que $\frac{16}{2} = 8$ phéromones qui indiquent en D comme en C le chemin en passant par H. Les nouvelles fourmis vont donc majoritairement choisir le chemin par C. C’est ce que l’on voit à $T=2$.

Dans l’exemple précédent, les phéromones s’évaporent de moitié à chaque nouveau pas d’horloge. Mais le schéma de décroissance peut être différent, c’est l’un des paramètres de modulation des colonies de fourmis avec le nombre de fourmis.

10.2 Adaptation des colonies de fourmis au découpage de l’espace aérien

Nous avons adapté la méthode d’optimisation par colonies de fourmis à notre problème de découpage de l’espace aérien en zones de qualification [BA05a]. En premier lieu, nous avons considéré que chaque zone de qualification était le territoire d’une colonie de fourmis. Les différentes colonies sont mises en concurrence sur l’appartenance d’un territoire et de la nourriture qui est dessus ; la nourriture représentant le trafic entre secteurs. Pour garder leur territoire, les fourmis doivent créer des liens forts avec lui sous forme de phéromones. Ainsi un secteur appartient à la colonie qui possède le plus de phéromones le désignant. Les

fourmis se déplacent les unes après les autres, zones de qualification après zones de qualification.

Les fourmis pour se déplacer suivent les phéromones qui marquent un chemin entre deux secteurs avec une certaine probabilité comme expliqué en figure 7, à laquelle vient s'ajouter une probabilité fixe notée P_f pour n'exclure a priori aucun secteur. L'algorithme déplace les colonies les unes après les autres, puis pour chaque colonie, il déplace les fourmis séparément. Une fois le déplacement d'une fourmi effectué, on regarde l'énergie du nouvel état. Si l'énergie de l'état résultant du déplacement d'une fourmi diminue, cet état est accepté, sinon il est accepté avec une probabilité de Boltzmann dépendant de cette énergie et de la température. Dans le cas où l'état n'est pas accepté, la fourmi reste immobile. Ce mécanisme implique de réévaluer chaque état après le passage d'une fourmi, et non pas de la colonie tout entière, ou de toutes les colonies. Il est donc inutile d'enregistrer la route de chaque fourmi afin d'y déposer par la suite des phéromones. Celles-ci sont posées après le passage de chaque fourmi. Voici la fonction de probabilité de Boltzmann utilisée :

$$h_k(x_{k+1}|x_k) = \exp^{-\frac{E(x_{k+1})-E(x_k)}{T}}$$

Où x_k représente l'état précédent et x_{k+1} l'état suivant, $E(x_k)$ l'énergie de l'état x_k , et T une constante qui permet d'ajuster les probabilités.

Une fois tous les déplacements effectués, une certaine quantité de phéromones est évaporée en fonction du temps et d'un paramètre t_{max} . Le paramètre t_{max} fixe la durée de vie d'une phéromone. Lors du déplacement d'une fourmi, on ne tient pas compte des phéromones placés au même « tour ». La quantité de phéromones signalant un chemin entre deux secteurs se calcule donc ainsi :

$$f(A \rightarrow B) = \sum_{t=1}^{t_{max}} \frac{Pheromones_deposees_il_y_a(t)}{t}$$

10.3 L'algorithme, son comportement et son utilisation

Le pseudo-code de notre adaptation des colonies de fourmis au problème du découpage aérien est donné par l'algorithme 3.

De par la nature de l'optimisation par colonies de fourmis, la construction d'une zone de qualification se fait par « connexité » des secteurs. En effet, une fourmi ne se déplace d'un secteur que vers un de ses voisins. Pourtant la connexité n'est pas forcée, en effet, rien n'interdit qu'une colonie soit divisée en deux parties par d'autres colonies. Les résultats obtenus montrent cependant que les zones de qualification sont toujours connexes lorsque l'on utilise notre implémentation des colonies de fourmis. C'est la preuve que l'algorithme fonctionne de manière assez satisfaisante puisque l'on arrive à des zones de plus faible énergie et donc à un état de faible énergie.

Les variables qui sont utiles à notre implémentation de l'optimisation par colonies de fourmis sont les suivantes :

- Le nombre de fourmis par colonies
- t_{max} qui fixe la durée de vie d'une phéromone
- Une constante T qui permet d'ajuster les probabilités
- La probabilité fixe P_f qui permet de n'exclure aucun secteur

Algorithme 3 – Colonies de fourmis adapté au découpage aérien

Pour $tour = 1 \dots tour_{max}$

Pour chaque colonie de fourmis $c = 1 \dots nb_de_zones$

Pour chaque fourmi $f = 1 \dots nb_de_fourmis$

Choisir un secteur voisin du secteur de départ de c

Déposer des phéromones sur le chemin entre les deux secteurs

Calculer grâce à C_{somme} l'énergie du nouvel état résultant du déplacement de la fourmi

Si L'énergie du nouvel état est inférieur ou **Si** l'état est accepté par la fonction de probabilité de Boltzmann

 Conserver le nouveau déplacement

Si non

 Rejeter le déplacement et considérer que la fourmi reste immobile

Fin Si

Fin Pour

Fin Pour

Evaporer les phéromones

Fin Pour

Pour conclure sur les colonies de fourmis, nous pouvons dire que notre adaptation se rapproche d'une méthode hybride entre colonies de fourmis et recuit simulé, puisque nous utilisons la règle d'acceptation de Metropolis (fonction de probabilité de Boltzmann) entre chaque déplacement de fourmis. Comme nous l'avons dit précédemment, l'initialisation de l'algorithme, s'effectue grâce au mécanisme de percolation, décrit en section 9.

11 Recuit simulé

Le recuit simulé [KGV83] [SD89] [Ing89] est une méthode d'optimisation découlant des méthodes de Monte-Carlo. Le terme de « recuit simulé » vient de la thermodynamique. Son principe est assez simple : partant d'un état initial perturbé ou « chaud », un nouvel état est créé à partir de l'état précédent par une modification élémentaire aléatoire. Il est accepté si son énergie est plus faible, sinon il est accepté avec une certaine probabilité. Ce schéma est réitéré en remplaçant l'état initial par l'état accepté tant qu'un équilibre thermodynamique n'est pas atteint. Lorsque cet équilibre est atteint, la température est diminuée. Puis les équilibres thermodynamiques sont renouvelés tant que le système n'est pas figé. Il existe différentes variantes de cette méthode, nous utiliserons la règle d'acceptation de Metropolis qui est la plus commune.

Notre adaptation du problème de découpage aérien au recuit simulé est simple. La modification élémentaire aléatoire consiste à prendre un secteur quelconque parmi toutes les zones de qualification et à l'intégrer dans une autre zone que celle d'où il vient. Tout l'intérêt de notre adaptation vient du choix de cette nouvelle zone. Elle est choisie de manière différente si la température est élevée ou basse. Dans le premier cas, quand la température est élevée, la zone choi-

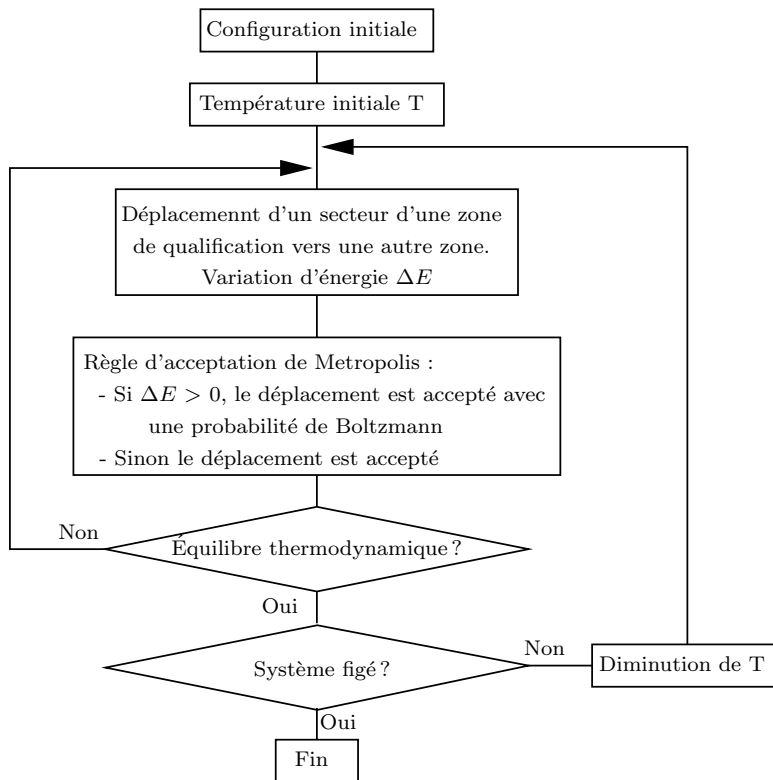


FIG. 8 – Organigramme de l’algorithme du recuit simulé appliqué au découpage aérien

sie est la zone de plus basse énergie⁸ parmi l’ensemble des zones. Et lorsque la température est basse, la zone est choisie parmi les zones connexes. Nous avons choisi ce critère de sélection afin de ne pas imposer la connexité entre les secteurs, tout en facilitant la recherche des états de faible énergie quand la température est basse. L’organigramme de notre adaptation de la méthode du recuit simulé au découpage aérien est donné figure 8. La configuration initiale est prise de manière aléatoire, ou après utilisation de la méthode de la percolation. L’équilibre thermodynamique est atteint lorsqu’un certain nombre d’état ont été refusés. Le système est figé lorsque la température est inférieure à un seuil fixé à l’avance.

12 Résultats

Dans cette section, nous allons présenter les résultats que nous avons obtenus. Nous avons appliqué les méthodes d’optimisation par recuit simulé, par colonie de fourmi hybride et par fusion/fission à notre problème de découpage du ciel aérien européen (sections 1 & 6).

Comme nous l’avons vu précédemment, notre méthode d’optimisation par

⁸ L’énergie d’une zone est le rapport entre masse aux frontières et masse interne, tel que défini en section 6.

fusion/fission possède de très nombreux paramètres. C'est un des points de notre méthode que nous devons rapidement améliorer. Un nombre trop grand de paramètre rend une méthode trop compliquée à utiliser. Et s'il est connu que donner des paramètres efficaces aux métaheuristiques est en soit un problème d'optimisation, nous savons aussi que moins il y aura de paramètres, plus la méthode sera utilisée facilement. Nous allons récapituler tous les paramètres de notre optimisation :

- Le nombre de zones de qualification *nb_de_zones*, dans lequel nous voulons répartir les secteurs.
- Le pas de la règle *pas_de_regle*.
- Les paramètres de l'optimisation par colonies de fourmis, initialisée par la percolation, lors d'une fission :
 - Le nombre de fourmis par zones de qualification.
 - Les températures initiales et finales pour le recuit simulé.
 - Une probabilité fixe de choisir un secteur.
 - La durée de vie d'une phéromone.
- Les températures maximales et minimales du plasma dans lesquelles se passent la fusion et la fission.
- L'énergie maximale que doit obtenir l'état trouvé par l'initialisation.
- Un coefficient permettant de savoir si lors d'une fission un nucléon possède une énergie assez grande pour recréer une fission.
- Pour la fonction de choix de l'étape d'initialisation :
 - Pour *choix₁*, $p_1 = \text{choix}_1(\frac{n}{2}, \frac{n}{2})$ et $p_2 = \text{choix}_1(\frac{n}{2}, \frac{3n}{2})$.
 - Pour *choix₂*, nous devons choisir γ .
- Pour la fonction de choix de l'étape d'optimisation :
 - Pour *choix_e*, les paramètres de *choix₂*, et m_α , r_α et m_β , r_β tels que $\alpha(t) = m_\alpha d(t) + r_\alpha$ et $\beta(t) = m_\beta d(t) + r_\beta$.
 - Pour *choix_n*, les coefficients k et r tels que $\alpha(t) = k d(t) + r$.
- La fonction de décroissance de la température.
 - Les suites u_1 et u_3 sont autonomes.
 - La suite u_2 prend en paramètre p qui permet de modifier la pente de la courbe, et n_{max} le nombre maximum d'itérations de la température.

La figure 9 nous montre que l'optimisation par fusion/fission donne les résultats les plus performants, tant en matière de rapidité que pour la valeur énergétique des états trouvés. Afin de présenter des résultats pertinents, chaque courbe est la moyenne de 9 optimisations appliquées à notre problème. Ces optimisations ont été réalisées pour le découpage de l'espace aérien européen en 24 zones de qualification. Pour la méthode d'optimisation par fusion/fission, nous avons utilisés les fonctions de choix, *choix₂* pour l'initialisation puis *choix_e* pour l'optimisation, ainsi que la suite de décroissance u_3 pour la température. À titre indicatif, voici la liste des paramètres utilisés :

- Le pas de croissance des probabilités, *pas_de_regle* = 0,001.
- Pour le mécanisme de règles : $\begin{cases} \text{fusion} : & \{0,85;0,1;0,04;0,01\} \\ \text{fission} : & \{0,7;0,2;0,08;0,02\} \end{cases}$
- Pour l'optimisation par colonies de fourmis :
 - 6 fourmis par atomes.
 - Température initiale : 20 000 et finale : 0.
 - Probabilité fixe : 10.
 - Durée de vie d'une phéromone : 5.
- Les températures maximales et minimales du plasma : 2 000 et 10.

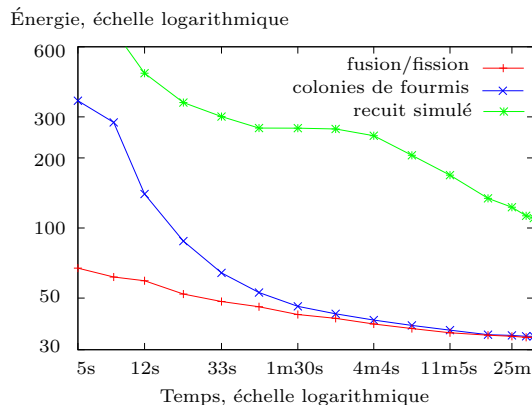


FIG. 9 – Courbes des énergies des états trouvés en fonction du temps par les méthodes d’optimisation par recuit simulé, par colonie de fourmi hybride et par fusion/fission, appliquées au problème de découpage du ciel aérien européen

- $E_{maxinit} = 70$.
- Coefficient de l’énergie pour recréer une fission : 1, 5.
- Pour $choix_2$, $\gamma = 3$.
- Pour $choix_e$, $\alpha(t) = 0,7d(t) + 0,8$ et $\beta(t) = 2d(t) + 3$.

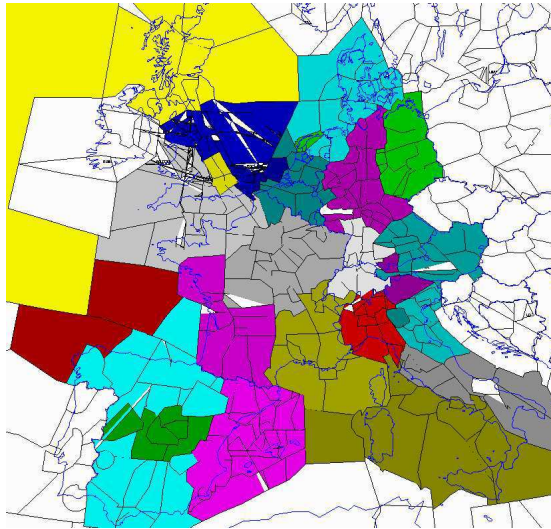
Une représentation des zones de qualification obtenues par notre méthode est donnée figure 10. Nous avons utilisé comme données pour ce découpage, l’ensemble des plans de vol européens du 8 février 2002. Nous avons intentionnellement limité notre étude à la « country core area » telle qu’elle est définie dans [BA05b]. La « country core area » comprend les pays possédant le plus fort trafic aérien en Europe.

Nous observons que les zones de qualification obtenues sont connexes (au sens de la définition donnée en section 6) et très bien délimitées. Nous remarquons aussi que les zones se superposent les unes sur les autres. Ainsi, si certaines zones ne semblent pas connexes, c’est que les secteurs de cette zones sont liés par des secteurs de niveaux supérieurs ou inférieurs. Nous constatons aussi que de nombreuses zones de qualification s’étalent sur plusieurs pays. La figure 11 permet de comparer le découpage trouvé avec le découpage qui existait fin 2002.

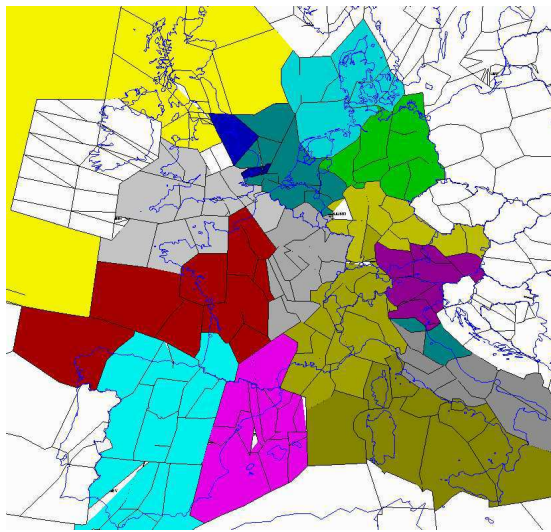
13 Conclusion

Nous avons abordé plusieurs méthodes d’optimisation. Du recuit simulé à une nouvelle méthode de « fusion/fission », en passant par les colonies de fourmis. Toutes ces méthodes servent à résoudre des problèmes NP-difficiles. Nous l’avons illustré par le problème du découpage aérien européen en zones de qualification. Ce problème de découpage, dont nous avons démontré la NP-complétude, peut paraître mal choisi pour illustrer une nouvelle méthode d’optimisation. Pourtant c’est un problème de découpage de graphe qui n’est pas très éloigné comme nous l’avons vu du problème du « minimum cut into bounded sets », ce dernier problème étant bien connu, et son dual, le « max cut », encore plus.

De nombreux progrès concernant cette méthode restent à faire. Le premier et le plus évident, consiste à réduire le nombre de paramètres de cette méthode.



(a) Niveau de vol supérieur FL180 (5 500m)



(b) Niveau de vol supérieur FL380 (11 500m)

FIG. 10 – Résultat du découpage du ciel aérien européen en 24 zones de qualification par la méthode de fusion/fission.

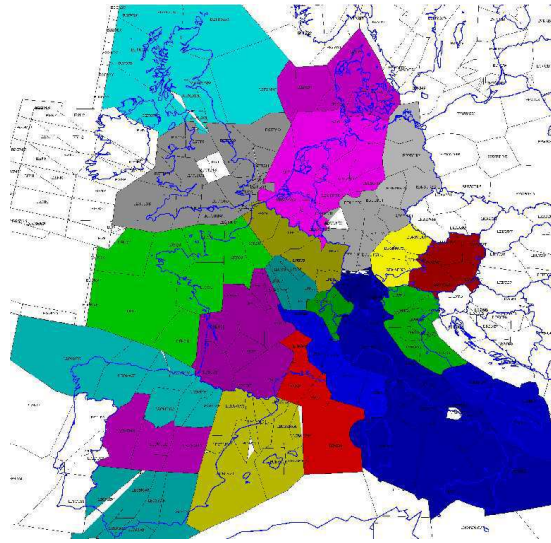


FIG. 11 – Découpage du ciel aérien européen fin 2002 au niveau supérieur FL360 (11 000m)

Cependant, il faut souligner que nous avons été assez exhaustifs dans la présentation des différentes configurations de l’algorithme, bien que ceci se soit fait un peu au détriment des résultats présentés. Considérant que de nombreuses idées valent mieux que de nombreux résultats sur un problème assez spécifique, nous avons préféré cette ouverture afin que notre méthode puisse être améliorée en partant de différents horizons.

Notre méthode, comme les autres optimisations que nous avons adapté à notre problème de découpage aérien, se divise en deux. Il y a une étape d’initialisation très importante qui est imposé par notre problème, suivie par l’optimisation proprement dite. Dans de nombreux problèmes, cette étape d’initialisation est très réduite. Notre méthode se trouve grandement simplifiée si nous n’utilisons pas l’initialisation. De même, si le nombre de paramètres utiles pour la méthode d’optimisation par colonies de fourmis n’est pas trop important, il le devient si l’on utilise la méthode de percolation comme initialisation. De plus, les fonctions de décroissance de la température, comme de choix entre fusion et fission peuvent être uniques comme cela se passe pour les autres méthodes d’optimisation. Enfin, lors de la fission, une solution plus simple que l’utilisation d’une autre méthode d’optimisation avec tous ses paramètres à redéfinir, peut être trouvée en fonction des problèmes à optimiser.

Nous espérons que le lecteur aura été intéressé par cette nouvelle méthode. Notre tâche consiste maintenant à utiliser cette nouvelle méthode sur d’autres problèmes de référence, comme le problème du voyageur de commerce, ou celui du sac à dos. Ces utilisations nous permettront d’affiner notre méthode et d’en enlever le superflu.

Références

- [AGP02] J.M. Alliot, G. Granger, and J.M. Pomeret. [Arithmetic simulation and performance metrics](#) . In *Digital Avionics System conference*, 2002.
- [All96] J.M. Alliot. [Techniques d'optimisation stochastique appliquées aux problèmes du contrôle aérien](#) . Mémoire de HDR, Mai 1996.
- [BA05a] C.E. Bichot and J.M. Alliot. [Optimisation par colonies de fourmis appliqué au découpage de l'espace aérien européen en zones de qualification](#) . In *RIVF2005*, 2005.
- [BA05b] C.E. Bichot and J.M. Alliot. [A theoretical approach to defining the European Core Area](#) . Technical report, ENAC/LOG, 2005.
- [BH57] S.R. Broadbent and J.M. Hammersley. Percolation processes i. crystals and mazes. *Proceedings of the Cambridge Philosophical Society*, 53(3) :629–641, 1957.
- [Bic04] C.E. Bichot. [Optimisation du découpage de l'espace aérien par diverses métaheuristiques](#) . Master's thesis, DEA Systèmes Informatiques, EDSYS, Juillet 2004.
- [Bla03] D. Blanc. *Précis de physique nucléaire*. Dunod, 2^e édition, 2003.
- [Che99] R. Chelouah. *Adaptation aux problèmes à variables continues de plusieurs métaheuristiques d'optimisation combinatoires*. PhD thesis, Université de Cergy-Pontoise et CEA Saclay, 1999.
- [Cou03] A. Courgey. Laboratoire d'informatique algorithmique : Fondements et applications. Rapport d'activité UMR 7089, C.N.R.S. Université Denis Diderot Paris VII, 2003.
- [DCG99] M. Dorigo, G. Di Caro, and L.M. Gambardella. [Ant Algorithms for Discrete Optimization](#) . *Artificial Life*, 5(2) :137–172, 1999.
- [Del95] D. Delahaye. *Optimisation de la sectorisation de l'espace aérien par algorithmes génétiques*. PhD thesis, ENSAE, Juillet 1995.
- [DMC96] M. Dorigo, V. Maniezzo, and A. Coloni. [The Ant System : Optimization by a Colony of Cooperating Agents](#) . *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26(1) :29–41, 1996.
- [DPST03] J. Dréo, A. Pétrowski, P. Siarry, and E. Taillard. *Métaheuristiques pour l'optimisation difficile*. Eyrolles, 2003.
- [Gia04] D. Gianazza. *Optimisation des flux de trafic aérien*. PhD thesis, Institut National Polytechnique de Toulouse (INPT), Novembre 2004.
- [GJ79] M.R. Garey and D.S. Johnson. *Computers and intractability. A guide to the theory of NP-completeness*. Freeman, 1979.
- [GJS76] M.R. Garey, D.S. Johnson, and L. Stockmeyer. Some simplified np-complete graph problems. *Theoretical Computer Science*, 1 :237–267, 1976.
- [Hao99] J.K. Hao. Métaheuristiques pour l'optimisation combinatoire et l'affectation sous contraintes. *Revue d'Intelligence Artificielle*, 13(2) :283–324, 1999.
- [Ing89] L. Ingber. [Very fast simulated re-annealing](#) . *Mathematical Computer Modelling*, 12(8) :967–973, 1989.

- [KGV83] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. [Optimization by simulated annealing](#) . *Science*, 220(4598) :671–680, May 1983.
- [Paj01] S. Pajot. *Percolation et économie*. PhD thesis, Université de Nantes, Octobre 2001.
- [P.S02] P.Siarry. Application des métaheuristiques d’optimisation en électronique. *Techniques de l’Ingénieur, traité Recherche et Innovation*, RE(8), octobre 2002.
- [SD89] P. Siarry and G. Dreyfus. *La méthode du recuit simulé : théorie et applications*. ESPCI - IDSET, 10 rue Vauquelin Paris, 1989.

Sites internet :

Le Laboratoire d’Optimisation Globale (CENA/ENAC) :

<http://www.recherche.enac.fr/opti/index>

Le projet FABOP de redécoupage du ciel aérien européen

<http://www.recherche.enac.fr/log/FABOP/>

Une introduction à la physique nucléaire :

<http://voyage.in2p3.fr/index.html>

Méthode du recuit simulé :

<http://www.ingber.com/>

Méthode des colonies de fourmis :

<http://iridia.ulb.ac.be/mdorigo/ACO/index.html>