



HAL
open science

Résolution du problème de rayon d'action d'un vol

Assia Kamal Idrissi, Arnaud Malapert, Rémi Jolin

► **To cite this version:**

Assia Kamal Idrissi, Arnaud Malapert, Rémi Jolin. Résolution du problème de rayon d'action d'un vol. Société française de recherche opérationnelle et d'aide à la décision, Feb 2018, Lorient, France. hal-02069642

HAL Id: hal-02069642

<https://hal.science/hal-02069642>

Submitted on 15 Mar 2019

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.

Résolution du problème de rayon d'action d'un vol.

Assia Kamal Idrissi^{1,2}, Arnaud Malapert¹, Rémi Jolin²

¹ Université Côte d'Azur, CNRS, I3S, France
{arnaud.malapert}@unice.fr

² Milanamos, 1047 route des Dolines, Sophia Antipolis, France
{assia.elafouani, remi.jolin}@milanamos.com

Mots-clés : *transport aérien, optimisation, plus court chemin, base de données orientée graphe*

Introduction Le transport aérien connaît depuis plusieurs décennies un très fort taux de croissance. Cette croissance a poussé les compagnies aériennes à améliorer leur qualité de service à travers la maîtrise de plusieurs facteurs sensibles comme la fréquence, le prix, ou la concurrence. Ainsi, les compagnies aériennes affrontent un ensemble de problèmes interdépendants de gestion des opérations sur différents horizons de temps [3]. À long terme, la gestion de la flotte nécessite de déterminer les dimensions et la composition de la flotte ainsi que le type d'appropriation (achat ou location). La conception du réseau opéré [2, *Route Network Development*] nécessite quant à elle de déterminer l'ensemble des routes à desservir. Une route est une séquence de vols effectuée par le même avion qui relie un aéroport de départ à un aéroport d'arrivée. À moyen terme, une compagnie doit planifier la fréquence et les horaires des vols. Finalement, à court terme, il faut affecter les avions et les équipages.

L'entreprise Milanamos développe une application d'analyse et de simulation des marchés aériens. La création d'un nouveau vol par une compagnie consiste à déterminer une origine et une destination à desservir, puis à choisir la fréquence et l'horaire pour maximiser la qualité de service (QSI). Notre objectif est d'aider le décideur à choisir l'origine et la destination puis d'accélérer le calcul des indices QSI [5]. Le QSI permet d'estimer la part de marché pour un itinéraire donné. Ainsi, ce calcul prend en compte l'ensemble des itinéraires reliant l'origine à la destination. En pratique, ce calcul doit être réalisé en quelques secondes.

Pour modéliser le réseau aérien, nous avons choisi un graphe indépendant du temps dans lequel les noeuds représentent les aéroports et un arc indique l'existence d'un vol entre les deux aéroports. Étant donné une origine et une destination, l'application affiche tous les aéroports reliés par un arc vers l'origine ou issu de la destination. Ce sous-graphe est souvent trop grand et dense pour être visualisé correctement dans l'application. De plus, certains des chemins du sous-graphe n'ont aucun intérêt en pratique. Par exemple, un trajet New-York-Paris-Los Angeles n'est pas une alternative réaliste pour relier New-York à Los Angeles. Ensuite, l'application calcule les critères de QSI et estime la part de marché pour chaque chemin de ce sous-graphe. Le *Flight Radius Problem* permet de déterminer un sous-graphe pertinent pour améliorer la visualisation et accélérer le calcul des indices QSI et des parts de marché.

Flight Radius Problem Le problème consiste à déterminer un sous-graphe maximal dans lequel au moins un chemin pertinent passe par chaque noeud. Un chemin pertinent passe par l'arc entre l'origine et la destination en limitant la perte de temps et le surcoût par rapport aux meilleurs chemins. De manière formelle, le problème est défini de la manière suivante :

Entrée Un graphe $G = (V, E)$, un arc (o, d) , et une fonction booléenne de regret R .

Sortie Un sous-graphe maximal $G' = (V', E')$ tel que chaque noeud supporte un chemin valide, *i.e.* passant par l'arc (o, d) et accepté par la fonction de regret.

En pratique, nous définissons la fonction de regret d'un chemin entre i et j passant par l'arc (o, d) pour chaque critère (temps et coût) de cette façon :

$$R_{od}^+(i, j) = l(i, j) \leq l^*(i, j) + K$$

où $l^*(i, j)$ est la longueur du plus court chemin de i à j tandis que $l(i, j)$ est la longueur du chemin passant par l'arc (o, d) . En remarquant que tout sous-chemin d'un plus court chemin est aussi un plus court chemin [1], la recherche des chemins valides peut être restreinte à la recherche des plus courts chemins valides à partir de o et arrivant à d . Remarquez que cette propriété n'est plus vérifiée si la constante K est multiplicative.

Méthodes de résolution Le graphe représentant le réseau aérien est stocké dans une base de données orientée graphe : neo4j [6]. Dans notre contexte, une base de données orientée graphe est plus adaptée à la résolution des problèmes auxquels est confronté Milanamos qu'une base de données NoSQL. Nous avons proposé deux méthodes de résolution pour le problème bicritère.

La première méthode consiste à exécuter une requête dans la base de données utilisant uniquement des fonctions disponibles dans neo4j, notamment l'algorithme de Dijkstra [1] de recherche du plus court chemin entre deux noeuds. Le problème est résolu indépendamment pour chaque critère. Pour chaque critère, l'algorithme de Dijkstra est appelé quatre fois.

La deuxième méthode est basée sur une adaptation de l'algorithme de Dijkstra de recherche des plus courts chemins partant d'un noeud. Cette adaptation repose sur deux idées principales. Le problème n'est plus résolu indépendamment pour chaque critère et l'espace de recherche est réduit entre deux appels consécutifs de l'algorithme de Dijkstra. À chaque appel de l'algorithme de Dijkstra, les noeuds sont ouverts seulement lorsque un chemin valide les atteint.

Conclusion Le *Flight Radius Problem* est un problème polynomial issu d'une application dans le domaine du transport aérien qui peut être résolu par l'algorithme proposé. Même si les deux méthodes renvoient une solution optimale après un nombre fini d'algorithmes de plus court chemin, la requête est plus lente que l'algorithme de plusieurs ordres de grandeur. Les performances de l'algorithme respectent déjà les contraintes sur le temps de réponse de l'application. Cependant, nous avons plusieurs pistes d'amélioration : une optimisation fine de l'algorithme de Dijkstra et notamment de la queue de priorité [4], mais aussi l'évaluation des performances d'autres algorithmes de plus courts chemins comme Bellman ou A*. Finalement, les résultats préliminaires montrent que le *Flight Radius Problem* peut être résolu rapidement en quelques secondes maximum et que le sous-graphe obtenu est beaucoup plus petit ce qui laisse espérer que l'accélération du calcul des indices QSI sera conséquente.

Références

- [1] Ahuja, Ravindra K and Magnanti, Thomas L and Orlin, James B. Network flows : theory, algorithms, and applications. *Prentice hall*, 108-112, 1993.
- [2] Barnhart, Cynthia and Cohn, Amy. *Airline schedule planning : Accomplishments and opportunities*. Manufacturing & service operations management, 6(1), 3-22, INFORMS, 2004.
- [3] Charfeddine Souhir. *Optimisation de l'offre d'une compagnie aérienne en environnement incertain*. Université Toulouse le Mirail-Toulouse II, 2004.
- [4] Cherkassky, Boris V and Goldberg, Andrew V and Radzik, Tomasz. *Shortest paths algorithms : Theory and experimental evaluation*. Mathematical programming, 129-174, 1996.
- [5] Jacobs, Timothy L and Garrow, Laurie A and Lohatepanont, Manoj and Koppelman, Frank S and Coldren, Gregory M and Purnomo, Hadi. *Airline planning and schedule development*. Quantitative Problem Solving Methods in the Airline Industry, 35-99, Springer, 2012.
- [6] Neo Technology. *Neo4j*. <https://www.neo4j.com>, 2017.