

UNIVERSITÉ DE MONTRÉAL

**HEURISTIQUE POUR LA CONSTRUCTION DE BLOCS
MENSUELS PERSONNALISÉS D'AGENTS DE BORD**

**PATRICK JEANDROZ
DÉPARTEMENT DE MATHÉMATIQUES
ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL**

**MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MATHÉMATIQUES APPLIQUÉES)**

JANVIER 2000

© Patrick Jeandroz, 2000



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*

Our file *Notre référence*

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-57412-1

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

**HEURISTIQUE POUR LA CONSTRUCTION DE BLOCS
MENSUELS PERSONNALISÉS D'AGENTS DE BORD**

présenté par : **JEANDROZ Patrick**

en vue de l'obtention du diplôme : **Maîtrise ès sciences appliquées**

a été dûment accepté par le jury d'examen constitué de :

M. **LANGEVIN André**, Ph.D., président

M. **SOUMIS François**, Ph.D., membre et directeur de recherche

M. **DESROSIERS Jacques**, Ph.D., membre

À Andrée, Serge et Marie-Josée,

REMERCIEMENTS

Je désire tout d'abord remercier mon directeur de recherche M. François Soumis qui a fait preuve d'un impeccable dévouement semaine après semaine tout au long de ma maîtrise. Malgré l'énorme quantité de travail qui pèse sur ses épaules, François a démontré un très grand intérêt dans la poursuite de mes recherches et chaque minute passée à lui demander conseil aura été excessivement profitable. Il est important de préciser aussi que la généreuse aide financière qu'il m'a accordée a été indispensable durant ces mois de recherches. Dans la même veine, je remercie aussi le C.R.S.N.G. pour la bourse d'excellence qu'ils m'ont offerte.

Je tiens aussi à remercier M. Beyime Tachefine qui avait déjà commencé à travailler sur ce projet avant que je n'arrive au G.E.R.A.D. Beaucoup de crédits lui reviennent de droit puisqu'il est à l'origine d'une grande partie du code informatique et qu'il a passé de nombreuses heures à me l'expliquer pour que je puisse prendre sa relève. Les nouvelles versions de code n'auraient jamais vu le jour si rapidement si ce n'avait été de sa grande disponibilité à mes débuts.

Je ne peux de plus négliger l'aide que m'ont apportée Mme Sylvie Gélinas, M. José Manuel Perez, M. Gérald Marquis, M. Éric Gélinas, M. Olivier Milon, M. Norbert Lingaya et tous les autres membres de l'équipe P.B.S.. Lors de mon long séjour chez

« ADOPT Technologies », j'ai pu, grâce à eux, travailler dans un milieu des plus accueillant et propice à l'apprentissage.

Je dédie des remerciements les plus sincères à mon père Serge et à ma mère Andrée pour l'éducation exemplaire qu'ils m'ont donnée et pour tout le soutien affectif et financier qu'ils ont su m'offrir durant toutes ces années passées sur les bancs d'école. C'est en me servant de vous comme modèles que je suis devenu ce que je suis présentement.

Merci à toi chère Marie-Josée, le support que tu m'apportes depuis maintenant sept ans m'a permis de toujours rester optimiste et de garder confiance en moi, même dans les moments les plus pénibles.

RÉSUMÉ

Ce mémoire décrit un nouvel algorithme heuristique qui permet de construire des blocs mensuels personnalisés pour les membres d'équipage d'une flotte aérienne. Nous définissons un bloc comme une série de « rotations », de repos, d'entraînements et de congés. Une rotation est une séquence de vols dont l'origine du premier vol et la destination du dernier correspondent.

Les blocs qui sont attribués aux employés doivent être construits de façon à maximiser leurs préférences, sans toutefois nuire à la réalisation de l'ensemble des blocs résiduels. De façon plus précise, la confection des blocs s'effectue de manière séquentielle, en privilégiant les employés les plus seniors au commencement et en complétant par la résolution des employés plus juniors. Les employés ont un choix de préférences (ex : ne pas travailler les fins de semaine, commencer les journées de travail après 8h00 am, ...) dont ils déterminent eux-mêmes la pondération en fonction de leurs goûts. Ces données permettent d'associer des coûts aux arcs d'un réseau. Les arcs correspondent aux rotations qui peuvent être effectuées par un employé considéré. Le bloc optimal d'un employé s'obtient en résolvant, à l'aide d'un algorithme de programmation dynamique, un plus court chemin avec contraintes de ressources.

Des compteurs dynamiques sont ajoutés pour évaluer la faisabilité des blocs résiduels. Un attribut est défini comme étant une compétence requise par un employé

qui effectuera une certaine rotation (ex : être bilingue, parler le portugais,...). Des couples de compteurs « C₁ » et « C₂ » sont créés pour chacun des attributs de même que pour l'attribut dit « global » (tous les arcs et tous les employés). On définit pour chaque attribut des intervalles de temps tels que les débuts d'intervalles correspondent à un début ou une fin de rotation possédant l'attribut considéré. Pour chacun de ces intervalles, les compteurs C₁ qui évaluent « l'offre » (employés résiduels disponibles) et la « demande » (rotations non encore couvertes) sont mis à jour chaque fois qu'un nouveau bloc est fixé. Cela permet donc de forcer un employé courant à effectuer, s'il est disponible et qualifié, une tâche dans un intervalle précis si l'offre est égale à la demande. Ces compteurs permettent donc d'évaluer, de manière excessivement rapide, la faisabilité des blocs restants et de modifier en conséquence le réseau servant à la construction du bloc courant.

Les compteurs C₂ permettent pour leur part de calculer, pour chaque attribut, le nombre de crédits de vols (unité de mesure correspondant à un nombre d'heures de travail effectué) des tâches résiduelles versus la disponibilité, calculée aussi en crédits, des employés encore disponibles. Le compteur C₂ « global » détermine le critère d'arrêt de l'heuristique. Les blocs des employés restants sont alors construits avec la méthode exacte qui utilise le logiciel Gencol. En effet, sans ce critère d'arrêt, la relaxation de certaines contraintes et les approximations de l'heuristique pourraient mener à l'infaisabilité du problème.

La rapidité d'exécution est le principal avantage de cet algorithme puisque ces problèmes étaient jusqu'à présent résolus à l'aide de Gencol, basé sur un algorithme de génération de colonnes qui emploie des outils complexes tels que l'algorithme du simplexe et des arbres de branchement. Les essais comparatifs avec Gencol démontrent que le pourcentage d'erreur est relativement faible et que le temps d'exécution est nettement inférieur à celui de Gencol. Ceci permettra de résoudre des problèmes de pilotes comprenant beaucoup plus d'employés que ceux qui sont résolus actuellement. Tout cela répond à un besoin pressant puisque le nombre de pilotes pour un même type d'avion est en croissance dans plusieurs compagnies aériennes et les temps de résolution deviennent excessifs. De plus, cet heuristique ouvre la possibilité de résoudre les problèmes d'agents de bord qui sont beaucoup plus nombreux que les pilotes dans les compagnies aériennes. Il s'agit donc d'un nouveau marché encore plus considérable que le marché des pilotes pour lequel le système actuel est commercialisé.

ABSTRACT

This master's thesis considers a new heuristic algorithm to construct personalized monthly schedules for employees in the airline industry. A block is a series of pairing, rests, training periods, and days off.

The problem consists in constructing blocks to the employees, while considering a set of weighted bids that reflect individual preferences. However, each time a block is build, a so-called residual problem has to be feasible. This means all the unsolved employees must be able to get a legal block (a block which respect federal laws and collective agreement of the company) so that these blocks cover all the residual pairings. The assignment is done sequentially, beginning with the most senior employee and finishing with the most junior one. An employee selects his preferences by associating numbers with some sentences (ex: don't work on week ends, begin working day after 8h00 am, ...). After that, the bids of the employee are used to put costs on a network. The arcs of this network correspond to the residual pairings that can be made by the employee. A dynamic programming algorithm is used to find a shortest path, which is the optimal block of the considered employee.

The Heuristic use dynamic counters to evaluate the feasibility of the residual problem. A feature is a skill that an employee must have to take part of a trip (ex: being bilingual). For each feature of the problem, and for a new global feature (every

employee and every pairing of the problem has it), a couple of counters C_1 and C_2 are associated. For each of these features, we determine time intervals so that each interval beginning corresponds to the beginning or the end of a pairing with the considered feature. Moreover, after every employee resolution, the demand (residual pairings crossing the interval) and the offer of the C_1 counters (residual crewmembers who are available to work in the interval) are updated for each one of these intervals. These values are used to force a current employee to work during a considered interval if *offer* = *demand* and if he is qualified and available.

For each feature of the problem, the C_2 counters are used to count the total flight credits (hours of work) of the residual pairings and to evaluate if there's too much (or not enough) work to do for the residual crewmembers. The global C_2 counter determine when the exact method must be used to solve the residual problem. Without this stopping criterion, the residual would have a chance to become unsolvable because of some global constraints relaxation.

Actually, the program named Gencol (from the G.E.R.A.D. laboratory) is used to solve these Preferential Bidding problems. Gencol use an exact method which is based on a column generation algorithm that uses complex tools like the simplex and a branch and bound algorithm. This method use to found the exact optimal solutions, but the resolution time is much to high with big problems. Therefore, the heuristic method is there to lower this resolution time. For the time being, the performed tests have been

excellent: the time resolution has been highly reduced in the bigger problems, and the error percentage is insignificant. Consequently, the heuristic will be used to solve larger pilot problems. This is welcome if we think of the growing number of pilots in the airline industry and the fact that the resolution time is becoming excessive. In fact, the heuristic should be able to solve crewmembers problems which are much bigger than the pilots problems. Furthermore, this is a much considerable market.

TABLE DES MATIÈRES

DÉDICACE.....	iv
REMERCIEMENTS.....	v
RÉSUMÉ.....	vii
ABSTRACT	x
TABLE DES MATIÈRES	xiii
LISTE DES TABLEAUX.....	xv
LISTE DES FIGURES.....	xvi
INTRODUCTION.....	1
CHAPITRE 1 : FABRICATION DE BLOCS MENSUELS PERSONNALISÉS	3
1.1 Contexte	3
1.2 Le « Preferential Bidding System »	6
1.2.1 Définitions générales.....	6
1.2.2 Terminologie et conventions chez Air Transat.....	10
1.2.3 Méthode générale.....	13
CHAPITRE 2 : MÉTHODE DE RÉOLUTION EXISTANTE.....	16
2.1 Réseau et algorithme de plus court chemin	16

2.1.1	<i>Structure du réseau</i>	16
2.1.2	<i>Coûts des arcs du réseau</i>	21
2.1.3	<i>Les ressources et le plus court chemin</i>	22
2.2	Résolution par génération de colonnes (GENCOL)	25
CHAPITRE 3 : NOUVELLE MÉTHODE DE RÉOLUTION		29
3.1	Les compteurs C1	32
3.1.1	<i>La demande</i>	34
3.1.2	<i>L'offre</i>	37
3.1.3	<i>Modifications du réseau</i>	45
3.2	Les compteurs C2.	50
3.3	Hybride Gencol-Heuristique	55
3.4	Relation entre l'Heuristique et la méthode exacte.	57
3.5	Open Time	63
CHAPITRE 4 : TESTS		65
4.1	Juin 1998.....	66
4.2	Juillet 1998.....	74
4.3	Août 1998.....	81
CONCLUSION		85
RÉFÉRENCES BIBLIOGRAPHIQUES		88

LISTE DES TABLEAUX

Tableau 1.1 – Deux types de personnel d'équipage.	9
Tableau 1.2 – Catégories d'activités chez Air Transat.....	10
Tableau 1.3 – Paramètres globaux chez Air Transat.....	11
Tableau 3.1 – Résultats comparatifs (Gencol vs. Hybride).....	56
Tableau 4.1 – Test (juin 1998)	67
Tableau 4.2 – Test (juillet 1998 , 183 premiers employés).....	75
Tableau 4.3 – Test (août 98 , 1 ^{er} au 59 ^{ième} employé)	82

LISTE DES FIGURES

Figure 2.1 – Description du réseau.....	18
Figure 2.2 – Chemins avec consommation de ressource.	23
Figure 3.1 – Modification de l’arc de rotation.	34
Figure 3.2 – Calcul de la demande du compteur C ₁	35
Figure 3.3 – Exemple d’un mauvais calcul avec l’Offre ₁	38
Figure 3.4 – Preuve que $Offre_2 \geq OFFRE$	41
Figure 3.5 – Analyse des compteurs C ₁ (arcs TRP).....	45
Figure 3.6 – Analyse des compteurs C ₁ (arcs LU).....	46
Figure 3.7 – Analyse des compteurs C ₁ (Arcs LD & LN)	47
Figure 3.8 – Analyse des compteurs C ₁ (Arcs DO)	48
Figure 3.9 – Analyse des compteurs C ₁ (Arcs CN).....	49
Figure 3.10 – Représentation graphique d’un compteur C ₂	52
Figure 3.11 – Compteur C ₂ global avec Open Time.....	64
Figure 4.2 – Compteur C ₂ Global (Juin 1998, 1 ^{er} au 190 ^{ième} employé)	73
Figure 4.3 – Compteur C ₂ Global (Juillet 1998, 1 ^{er} au 183 ^{ième} employé)	80
Figure 4.4 – Compteur C ₂ Global (Août 1998, 1 ^{er} au 59 ^{ième} employé)	84

INTRODUCTION

En ce début du 21^{ème} siècle, les compagnies aériennes doivent faire face à des problèmes de gestion qui sont de plus en plus ardues vu l'augmentation considérable de personnel et du nombre d'avions. Alors qu'il y a quelques années toutes ces planifications étaient effectuées à la main, certaines le sont maintenant à l'aide de logiciels informatiques qui permettent des économies de plusieurs millions de dollars annuellement. Quelquefois, malheureusement même les ordinateurs les plus puissants ne permettent pas de résoudre dans des temps raisonnables certains problèmes complexes. En conséquence, beaucoup de recherche reste à faire et de nouveaux algorithmes doivent être développés pour permettre d'obtenir les économies potentielles tout en permettant aux employés de disposer de la meilleure qualité de vie possible.

Le système « PBS 500 » (Preferential Bidding System), qui est développé par la compagnie « AD OPT Technologies », est présentement utilisé par plusieurs importantes compagnies aériennes telles que Air Canada, TWA, America West, Air Transat et Delta Airlines. Ce système permet de construire le bloc mensuel d'un employé en tenant compte de ses préférences et de diverses contraintes. Un bloc mensuel assigné à un employé constitue en fait son horaire de travail pour une période d'un mois. Ce bloc est constitué de périodes de vol, de repos, d'entraînement et de congés. La formation des blocs doit être faite de manière à optimiser l'utilisation du personnel tout en maximisant le taux de satisfaction de celui-ci. Avant la résolution du

problème, le programme reçoit en entrée les rotations construites antérieurement ainsi que les préférences des employés sous forme de pointage. Ce système de pointage permet à chaque employé d'exprimer ses goûts en cotant certaines options telles que : « ne pas commencer à travailler avant 8 heures am », « ne pas travailler la fin de semaine », « effectuer des vols à destination de l'aéroport X » etc.

Le « cœur » du système, le logiciel d'optimisation Gencol, possède l'énorme avantage de traiter une formulation optimale du problème. Cependant, la défaillance se situe au niveau du temps d'exécution : celui-ci est très grand pour des problèmes de plus de 200-300 employés et la version présente du code ne permet pas de traiter les problèmes d'agents de bord des grandes compagnies qui comptent quelques milliers d'employés. C'est pour cette raison qu'il est nécessaire de créer une nouvelle méthode de résolution plus rapide. L'heuristique présentée dans ce présent mémoire possède les qualités premières d'un bon algorithme : sa rapidité de résolution est exemplaire et son taux d'erreur est relativement faible si on compare les résultats obtenus à ceux de l'optimiseur Gencol.

Dans un premier temps, la théorie sur le système « PBS 500 » et l'heuristique seront vues en détails et des résultats obtenus à l'aide de multiples données d'Air Transat seront par la suite étudiés. Des comparaisons avec les résultats de Gencol pourront alors mettre en évidence l'efficacité de l'heuristique.

CHAPITRE 1 : FABRICATION DE BLOCS MENSUELS PERSONNALISÉS

1.1 Contexte

À l'origine, les compagnies aériennes ont à gérer plusieurs éléments qui doivent s'agencer avec une parfaite cohésion. En effet, il est évident qu'il existe une quantité d'étapes à franchir avant de pouvoir fixer tous les éléments nécessaires à un vol d'avion : lieux de départ et de destination, horaire du vol, avion, pilotes et agents de bords.

De façon à diminuer la complexité de ces problèmes, la résolution s'effectue de manière séquentielle, i.e., en déterminant premièrement l'horaire des vols et le routage des appareils pour ensuite confectionner les rotations d'équipage et les blocs des employés. Toutes ces étapes constituent de considérables défis technologiques et énormément de recherches y ont été et y sont encore consacrées.

Voici une description simplifiée de chacune des quatre étapes :

1) Fixer l'horaire des vols

Cette toute première phase, généralement effectuée par le personnel du marketing, consiste à créer les segments de vols (portion de vol déterminée par le décollage d'un avion et son atterrissage) pour une période donnée. Des études statistiques et probabilistes doivent avoir lieu au sujet de la demande des clients

et de l'offre des concurrents. Tout cela doit de plus tenir compte du temps alloué dans les différents aéroports pour faire atterrir ou décoller les avions.

2) Déterminer l'affectation et le routage des avions

Vu les coûts astronomiques associés au vol d'un avion, cette étape est importante en terme des économies d'argent pour les compagnies aériennes. Il est relativement difficile de déterminer les bons types d'avions qui seront assignés aux segments de vols pour pouvoir couvrir ceux-ci de la façon la plus efficace (i.e. la moins coûteuse) possible. Diverses contraintes doivent être prises en compte et ce, pour chaque type d'appareils : capacité, vitesse, nombre d'avions disponible en tout temps, contraintes et coûts d'entretien, etc.

3) Former les rotations d'équipage

Cette étape est, tout comme la précédente, d'une importance capitale en terme économique. La différence principale vient du fait que le niveau de complexité est nettement plus élevé. Le but est en effet de former, à coût minimum, des rotations d'équipage qui pourront couvrir tous les vols de l'étape précédente en respectant les lois gouvernementales et les conventions de travail des employés. Ces rotations sont définies comme des ensembles de segments de vols consécutifs entrecoupés de repos réglementaires et partant d'une base pour ensuite y revenir. Il est à noter qu'une équipe d'employés peut prendre place à bord d'un avion en tant que passagers aux fins de se déplacer entre deux aéroports durant une rotation.

4) Construire les blocs mensuels

Cette dernière phase consiste en la création de blocs mensuels réglementaires, pour les employés de la compagnie, qui devront attribuer aux employés de manière optimale toutes les rotations déterminées à l'étape précédente. On définit le bloc mensuel d'un employé comme une suite de rotations, de repos, de congés et d'entraînements sur un horizon d'un mois.

Trois différentes méthodes sont utilisées pour traiter ce problème :

- *Bidline* : Des blocs sont formés dans un premier temps et les employés choisissent par la suite, par ordre d'ancienneté, ceux qui sont les plus attrayants pour eux.
- *Rostering* : Des blocs sont créés pour chacun des employés mais en tenant compte de certaines activités fixes (entraînements, repos, congés, etc) qui ont été préassignées à ceux-ci.
- *Preferential Bidding* : Cette dernière méthode, qui constitue le sujet du présent mémoire, permet de créer des blocs entièrement personnalisés basés sur les préférences pondérées des employés. Cette technique, qui possède aussi les avantages du Rostering, est la plus intéressante pour les employés et les compagnies aériennes et s'établit de plus en plus comme étant la norme dans ce domaine. Des explications plus précises suivront dans les prochains chapitres du mémoire.

1.2 Le « Preferential Bidding System »

Le système PBS 500 présentement commercialisé par la compagnie AD OPT Technologies est utilisé par plusieurs chefs de file du transport aérien tels que : Air Canada, Air Transat, Delta Airlines, TWA et SwissAir. Ce logiciel produit présentement des blocs optimaux pour des groupes pouvant compter jusqu'à 300 employés.

Dans cette section, la structure générale du système PBS sera abordée de même que certaines règles importantes des conventions collectives d'Air Transat. En effet, les tests effectués l'ont été grâce aux données d'Air Transat fournies par AD OPT.

1.2.1 Définitions générales

Tout au long du mémoire, plusieurs mots précis seront utilisés. Pour en faciliter la compréhension, voici quelques définitions détaillées :

- **Base** : Aéroport auquel sont affectés un groupe de membres de l'équipage. Généralement, ces employés sont domiciliés à la base ou dans les environs. Les horaires de travail doivent être construits tel que ces membres d'équipage soient de retour à leur base pour leurs périodes de congé.
- **Repos** (« rest ») : Période de repos allouée aux employés durant laquelle ceux-ci ne sont aucunement à la disposition de la compagnie pour travailler. Ces repos sont généralement assignés suite à un vol ou à un entraînement.

- Congé (« *day off* ») : Période de repos de plus longue durée que le repos. Généralement aux moins un ou deux jours.
- Tronçon de vol (« *leg* ») : Portion de vol déterminée par le décollage et l'atterrissage de l'avion à destination.
- Entraînement (« *training* ») : Période durant laquelle les employés effectuent de l'entraînement (ex : simulateur de vol).
- Rotation ou Tâche (« *Pairing* » , « *Trip* ») : Suite chronologique de tronçons de vol et de repos telle que le départ du premier tronçon de vol et l'arrivée du dernier tronçon de vol ont lieu à la base. Tous les membres d'équipage qui effectuent une même rotation doivent donc être assignés à une même base. Généralement, une rotation est d'une durée allant de quelques jours à quelques semaines.
- Attribut (« *Feature* ») : Caractéristique que doit posséder un employé pour pouvoir effectuer une certaine rotation précise. (ex: parler le grec, être bilingue, ...)
- Tâche fixe : Entraînement, rotation, ou congé qui est assigné à un employé avant la création de son bloc.
- Crédits de vol : Unité de mesure qui représente le nombre d'heures de travail associées à une rotation.
- Bloc mensuel : Ensemble de rotations, d'entraînements et de congés assignés à un employé pour une durée d'un mois.
- Bloc court (« *Low block* ») : Définition identique à celle du bloc mensuel excepté le fait que le bloc court possède moins de crédits de vol au total. Il y a généralement un nombre très limité de blocs courts qui sont assignés à chaque mois.

- Employé assigné (« *Block holder* ») : Titre pour désigner un employé qui obtient, pour le mois donné, un bloc fixe construit selon ses préférences (bloc mensuel courant ou bloc court).
- Réserve : Titre pour désigner un employé ne possédant pas de bloc établi à l'avance comme pour l'employé assigné. Cet employé doit être disponible en tout temps pendant certaines périodes prédéterminées du mois.
- Rotations non assignées (« *Open time* ») : Représente l'ensemble des rotations qui ne sont pas effectuées par des employés assignés. Celles-ci sont donc attribuées aux employés en réserve.
- Blocs ouvert (« *Open time block* ») : Bloc non assigné à un employé, contenant des rotations de l'open time mais ne respectant pas nécessairement la convention collective des employés. Des explications supplémentaires à ce sujet seront vues au chapitre 2.

Il existe de plus deux grands types de personnel d'équipage :

Tableau 1.1 – Deux types de personnel d'équipage.

<u>PERSONNEL</u>	<u>DESCRIPTION</u>
PNC	« Personnel navigant commercial ». Inclut les chefs de cabines et l'ensemble des agents de bord.
PNT	« Personnel navigant technique ». Inclut les pilotes, copilotes et mécaniciens. Ils sont qualifiés et assignés à un seul type d'appareil.

Il est à noter que dans le présent mémoire, nous nous intéressons plus particulièrement aux PNC, vu leur plus grand nombre et leurs conventions collectives légèrement simplifiées par rapport aux PNT.

1.2.2 Terminologie et conventions chez Air Transat

Dans cette section, il sera traité des termes et conventions collectives qui régissent la création des blocs des employés chez Air Transat. Il est à noter que les règles propres à cette compagnie ne font pas de l'heuristique un algorithme exclusif à celle-ci. En effet, la majorité des compagnies, particulièrement en Amérique du Nord, possèdent sensiblement le même type de conventions collectives.

Tout d'abord, il existe trois principales catégories d'activités chez Air Transat:

Tableau 1.2 – Catégories d'activités chez Air Transat.

<u>ACTIVITÉ</u>	<u>DESCRIPTION</u>
TRP (trip)	Rotation pouvant débuter et se terminer à n'importe quelle heure de la journée. Un temps de repos doit être prévu après cette activité.
TRN (training)	Période d'entraînement (ex : simulateur) toujours <u>fixée</u> à l'avance et pouvant débuter et se terminer à n'importe quelle heure de la journée. Un temps de repos doit être prévu après cette activité.
GND (ground)	Période où un employé est non disponible (ex : congé). Ces périodes sont <u>fixées</u> à l'avance et doivent couvrir une ou plusieurs journée(s) complète(s) débutant à minuit et se terminant à minuit. Aucun temps de repos n'est alloué avant ou après.

Voici maintenant les principaux paramètres globaux que les utilisateurs du système PBS peuvent ajuster chez Air Transat. Peu de ces paramètres sont modifiés par les planificateurs.

Tableau 1.3 – Paramètres globaux chez Air Transat.

<u>PARAMÈTRE</u>	<u>DESCRIPTION</u>
Maximum block time	Nombre maximal de crédits de vol que peut posséder un employé assigné. Fixé à 85 dans les tests effectués.
Minimum block time	Nombre minimal de crédits de vol que peut posséder un employé assigné, excepté s'il a un « low time block » (voir prochaine rubrique). Fixé à 68 dans les tests effectués.
Low block time	Détermine le nombre minimum de crédits que peut contenir un bloc court. (< minimum block time).
Maximum number of low time block	Détermine le nombre maximal d'employés à qui peuvent être attribués des « low time block ». Fixé à 0 dans les tests.
Maximum open time	Nombre maximal de crédits de vol qui peuvent ne pas être affectés aux employés assignés . Fixé à 0 dans les tests.
Maximum number of open time block	Nombre maximal de blocs qui pourront être formés pour couvrir les rotations de l'open time. Fixé à 0 dans les tests.
Home base rest (HBR) (standard)	Nombre minimal d'heures de repos à être alloué suite à une rotation (TRP) normale. Fixé à 14h dans les tests.

HBR (after intl. pairing)	Nombre minimal d'heures de repos à être alloué suite à une rotation (TRP) internationale. Fixé à 24h dans les tests.
HBR (after a trip of 15h00 or more)	Nombre minimal d'heures de repos à être alloué suite à une rotation (TRP) de plus de 15h de crédits. Fixé à 24h dans les tests.
HBR (after a trip with 3h00 between 0h00 and 06h00)	Pour les PNC (légèrement différent pour les PNT), ce paramètre donne le nombre minimal d'heures de repos à être allouées suite à une rotation de plus de 3h ayant eu lieu entre 0h00 et 06h00. Fixé à 18h dans les tests.
HBR (Active ground activity to pairing)	Détermine le nombre minimal d'heures de repos à allouer suite à une activité TRN. Fixé à 24h dans les tests.
Maximum consecutive days on reserve	Détermine le nombre maximal de jour de réserve que l'on peut attribuer à un employé. Fixé à 6 jours dans les tests.
Minimum number of reserve	Nombre minimal d'employés qui doivent être en réserve. Fixé à 0 dans les tests.
Maximum number of reserve	Nombre maximal d'employés qui peuvent être en réserve. Fixé à 10 000 dans les tests (pas de limite).
Maximum number of consecutive days worked	Nombre maximal de jours consécutifs de travail pouvant être alloués. Fixé à 35 jours dans les tests (pas de limite).

1.2.3 Méthode générale.


À l'aide des notions vues aux sections précédentes, on peut maintenant résumer deux grandes catégories de contraintes qui prennent place lors de la confection du bloc mensuel d'un employé :

- **Contraintes locales :** Contraintes qui font que le bloc de l'employé courant est légal (respecte les lois gouvernementales et la convention collective de la compagnie):
 - ✓ Nombre d'heures de repos suffisant suite aux activités TRP ou TRN.
 - ✓ Nombre de crédits de vol légal. (compris entre *Maximum Block Time* et *Minimum Block Time*)
 - ✓ La convention collective de l'employé est respectée.
- **Contraintes globales :**
 - ✓ Chaque employé assigné possède un bloc qui respecte les contraintes locales.
 - ✓ Chaque rotation est couverte exactement une fois, soit par un employé assigné, soit par un bloc ouvert.
 - ✓ Le nombre de crédits des rotations non assignées (open time) et le nombre de blocs ouverts sont conformes aux paramètres fixés par le planificateur.

En conséquence, la principale notion qui complexifie le problème de confection des blocs mensuels est le fait de respecter, en tout temps lors de la résolution, les contraintes globales.

Voici, de façon très générale, le protocole suivi par le système PBS :


1) DÉTERMINER UN CADRE INITIAL.

- 
- a) Déterminer quels employés seront assignés et quels employés seront réserves. (n = nombre d'employés assignés)
 - b) Classer les employés assignés en ordre d'ancienneté stricte selon les données de la compagnie aérienne.

(Note: Les deux points précédents ne seront pas abordés dans ce mémoire.)

2) CONSTRUIRE LES BLOCS DES EMPLOYÉS.

Traiter de manière séquentielle ($k = 1 \dots n$) les employés assignés.

- 
- a) Choisir le prochain employé le plus ancien(**employé courant k**) et calculer le coût des rotations de son réseau en fonction de ses préférences.
 - b) Résoudre le **problème k** . (Trouver un bloc optimal pour l'employé courant qui respecte les contraintes locales et globales pour les tâches résiduelles et employés résiduels ($k+1 \dots n$).
 - c) Fixer la solution de cet employé courant et retourner en a).

(Voir la page suivante pour des détails au sujet des points a, b et c.)

- a) Un réseau personnalisé est construit pour chaque employé assigné. Le réseau d'un employé courant contient entre autre les rotations résiduelles sur lesquelles des coûts sont calculés en fonction de ses préférences.

- b) Grâce à ces coûts, un algorithme de programmation dynamique est utilisé pour trouver un plus court chemin optimal qui respecte les contraintes locales pour cet employé courant. La solution au problème k est un chemin optimal qui respecte en plus l'ensemble des contraintes globales.

- c) Fixer la solution pour l'employé k consiste à lui attribuer un bloc de valeur maximale. Dans certains cas, plusieurs blocs sont possibles mais un seul est choisi. Le choix (arbitraire) étant fait, on enlève certains arcs des réseaux des employés résiduels. Ces arcs correspondent aux rotations du bloc de l'employé courant qui ne feront jamais partie des blocs des employés résiduels.

CHAPITRE 2 : MÉTHODE DE RÉOLUTION EXISTANTE

Les deux méthodes de résolution abordées, soit l'heuristique et GENCOL, ont un important point en commun. Il s'agit du fait que tous les deux emploient un algorithme de programmation dynamique appliqué à un réseau acyclique pour créer les blocs mensuels. Un bloc mensuel pour un employé est donc en fait un chemin allant d'une origine à un puits dans ce réseau.

2.1 Réseau et algorithme de plus court chemin

Cette section présente tout d'abord la physionomie du réseau en décrivant les nœuds, les arcs, la structure de coût décrivant les préférences du personnel et la notion de ressource utilisée pour modéliser les contraintes de la convention collective. Par la suite, la méthode de résolution du plus court chemin avec contraintes de ressource sera abordée.

2.1.1 Structure du réseau

Comme il sera décrit clairement au cours des prochaines lignes, les réseaux contiennent plusieurs types d'arcs : rotations, repos, tâches fixes, etc. On utilise réseaux au pluriel puisque pour chacun des employés, un réseau particulier est créé à partir du même réseau global qui contient tous les arcs. En effet, puisque certains employés possèdent des tâches fixes, cela nous permet de supprimer dans leurs réseaux des arcs inutiles pour le plus court chemin.

Avant d'entrer dans les détails de la constitution des réseaux, il est important de définir deux types d'arcs de rotations qui pourraient être utilisés :

- Tâche mère : Arc représentant une rotation pour tous les membres d'équipage.
(ex : sept agents de bord)
- Tâche fille : Arc représentant une rotation pour un membre d'équipage en particulier. (ex : un des sept agents de bords)

Dans le cas de Gencol et de l'heuristique, ce sont les tâches filles qui sont utilisées. De nouveaux développements seront faits dans les prochains mois afin d'utiliser les tâches mères et ainsi réduire la taille des réseaux. Il en sera question un peu plus loin.

On peut définir le réseau d'un employé e comme un graphe orienté acyclique $G_e^k = (V_e^k, A_e^k)$ où V_e^k et A_e^k sont respectivement l'ensemble des nœuds et l'ensemble des arcs du réseau de l'employé e . L'indice k signifie que la résolution en est rendue au k ième employé assigné. On peut donc remarquer par exemple que pour tout employé e et k respectant $k \leq e \leq \text{nb. d'employés assignés}$, A_e^{k+1} est obtenu de A_e^k en supprimant les rotations du bloc k et les arcs qui lient ces rotations au reste du réseau. C'est à partir de ces réseaux que l'algorithme de plus court chemin est effectué. Pour bien assimiler cette notion, la page suivante contient un exemple graphique permettant de comprendre la structure générale du réseau d'un employé. Il est à noter que le graphique suit l'échelle du temps, i.e., une activité à gauche survient avant celle de droite.

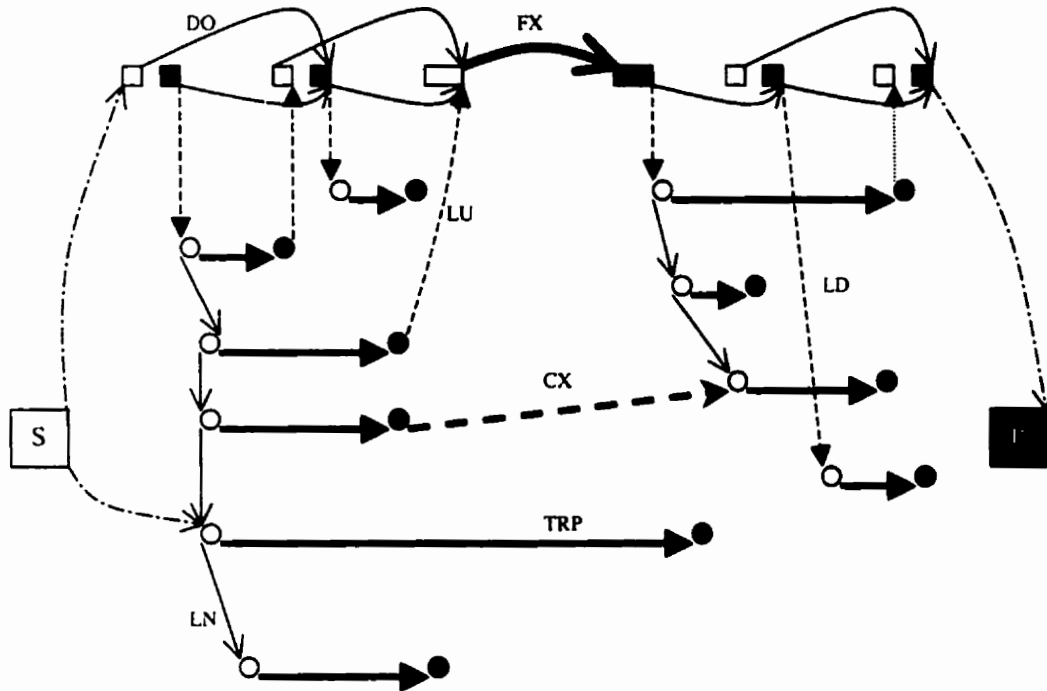



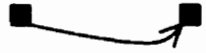
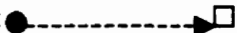
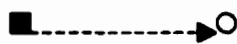




Figure 2.1 – Description du réseau.




- Les **rotations** (TRP) sont représentées par des arcs :  où le nœud blanc et le nœud noir correspondent respectivement au début et à la fin de la rotation.
- Les **tâches fixes** (FX) sont représentées par des arcs :  où le nœud blanc et le nœud noir correspondent resp. au début et à la fin de la tâche.
(Note: Ces tâches fixes sont des activités TRN ou GND (voir chapitre 1))
- Les **congés ou « days-off »** (DO) sont représentés par deux types d'arcs, tous deux d'une « durée » de 24 heures.

- Les arcs  correspondent à un congé pris suite à une rotation s'étant terminée la journée précédente.
- Les arcs  correspondent à des journées supplémentaires de congé. Ces arcs sont donc utiles lorsque 48 heures ou plus de congé sont nécessaires.
- Les arcs « **link up** » (LU) sont représentés par des arcs : 

Ils connectent la fin d'une rotation au début d'une tâche fixe ou au début d'une journée de congé (DO)
- Les arcs « **link down** » (LD) sont représentés par des arcs : 

Ils connectent la fin d'une journée de congé ou d'une activité fixe au début d'une rotation.
- Les arcs « **Lien** » (LN) sont représentés par des arcs : 

Ils lient toutes les rotations d'une journée (24h) selon l'ordre chronologique des débuts de tâches.
- Les arcs « **Connexion** » (CX) sont représentés par des arcs : 

Ils correspondent au passage d'une rotation à une autre.  correspond au début de la première rotation pouvant être effectuée en respectant le repos minimal requis par le TRP prédécesseur.
- Le nœud **Source** et le nœud **Puits** sont représentés par :  et  respectivement. Certains arcs et nœuds qui leur sont associés ont été volontairement omis de manière à faciliter la lisibilité du graphique.

Il est à noter qu'un récent mémoire de maîtrise porte sur la réduction de réseau pour le problème d'horaires de personnel aérien. Une des techniques présentée dans le mémoire de M. Laurent Deirmendjian pourra éventuellement être développée pour être utilisée en parallèle avec l'heuristique. Il s'agira de construire le réseau à l'aide des *tâches mères* au lieu des *tâches filles*. La taille du réseau sera alors réduite par un facteur d'environ 5 et les demandes en temps et en mémoire vive de l'algorithme de plus court chemin seront bien moindres. Les tâches mères pourront tout simplement posséder un compteur indiquant le nombre de tâches filles résiduelles. Cette technique sera facilement implantable dans l'heuristique puisque le réseau utilisé n'est pas « multi-commodités ». En effet, lors de la résolution d'un employé, un seul exemplaire de chaque tâche est nécessaire puisque aucun plus court chemin n'est effectué simultanément pour les employés résiduels. Il est intéressant de noter que malgré le fait que Gencol emploie un réseau multi-commodités, une nouvelle version pourra aussi être développée à l'aide des tâches mères.

2.1.2 Coûts des arcs du réseau

Une fois le réseau construit pour un employé donné, des coûts doivent être associés aux arcs de rotation et de repos pour permettre l'application de l'algorithme de plus court chemin. Ces coûts devant refléter les préférences de l'employé traité, plus un arc est « attirant » pour celui-ci, plus cet arc aura une forte valeur négative. De plus, certaines préférences nécessitent la création de nouveaux arcs spéciaux. Les employés expriment leurs préférences à l'aide d'un outil du logiciel PBS : le « Bidder Graphical Interface ». Cette interface leur présente une liste de choix sur lesquels ils peuvent poser une cote allant de 0 à 1000. Voici quelques exemples tirés de l'application à la compagnie « Delta Airlines » :

- Avoir ou éviter une certaine rotation précise.
- Être en congé une certaine journée du mois.
- Avoir les fins de semaine en congé.
- Commencer à travailler avant ou après un certaine heure.
- Voler en compagnie d'un certain employé.
- etc

2.1.3 Les ressources et le plus court chemin

Une ressource est en fait une variable numérique qui cumule un élément important d'un chemin, qui est un bloc mensuel dans notre cas. Par exemple, pour les problèmes d'Air Transat, six ressources sont utilisées. Parmi les plus simples, une sert à calculer le nombre de *crédits de vols* (nb. de crédits inclus dans [MinimumBlockTime , MaximumBlockTime]); deux autres calculent respectivement le nombre de *jours consécutifs de travail* et les temps de *repos minimum* suite aux tâches; deux ressources vérifient les *patrons de congés*, et une cinquième sert à vérifier les « *CARS* » (Canadian Aviation Rules). Les ressources permettent de valider les chemins et par conséquent de trouver des solutions qui respectent les contraintes locales. Ils permettent en plus de comparer les chemins entre eux durant leur construction pour éliminer ceux qui ne pourront pas être prolongés en chemins optimaux. Toutefois, le défi est d'utiliser le moins de ressources possibles puisque le taux d'accroissement de la complexité de calcul en fonction du nombre de ressources est très élevé. Celui-ci est, à titre de comparaison, nettement supérieur au taux d'accroissement de la complexité en fonction du nombre d'arcs dans le réseau.

La consommation des ressources s'effectue sur les arcs et la validation se fait sur les nœuds à l'aide d'intervalles de faisabilité. Voici un exemple simple de deux chemins et de la consommation d'une seule ressource :

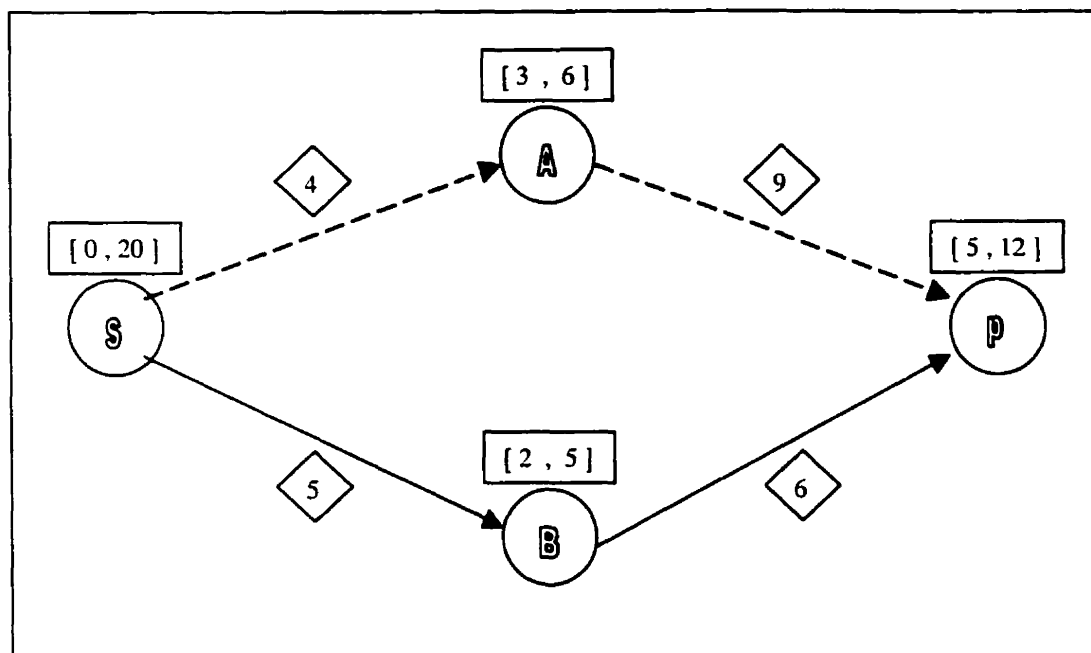


Figure 2.2 – Chemins avec consommation de ressource.

On peut voir ici que le chemin « S-A-P » est invalide puisqu'il y a une consommation de ressource de 13 unités ($4 + 9$) au nœud P et cela ne respecte pas l'intervalle de faisabilité $[5,12]$ du nœud. Par contre, le chemin « S-B-P » est valide puisqu'il l'est en tout nœud du chemin : $\{5 \in [2,5]$ au nœud B} et $\{11 \in [5,12]$ au nœud P}.

Le problème de plus court chemin avec contraintes de ressource a été introduit par Desrochers (1986) et est utilisé dans une multitude de types de problèmes. Plusieurs articles ont d'ailleurs été écrit dans la littérature à ce sujet; entre autres : Desrochers et Soumis (1989), Desrosiers et al. (1995), Desaulniers et al. (1998). Dans ces ouvrages,

on introduit entre autre les notions de dominance et d'étiquette qui sont les principaux outils utilisés par l'algorithme de programmation dynamique. Une étiquette est une structure associée à un chemin pour un nœud particulier. Cette structure contient plusieurs données qui permettent de quantifier un chemin à un nœud donné. Entre autre, une étiquette contient différentes valeurs associées aux consommations de ressources. Puisque plusieurs chemins potentiellement optimaux peuvent passer par un certain nœud, un nœud contient un ensemble d'étiquettes associées aux chemins passant par celui-ci. Cependant, il est très complexe de gérer toutes ces étiquettes. Pour permettre d'améliorer l'efficacité de l'algorithme, la dominance est utilisée sur certaines ressources pour diminuer le nombre d'étiquettes à chaque nœud. Par exemple, si pour un nœud donné, deux chemins a et b ont le même coût mais que la ressource « dominée » du chemin a est de meilleure valeur que celle du chemin b , alors l'étiquette de b sera enlevée du nœud et le chemin b ne sera plus considéré par l'algorithme comme étant un potentiel chemin optimal. Évidemment, la gestion des étiquettes et de la dominance sur plusieurs ressources nécessite de la précaution pour ne pas risquer d'éliminer les chemins optimaux. L'algorithme utilisé est donc relativement complexe et le lecteur est invité à consulter la littérature pertinente pour plus de renseignements.

2.2 Résolution par génération de colonnes (GENCOL)

GENCOL, l'outil d'optimisation du logiciel PBS 500, a été développé au GERAD (Groupe d'Études et de Recherche en Analyse des Décisions, Montréal). Pour trouver le bloc optimal d'un employé, GENCOL résout alternativement (voir figure 2.3, flèche :➔) un problème maître et un sous-problème jusqu'à l'obtention de la solution optimale en nombres réels. Le problème maître est un programme linéaire résolu à l'aide de CPLEX qui permet de trouver un bloc optimal pour un employé courant tout en considérant un sous-ensemble de blocs réalisables et toutes les contraintes globales. Le sous-problème est un problème de plus court chemin résolu à l'aide de l'algorithme de programmation dynamique abordé à la section 2.1.3. Celui-ci reçoit les valeurs des variables duales du problème maître et peut ainsi lui fournir en retour des blocs réalisables qui ont des coûts marginaux négatifs. Ces blocs définissent des colonnes qui sont ajoutées au problème maître pour l'itération suivante. Puisque seuls les blocs de coût marginal négatif permettront potentiellement d'améliorer la solution courante du problème maître, c'est en quelque sorte le sous-problème qui détermine si une solution est optimale ou non pour un problème k donné. À chaque itération (voir figure 2.3, flèche : - - - - ➔), la solution finale en nombres entiers pour l'employé courant est obtenue à l'aide d'un arbre de séparation et d'évaluation progressive. Voici dans les prochaines pages une description détaillée de la formulation du problème maître qui est utilisé pour notre application, de même qu'une figure expliquant le fonctionnement général de GENCOL.

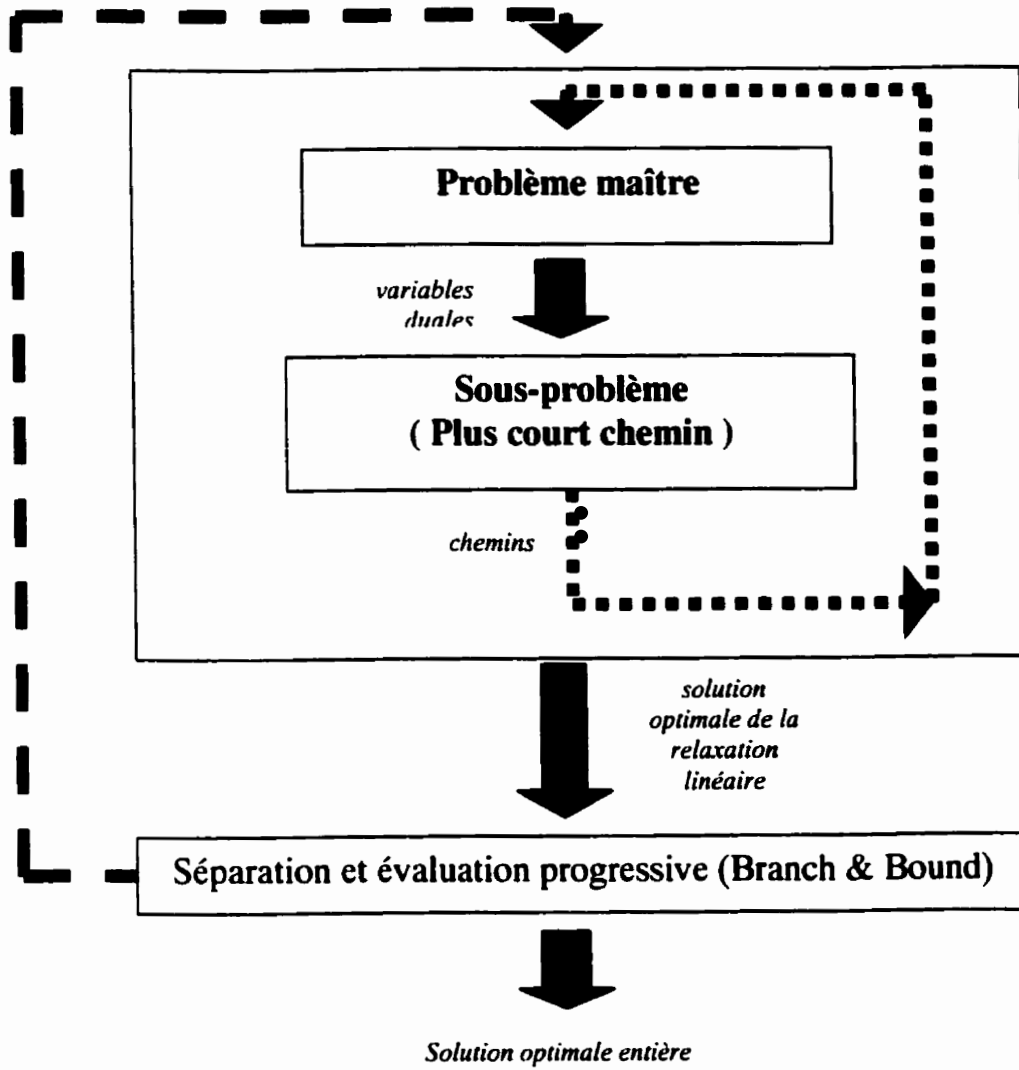


Figure 2.3 - Fonctionnement de Gencol

Le problème maître est un problème restreint aux rotations non assignées et aux employés qui n'ont pas encore été résolus. Un nouveau problème maître de plus en plus petit est donc résolu à mesure que les blocs des employés sont fixés. Il s'exprime comme la relaxation linéaire du problème suivant (supposons ici que les blocs ont été assignés aux $k-1$ employés les plus anciens):

$$(2.1) \quad \text{MAX} \quad \sum_{s \in \Omega^k} C_s Y_s$$

S.C.:

$$(2.2) \quad \sum_{s \in \Omega^k} a_{ps} Y_s = 1, \quad \forall p \in P^k$$

$$(2.3) \quad \sum_{s \in \Omega_e^k} Y_s = 1, \quad e = k, k+1, \dots, m$$

$$(2.4) \quad \sum_{s \in \Omega^k} l_s Y_s \leq w^k$$

$$(2.5) \quad \sum_{s \in \Omega_{m+1}^k} Y_s \leq h$$

$$(2.6) \quad \sum_{s \in \Omega_{m+1}^k} d_s Y_s \leq v$$

$$(2.7) \quad Y_s \in \{0, 1\}, \quad \forall s \in \Omega^k$$

- m est le nombre d'employés et $m+1$ l'indice des blocs ouverts.
- Ω_e^k est l'ensemble des horaires résiduels pour l'employé e , $k \leq e \leq m+1$
- Ω^k est l'ensemble des horaires résiduels pour tous les employés.
- P^k est l'ensemble des rotations (tâches filles) à couvrir.

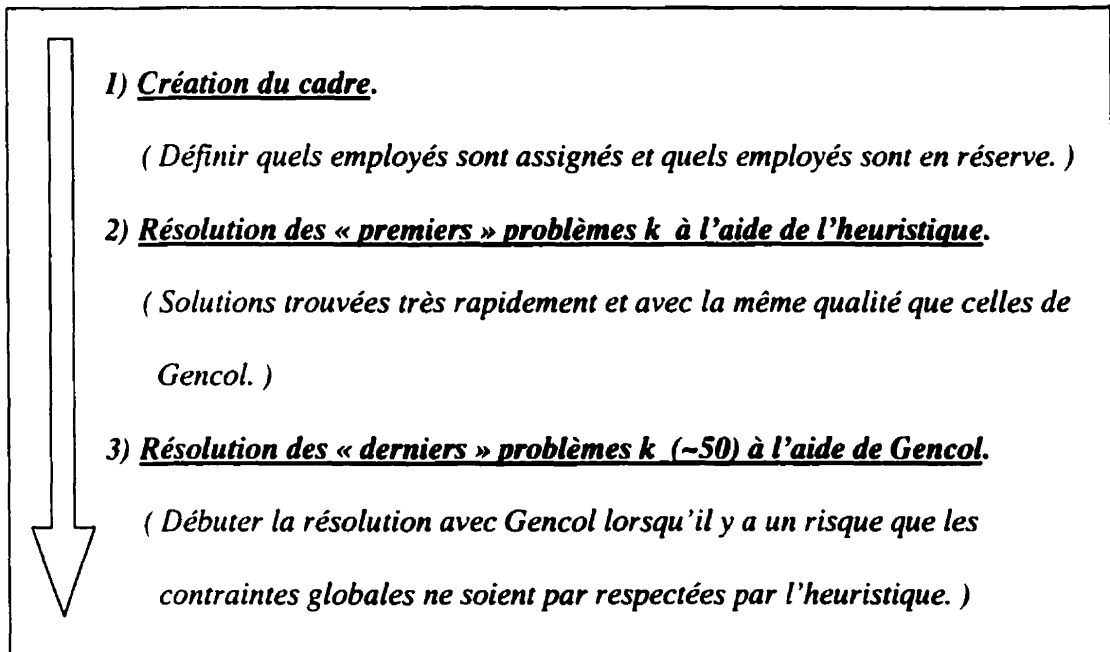
- C_s est le coût associé à un bloc (étant donné l'employé qui est résolu).
- w^k est le nombre de blocs résiduels courts pouvant être construits.
- h est le nombre de blocs ouverts pouvant être construits (open time).
- v est la durée maximale de l'ensemble des blocs ouverts (open time).
- $l_s = 1$ si le bloc est court, 0 sinon.
- $d_s =$ durée d'un bloc s'il est ouvert, 0 sinon.
- $a_{ps} = 1$ si le bloc s couvre la rotation p , 0 sinon.
- Y_s est la *variable* binaire qui est associée à un bloc s . $= 1$ si le bloc s fait partie de la solution, $= 0$ sinon.

Voici des explications sur les contraintes globales 2.2 à 2.6 :

- (2.2) Toutes les rotations de l'ensemble P^k doivent être couvertes une et une seule fois.
- (2.3) Chaque employé doit posséder exactement un bloc.
- (2.4) Il y a un nombre limite (w^k) de blocs courts qui peuvent être créés.
- (2.5) Il y a un nombre limite (h) de blocs qui peuvent être laissés en open time.
- (2.6) Il y a une durée limite (v) que l'ensemble des blocs de l'open time ne peut dépasser.

CHAPITRE 3 : NOUVELLE MÉTHODE DE RÉOLUTION

L'heuristique permet de résoudre rapidement une grande partie de la suite des problèmes k jusqu'à ce que le problème devienne « trop difficile ». Gencol est par la suite utilisé pour résoudre les quelques problèmes k restants. Dans une vue d'ensemble, l'heuristique est donc utilisée de cette façon :



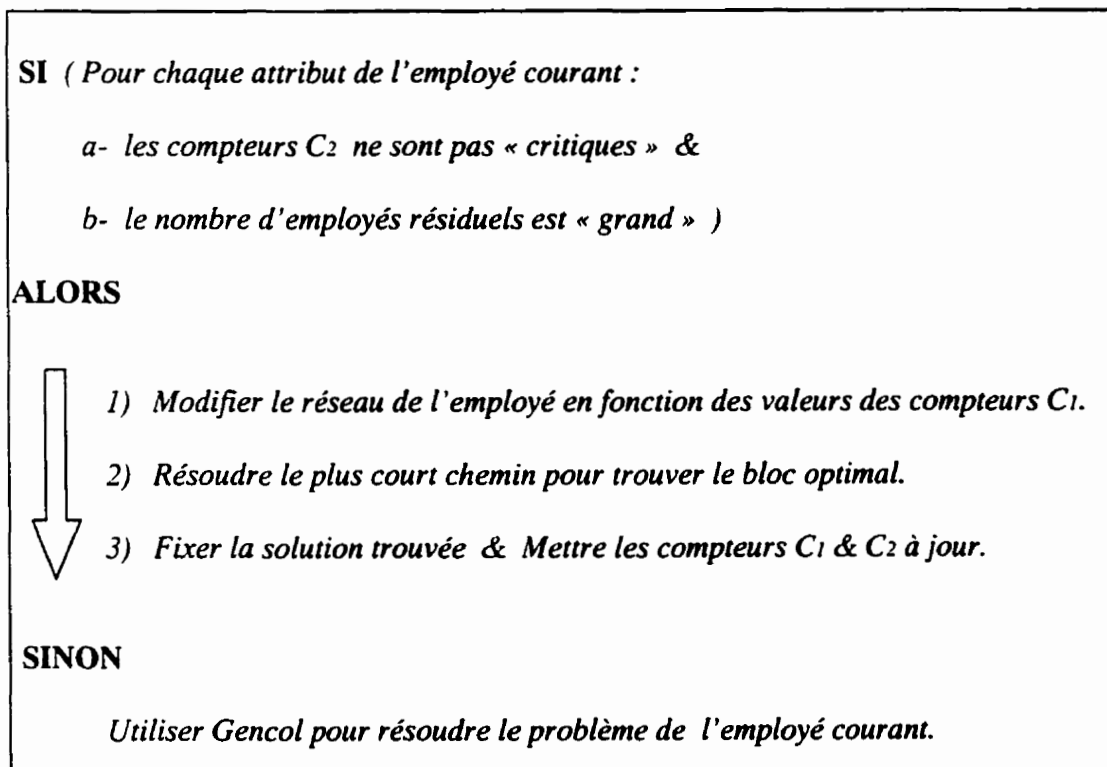
Comme cité précédemment, la raison d'être de l'heuristique est sa grande rapidité d'exécution. Celle-ci s'explique par la manière avec laquelle les contraintes globales sont traitées. En effet, le système PBS 500 optimise le bloc d'un employé courant tout en incluant dans le problème maître l'ensemble des rotations et employés résiduels. Gencol résout donc un sous-problème par employé en effectuant plusieurs

itérations problème maître – sous-problème. Ceci implique alors de résoudre plusieurs fois le problème de plus court chemin pour chacun des employés.

Par contre, l'heuristique résout un seul plus court chemin pour chaque problème k , sans faire appel au problème maître pour le respect des contraintes globales. Pour ce faire, des compteurs dynamiques (*Compteurs C_1*) s'assurent, de façon approximative (mais sans risque de rendre le problème résiduel irréalisable), que pour chaque attribut présent dans le problème, les employés résiduels pourront facilement couvrir l'ensemble des tâches résiduelles. Les données cumulées par ces compteurs C_1 permettent de savoir exactement quelles modifications sont à apporter au réseau d'un employé courant pour s'assurer du respect de ces contraintes globales. Les modifications consistent à enlever certains arcs du réseau pour forcer le plus court chemin à emprunter certains autres arcs qui doivent obligatoirement se trouver dans le bloc de l'employé courant. Ainsi, un seul plus court chemin est nécessaire pour chaque employé, d'où l'énorme diminution des temps d'exécution.

Pour leur part, les *Compteurs C_2* déterminent à quels moments l'utilisation des compteurs C_1 risque de ne plus être satisfaisante pour un attribut donné ou pour l'ensemble du problème résiduel. C'est alors que Gencol est utilisé pour la résolution afin d'assurer avec certitude le respect des contraintes globales.

De façon générale, on peut affirmer que la résolution d'un problème k à l'aide de l'heuristique s'effectue en suivant cette procédure :



Pour chacun des attributs (ex : parler le Grec) présents dans le problème, un couple de compteurs C_1 et C_2 lui est associé. De plus, un attribut « *global* » est créé. Celui-ci considère l'ensemble complet des rotations et des employés résiduels (toute rotation et tout employé possède cet attribut).

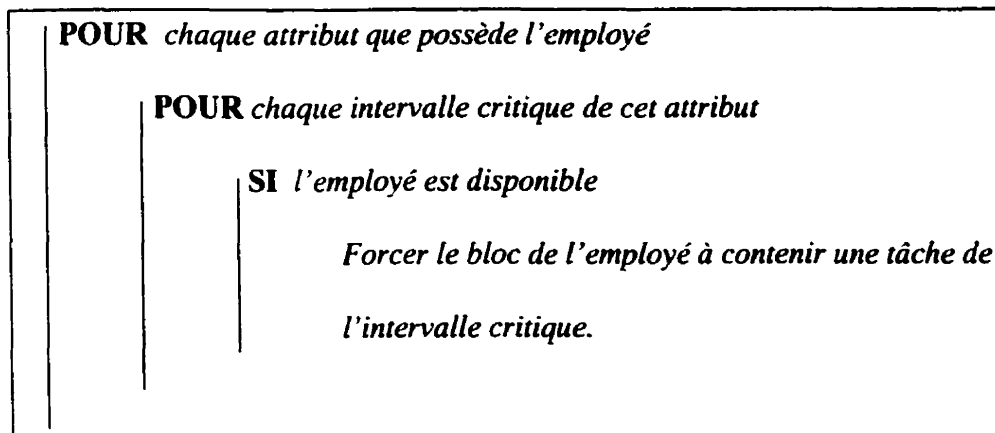
Dans les prochaines sections, une description détaillée du mode de fonctionnement des compteurs est présentée.

3.1 Les compteurs C_1

Les compteurs permettent de calculer l'offre et la demande pour chaque attribut dans un intervalle de temps du mois considéré. Une première notion d'offre, appelée $Offre_1$ est définie ici. Deux autres variantes plus complexes seront introduites plus tard dans ce chapitre. Pour un certain intervalle de temps $[a, b]$ et un attribut X , on définit ces trois termes:




- **Offre**: Nombre d'employés résiduels possédant l'attribut X et qui peuvent effectuer l'une des rotations de l'intervalle $[a, b]$.
- **Demande**: Nombre de rotations résiduelles qui intersectent l'intervalle $[a, b]$ et qui nécessitent un employé possédant l'attribut X .
- **Intervalle critique**: Intervalle où l' $offre_1$ est égal à la demande. (Plusieurs définitions d'offre pourront être utilisées pour définir les intervalles critiques.)

Pour un employé courant, la procédure générale utilisée pour modifier le réseau en fonction des compteurs C_1 est la suivante:



Il faut considérer, pour les notions abordées dans cette section et dans les prochaines pages, que nous traitons les cas où un employé possède au maximum un attribut spécifique (ex : parler Grec) en plus de l'attribut global. Par contre, lors des tests effectués, il y avait des employés qui possédaient trois attributs (ex : global, italien, bilingue) mais au moins un des deux attributs spécifiques était « inactif », i.e : il n'y avait aucune rotation qui nécessitait un agent de bord qualifié de la sorte. Par conséquent, le cas où des employés posséderaient plus d'un attribut spécifique (dont la demande est non nulle) n'est pas traitée par la présente version du code. Une étudiante de maîtrise travaille déjà à implanter une nouvelle heuristique qui permettra de solutionner ce genre de problème. Des explications supplémentaires suivront à ce sujet.

3.1.1 La demande

Lorsque vient le temps de calculer la demande dans chacun des intervalles, on doit redéfinir temporairement les temps de début et de fin des arcs de rotation (sans modifier le réseau). Pour une rotation donnée, on définit son temps de *début* tel qu'il est défini dans le réseau. Pour ce qui est de son temps de *fin*, on y ajoute le temps minimum de repos requis pour cette rotation. On suppose donc que les arcs de rotation du réseau sont plus longs pour ainsi calculer de façon exacte la demande dans les intervalles. Pour les explications qui vont suivre, une rotation sera représentée par un arc  qui est la concaténation de la rotation  et de son temps de repos .

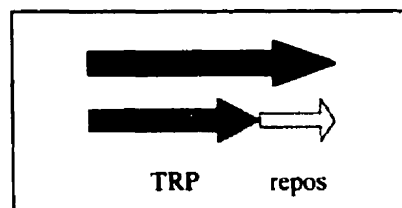


Figure 3.1 – Modification d'un arc de rotation.

La façon de subdiviser le réseau est la plus précise possible: pour un attribut donné, on définit un début d'intervalle (et donc la fin d'un précédent intervalle) dès qu'une fin ou un début de rotation est rencontré. La longueur de ces intervalles est calculée en minutes. Sur la page suivante se trouve une figure qui illustre bien cette méthode.

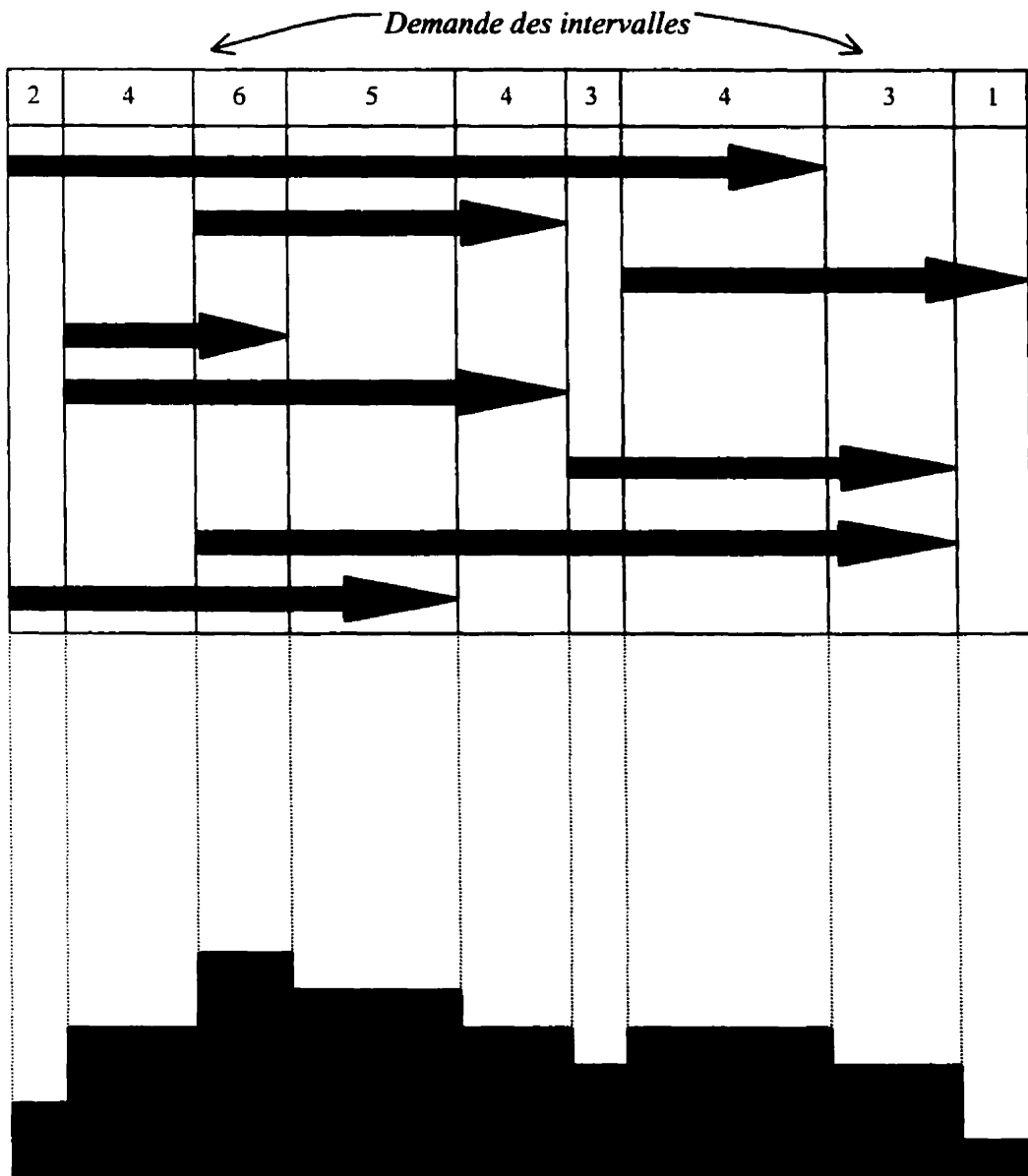


Figure 3.2 – Courbe de la demande du compteur C_i .

Le calcul des demandes initiales (i.e.: au tout début de la résolution) s'effectue simplement en utilisant un algorithme séquentiel qui parcourt tous les intervalles de tous

les attributs. Pour chaque intervalle $[a, b]$ d'un attribut X , on parcourt toutes les rotations ayant l'attribut X et on additionne une unité à la demande de $[a, b]$ si la proposition logique suivante est vraie pour une rotation donnée:

(Début de la rotation $< b$) ET ($a < Fin$ de la rotation)

On dit alors que la rotation « *fait partie de l'intervalle* ». Lorsque le bloc d'un employé a été construit et que la mise à jour des compteurs C_1 s'effectue, la même proposition logique est utilisée pour chacune des rotations du bloc :

Procédure de mise à jour des compteurs C_1 :

POUR (toutes les rotations du bloc de l'employé)

POUR (tout attribut de la rotation)

POUR (tous les intervalles $[a, b]$ de l'attribut)

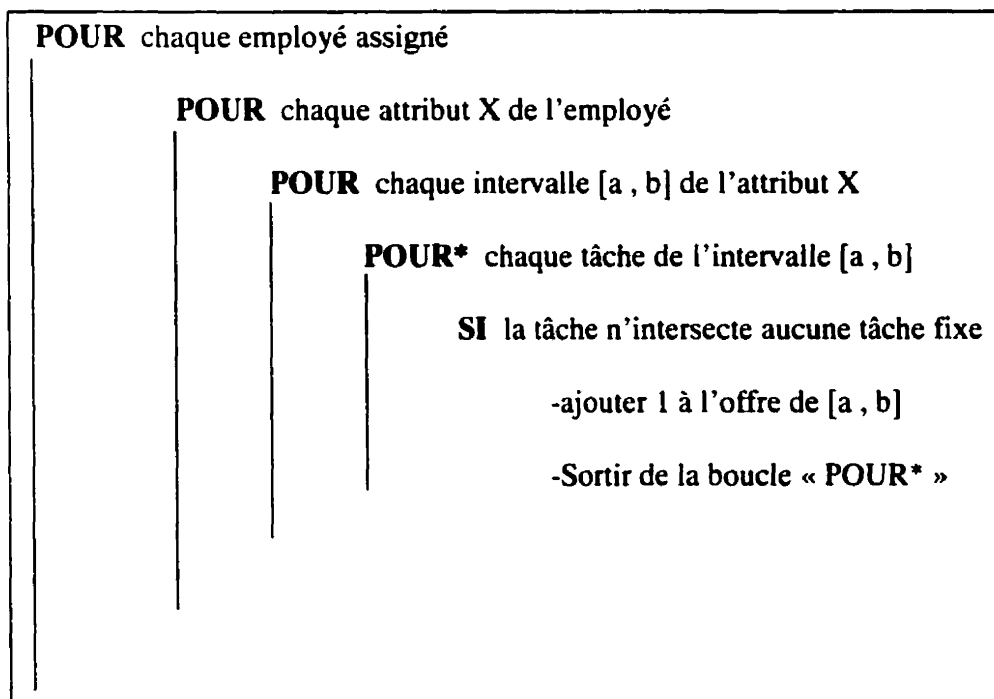
SI ((Début de la rotation $< b$) ET
 ($a < Fin$ de la rotation))

 - Soustraire 1 à la demande de l'intervalle $[a, b]$;

3.1.2 L'offre

Le calcul de l'offre est plus complexe que celui de la demande. Ceci vient du fait que les employés ont des tâches fixes (GND) assignées durant certaines périodes de temps. Un employé peut donc être disponible pour travailler durant un intervalle de temps donné, mais uniquement pour effectuer des rotations qui ne sont pas en conflit avec ses activités fixes. Voici tout d'abord la méthode pour le calcul de l'offre:

Procédure pour le calcul de l'offre :



Lors de la mise à jours des compteurs C_i , suite à la résolution d'un employé, l'offre_i était simplement diminuée d'une unité dans chaque intervalle où l'employé était

disponible. On peut observer que cette méthode n'est pas assez contraignante et pourrait donc mener à la violation des contraintes de recouvrement (2.2). Voici un exemple qui démontre bien la faiblesse de cette méthode :

Supposons qu'il y a 4 employés résiduels, dont 3 qui ont deux activités GND () d'affilée. De plus, il reste 3 rotations (→) qui couvrent la période de temps considérée.

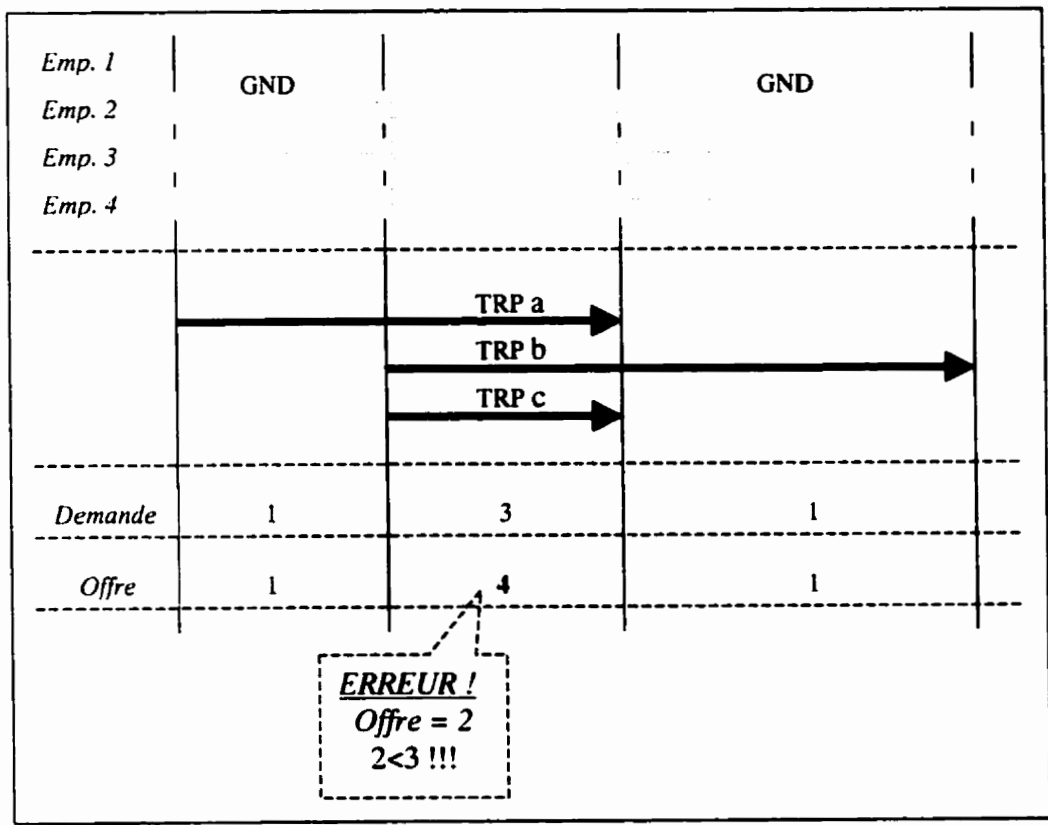


Figure 3.3 – Exemple d'un mauvais calcul avec l'Offre.

La notion d'offre₁ est donc inadéquate pour assurer qu'il y a assez d'employés pour satisfaire la demande. En effet, dans le cas de l'exemple précédent, une affectation maximale pour le deuxième intervalle serait d'affecter la rotation *a* à l'employé 1 et d'affecter la rotation *c* à l'employé 2. Le nombre d'employés pouvant effectuer une rotation est donc égal à 2 pour cet intervalle, ce qui est bien sur insuffisant puisque la demande est de 3. L'offre₁ surévalue le nombre d'employés disponibles et le problème est par conséquent irréalisable dans cet exemple.

Pour obtenir un calcul optimal de l'offre, il faut résoudre pour un intervalle donné, une affectation maximale entre les employés résiduels et les rotations non couvertes. Voici deux nouvelles notions qui permettront de bien définir les calculs d'offre qui seront présentés. Pour un intervalle $[a, b]$ d'un attribut X, on définit:

- **Employés disponibles** : Employés résiduel possédant l'attribut X et pouvant effectuer n'importe qu'elle tâche de l'intervalle $[a,b]$.
- **Employés potentiels** : Employés résiduel possédant l'attribut X et pouvant effectuer une partie seulement (>0 et $<$ demande) des tâches de l'intervalle $[a, b]$.

Soit le calcul optimal de l'offre :

- **OFFRE** = MAXIMUM { *Nombre de tâches de l'intervalle qui peuvent être affectées à des employés résiduels (tout en respectant les contraintes 2.2 à 2.7 du problème maître de Gencol) + Nombre d'employés disponibles à qui des tâches n'ont pas été affectées .* }

De façon plus précise, le calcul d'OFFRE consiste à trouver, parmi un ensemble d'affectations réalisables (i.e. : respectant les contraintes du P.M. de Gencol), celle qui maximisera le nombre d'employés potentiels à qui des tâches ont été affectées.

Évidemment, il faut tenter d'approximer OFFRE puisque ce calcul nécessite l'utilisation d'un outil aussi complexe que Gencol.

Soit une approximation de OFFRE :

- **Offre₂** = *Nombre maximal de tâches qui peuvent être effectuées par les employés potentiels (avec « interruption des tâches ») + employés disponibles .*

Le calcul d'Offre₂ consiste donc à maximiser le nombre de tâches couvertes par les employés potentiels en s'assurant simplement qu'une tâche (ainsi que le temps de repos ajouté) assignée à un employé potentiel n'intersecte pas une des tâches fixes de celui-ci.

En tout temps en cours de résolution, la relation suivante est vérifiée :

employés disponibles ≤ Offre₂ ≤ employés disponibles + employés potentiels

Le calcul de Offre₂ est beaucoup moins complexe puisqu'il n'est plus nécessaire de s'assurer que les tâches de l'intervalle seront effectuées du début à la fin par un même employé potentiel. Voici une proposition qui permettra de bien saisir les différences entre Offre₁, OFFRE et Offre₂ :

Proposition 3.1 : $Offre_1 \geq Offre_2 \geq OFFRE$

Voici une brève justification de ces inéquations, utilisant les notions d'employés potentiels et disponibles :

$Offre_1 \geq Offre_2$: $Offre_1 = \text{employés potentiels} + \text{employés disponibles} \geq Offre_2$.

$Offre_2 \geq OFFRE$: $Offre_2$ est obtenue en résolvant une relaxation du problème qui définit $OFFRE$. En effet, lors du calcul d' $Offre_2$, le respect des contraintes du P.M. de Gencol n'est pas vérifié.

Voici un exemple montrant que $Offre_2$ peut surestimer $OFFRE$. Soit un problème constitué de 2 employés résiduels et soit une portion du réseau :

Intervalle	A	B	C	D	E
Emp. 1	GND				
Emp. 2					GND
Demande à couvrir	TRP a	TRP c			TRP b
Demande	1	2	1	2	1
Offre2	1	2	2	2	1

Figure 3.4 – Preuve que $Offre_2 \geq OFFRE$.

On voit ici que $Offre_2 \geq Demande$ pour tous les intervalles A à E. Si on se fie à $Offre_2$ les contraintes globales semblent respectées alors que ce n'est pas le cas. En effet, il est aisé de voir qu'il nécessiterait au moins un troisième employé résiduel pour couvrir les trois rotations a, b et c. En supposant que ce problème à résoudre est le problème k, l'erreur serait donc survenue à un problème « k - i » ($i > 0$) où $Offre_2$ aurait surestimé OFFRE dans l'intervalle B et D. Ce mauvais calcul aurait donc amené l'heuristique à ne pas considérer les intervalles critiques B et D au problème k - i.

Heureusement, le genre de problème tel que celui de la figure 3.4 ne pourrait arriver qu'avec un problème résiduel « petit ». Hors, comme il sera expliqué à la section 3.3, Gencol est utilisé conjointement avec les compteurs pour résoudre ce genre de situation.

Évidemment, le calcul de $Offre_2$ nécessite un algorithme qui augmenterait considérablement la complexité de l'heuristique. Pour palier à ce problème, un algorithme vorace beaucoup plus rapide est utilisé pour approximer cette valeur. Appelons cette approximation : **Offre₃**. Voici la procédure présentement utilisée par l'heuristique après chaque résolution d'un problème k :

Procédure pour le calcul de l'Offre₂ (pour un intervalle donné) :

- *Classer les employés potentiels en ordre croissant du nombre de tâches qu'ils peuvent couvrir ;*

- *Offre = employés disponibles ;*

POUR *chaque employé potentiel :*

SI *(il existe une tâche non marquée de l'intervalle t.q. l'employé est disponible pour l'effectuer) :*

- *Marquer cette tâche;*

- *Offre = Offre + 1;*

Voici maintenant une proposition qui illustre la relation entre Offre₂ et Offre₃ :

Proposition 3.2 : Offre₂ ≥ Offre₃

Le nombre de tâches couvertes (par les employés potentiels) à l'aide de l'algorithme vorace de Offre₃ est une borne inférieure sur le nombre maximal de tâches couvertes.

Fait intéressant à noter au sujet des tests effectués : lorsqu'il y avait des intervalles critiques, il y avait toujours $Offre_3 = Offre_2$. Comme il a été dit précédemment, dans les cas où l'on utilise l'heuristique, les problèmes sont assez grands et il est probable que $Offre_2 = OFFRE$. Ces différentes évaluations de l'offre devront être testées sur un plus grand nombre de jeux de données.

3.1.3 Modifications du réseau

Maintenant que les conditions de modification du réseau ont été vues en détails, il s'agit maintenant de déterminer de façon exacte les changements à apporter au réseau d'un employé lorsqu'un intervalle est critique.

IMPORTANT : Nous utilisons maintenant la véritable définition d'une rotation (TRP) telle que décrite dans le chapitre 2, i.e., : sans ajouter de repos à la fin du TRP.

Dans les graphiques qui vont suivre, un intervalle critique sera représenté par deux lignes verticales et chaque type d'arc traversant cet intervalle sera analysé.

Arcs de rotation (TRP)

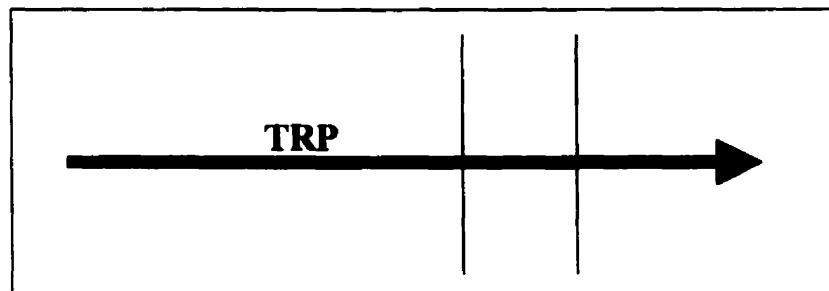


Figure 3.5 – Analyse des compteurs C_i (arcs TRP)

Enlever l'arc s'il ne possède pas le même attribut que l'intervalle critique.

Arcs Link Up (LU)

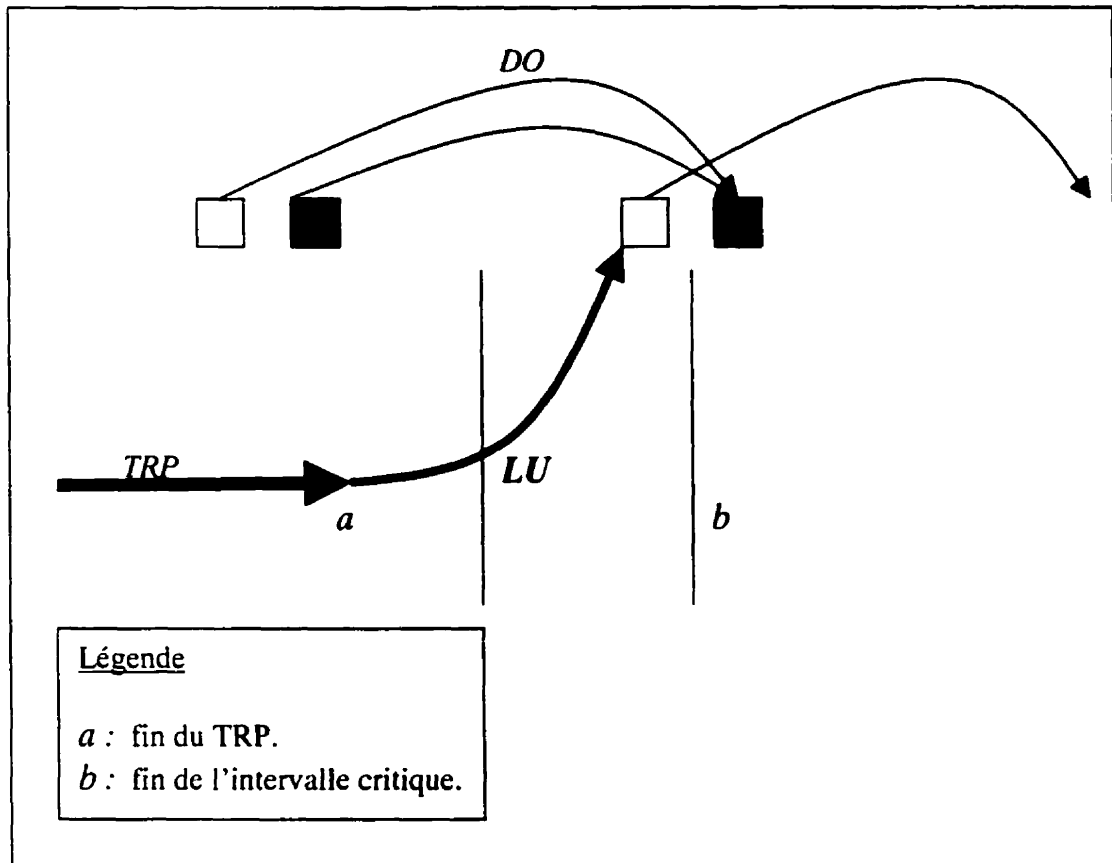


Figure 3.6 – Analyse des compteurs C_1 (arcs LU)

Enlever l'arc LU si TRP ne fait pas partie de l'intervalle:

1) Le TRP associé ne possède pas le même attribut que l'intervalle.

OU

2) $(b - a) > \text{repos minimum du TRP.}$

Arcs Link Down (LD) & Link Next (LN)

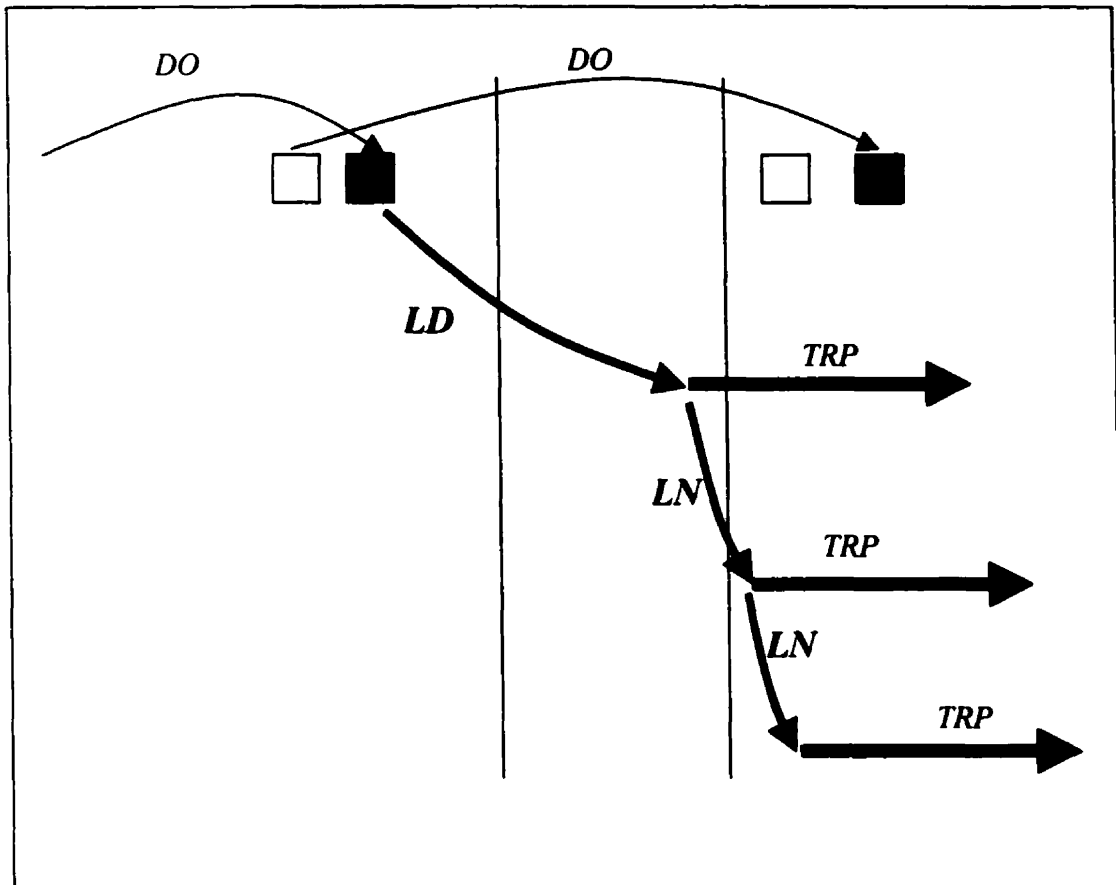


Figure 3.7 – Analyse des compteurs C_1 (Arcs LD & LN)

Enlever tous les arcs LD et LN qui croisent l'intervalle critique puisque cela correspond à un repos de plus de 24 heures (> DO).

(Note : Aucun TRP ne possède de repos de plus de 24 heures.)

Arcs Day Off (DO)

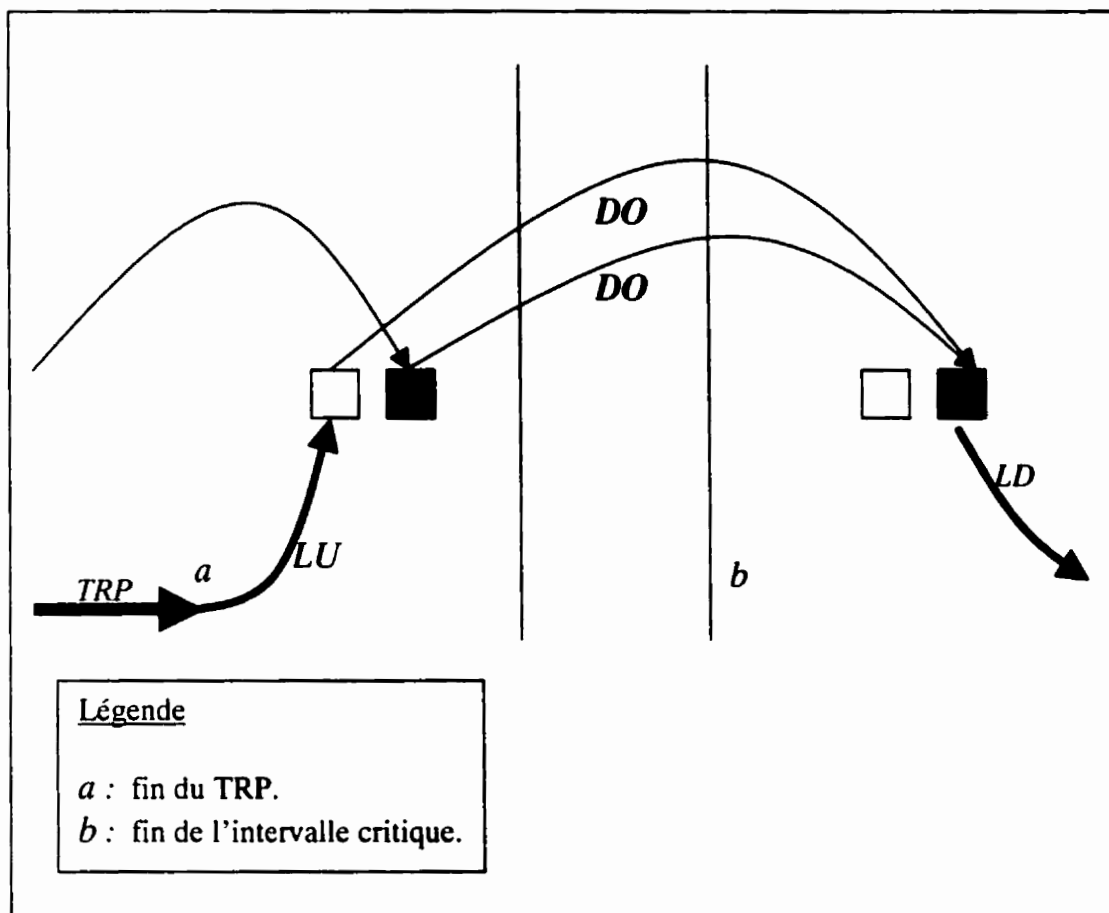




Figure 3.8 – Analyse des compteurs C_1 (Arcs DO)

Enlever les arcs DO  car repos > 24 heures.

Conserver les arcs DO  mais enlever les arcs LU « trop long » (Le TRP associé ne fait pas partie de l'intervalle) entrant au nœud  i.e. : Enlever les LU si $(b - a) > \text{repos minimum du TRP}$.

Arcs Connexion (CN)

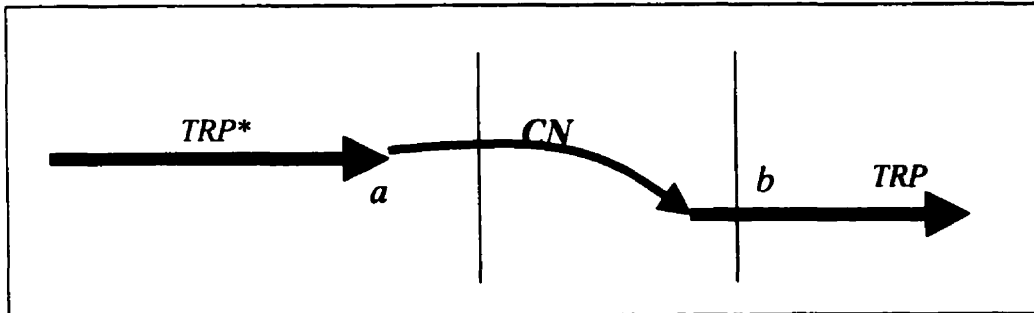


Figure 3.9 – Analyse des compteurs C_1 (Arcs CN).

Enlever l'arc CN si TRP ne fait pas partie de l'intervalle critique:*

1) L'arc TRP ne possède pas le même attribut que l'intervalle critique.*

OU

2) $(b - a) > \text{repos minimum de TRP}$.*

3.2 Les compteurs C₂.

Les contraintes portant sur le nombre de crédits de vols par bloc (nb. de crédits inclus dans [MinimumBlockTime , MaximumBlockTime]) sont beaucoup plus difficiles à vérifier que les contraintes traitées à la section précédente. Gencol devient alors indispensable et les compteurs C₂ sont là pour indiquer à quels moments son utilisation est requise. Pour chaque attribut présent dans le problème, on définit :

- **Demande:** Nombre de crédits de vols requis pour couvrir l'ensemble des rotations résiduelles qui possèdent l'attribut .

$$\sum_{\substack{\text{Rotations} \\ \text{résiduelles}}} [\text{Nb.de crédits de vol de la rotation}]$$

- **Offre Max réelle:** Nombre maximal de crédits de vols que peuvent effectuer les employés résiduels qui possèdent l'attribut.

$$\sum_{\substack{\text{employés} \\ \text{résiduels}}} [\text{MaximumBlockTime} - \text{Nb.de crédits de vol préassignés (tâches fixes)}]$$

- **Offre Min réelle:** Nombre minimal de crédits de vols que peuvent effectuer les employés résiduels qui possèdent l'attribut. (*Note* : Cette valeur est présente uniquement pour l'attribut **Global**)

$$\sum_{\substack{\text{employés} \\ \text{résiduels}}} [\text{MinimumBlockTime} - \text{Nb.de crédits de vol préassignés (tâches fixes)}]$$

- **Offre Max:**

$$\sum_{\substack{\text{employés} \\ \text{résiduels}}} [(0.9 * 85) + ((1 - 0.9) * 68) - \text{Nb.de crédits de vol préassignés}]$$

- **Offre Min (attribut Global):**

$$\sum_{\substack{\text{employés} \\ \text{résiduels}}} [(0.7 * 68) + ((1 - 0.7) * 85) - \text{Nb.de crédits de vol préassignés}]$$

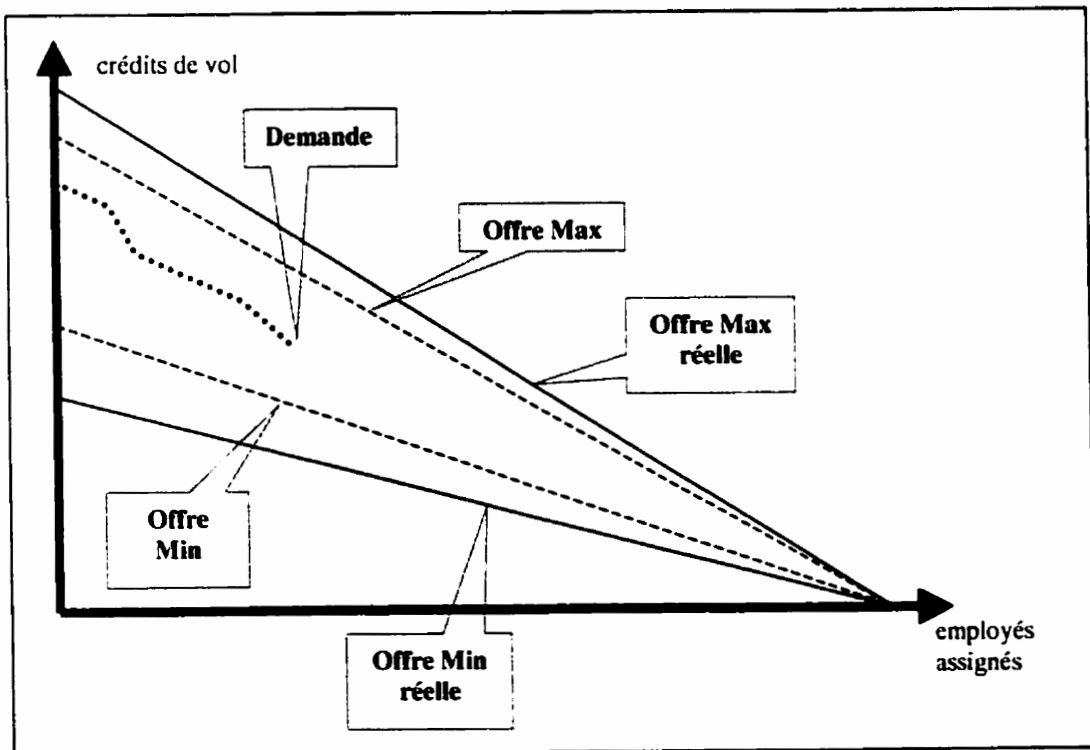


Figure 3.10 – Représentation graphique d'un compteur C2.

La mise à jour des compteurs C2 suite à la résolution d'un employé s'effectue donc selon cette procédure :

Procédure de mise à jour des compteurs C2 :

POUR (chaque attribut que possède l'employé) :

| $Offre\ Max = Offre\ Max - (0.9*85 + ((1-0.9)* 68) - crédits\ préassignés);$

| $Offre\ Min = Offre\ Min - (0.7*68 + ((1-0.7)*85) - crédits\ préassignés);$

POUR (chaque rotation incluse dans le bloc de l'employé) :

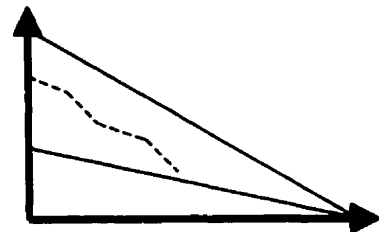
| **POUR** (chaque attribut que possède la rotation) :

| $Demande = Demande - nombre\ de\ crédits\ de\ la\ rotation;$

Grâce à ces valeurs qui sont mises à jour à chaque résolution d'employé, nous pouvons définir deux cas « critiques » où l'utilisation de Gencol sera nécessaire :

1- C2 critique pour l'attribut Global

($Demande \geq Offre\ Max$ ou $Demande \leq Offre\ Min$)



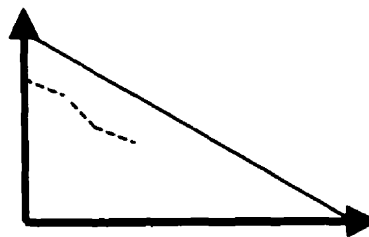
Dans ce cas, la résolution de l'ensemble des employés résiduels devient complexe puisque les employés auront en moyenne des blocs très « chargés » ou encore les employés auront en moyenne des blocs très « petits ». Il y a donc beaucoup moins

de solutions réalisables et même un risque que le problème soit non réalisable si la demande s'approche trop des bornes de l'intervalle [OffreMin , OffreMax]. L'utilisation de Gencol devient alors nécessaire pour terminer le problème. La résolution s'effectue donc selon la méthode standard (PBS 500).

Ce cas, qui est en fait la condition d'arrêt complet de l'heuristique, arrive normalement vers la toute fin de la résolution des employés. Le problème maître et les sous-problèmes (réseaux) étant alors petits, le temps de calcul de Gencol est généralement faible.

2- C2 critique pour un attribut « non-global »

(Demande = Offre Max)



NOTE : Demande \leq Offre Min n'est pas critique puisque par exemple, si l'attribut est : « parler portugais », les employés concernés peuvent effectuer des tâches globales (« non portugaises ») pour combler le reste de leurs blocs !

Pour résoudre le problème de manière efficace dans cette situation, il n'est pas nécessaire d'utiliser Gencol de la même façon que pour le cas précédent. Nous utilisons plutôt une méthode « hybride » qui utilise Gencol et les compteurs C1. La section 3.3 est entièrement consacrée à ce sujet.

3.3 Hybride Gencol-Heuristique

La méthode de résolution présentée dans cette section est utilisée dans deux situations bien précises. Celles-ci concernent toujours la résolution d'un employé qui possède un attribut « non-global » X où :

1- Le compteur C₂ est critique pour cet attribut X.

2- Il reste moins de 10 employés assignés qui possèdent l'attribut X.

Ces deux situations sont problématiques puisque le nombre d'employés et/ou de rotations (qui possèdent l'attribut X) est relativement petit. Ceci a pour effet de diminuer considérablement le nombre de solutions possibles et l'utilisation de Gencol devient alors nécessaire. Par contre, le fait de savoir que le compteur C₂ n'est pas critique pour l'attribut Global permet d'alléger grandement le problème maître de Gencol.

En effet, le problème combinatoire ne concerne en rien les employés et rotations qui ne possèdent pas l'attribut X. L'idée est donc de conserver dans un problème maître très restreint uniquement les rotations et employés qui possèdent l'attribut X. Plus précisément, les contraintes 2.2 de recouvrement des tâches sont actives uniquement pour les rotations qui possèdent l'attribut X et les contraintes 2.3 sont actives uniquement pour les employés qui possèdent l'attribut X.

Il faut par contre remarquer que ce problème restreint aux tâches et aux employés qui possèdent l'attribut X est une trop forte relaxation du problème complet. La contrainte de recouvrement (2.2) de l'ensemble des tâches est complètement relaxée puisque l'ensemble des tâches et employés ne possédant pas l'attribut critique ont été retirés du problème maître. Une amélioration consiste à réduire le réseau de l'employé courant à partir de l'information des compteurs C_i des attributs autres que X. Ainsi, tous les chemins générés par le sous-problème de Gencol seront tels que les employés résiduels ne possédant pas l'attribut X pourront couvrir les tâches ne demandant pas l'attribut X.

Lors des tests effectués, cette méthode hybride s'est avérée très efficace et les solutions trouvées ont été identiques aux solutions trouvées par le Gencol « standard ». Voici des résultats partiels convaincants :

Problème : Air Transat (juin 1998) , employés parlant le Grec :

Tableau 3.1 – Résultats comparatifs (Gencol vs. Hybride)

	<u>Gencol standard</u> temps requis	<u>Méthode hybride</u> temps requis
Employé #1205	976.44 sec.	46.44 sec.
Employé #1204	2790.12 sec.	47.72 sec.
Employé #1968	837.93 sec.	31.08 sec.
Employé #2079	763.47 sec.	3.05 sec.

3.4 Relation entre l'Heuristique et la méthode exacte.

Cette section vise à étudier de façon détaillée les relations qui existent entre les contraintes considérées par l'heuristique et les contraintes du problème traité avec Gencol. Comme il a été dit précédemment, le plus court chemin utilisé par l'heuristique est exactement le même que celui utilisé par Gencol. Rappelons que les contraintes locales sont alors respectées pour l'employé courant. Mais qu'en est-il du respect des contraintes globales ?

Tout d'abord, il est important de noter que les contraintes d'open time ne sont pas traitées par la présente version de l'heuristique. Il en sera d'ailleurs question dans la section 3.5. Supposons donc pour l'instant que ces contraintes ne sont pas considérées.

Redéfinissons les contraintes de façon détaillée, i.e : telles qu'elles auraient été si la résolution avait lieu sans décomposition de Dantzig-Wolfe:

- 1) *Chaque rotation est couverte une fois (2.2 dans le P.M. de Gencol).*
- 2) *Il existe un chemin par employé. (2.3 dans le P.M. de Gencol).*
- 3) *Nombre de crédits de vols légal pour chaque bloc. (S.-P. de Gencol)*
- 4) *Les blocs des employés respectent les « autres » règles des conventions collectives (ex : patrons de congés.) (S.-P. de Gencol)*
- 5) *Contraintes d'intégrité. (Branch & Bound de Gencol)*

Soit un problème contenant n employés assignés. Appelons l'employé courant « employé k » et les employés résiduels: employé $k+1$...employé n . Comme on le sait, lorsque Gencol résout le problème k , il trouve un bloc optimal pour l'employé courant tout en assignant simultanément des blocs légaux (solutions non entières) aux employés résiduels de manière à respecter les contraintes 1 à 4. Il respecte de plus les contraintes d'intégrité pour l'employé k .

L'heuristique utilise les compteurs C_1 pour permettre aussi de construire un bloc optimal pour l'employé k tout en respectant une partie des contraintes 1 à 5. Ceci constitue une certaine relaxation, légèrement différente de celle de Gencol qui trouve des solutions non entières pour les employés résiduels.

Étudions les résultats obtenus en utilisant $Offre_1$, $OFFRE$, ou $Offre_2$ dans la méthode heuristique.

Définissons des prédicats:

PO₁ : $Offre_1$ est supérieure ou égale à la demande dans chaque intervalle pour chaque attribut du problème.

PO : $OFFRE$ est supérieure ou égale à la demande dans chaque intervalle pour chaque attribut du problème.

PO₂ : $Offre_2$ est supérieure ou égale à la demande dans chaque intervalle pour chaque attribut du problème.

Proposition 3.3 : $PO \Leftrightarrow$ contraintes 1,2 et 5 respectées.

Preuve :

(\Rightarrow) S'il existe une affectation dans chaque intervalle, un employé n'effectue jamais 2 tâches en même temps. Donc, la suite de « morceaux » de tâches qu'il effectue constitue un et un seul chemin. De plus, un employé se fait assigner tous les morceaux consécutifs d'une même tâche puisque l'affectation ne permet pas d'interrompre les tâches dans le chemin d'un employé. D'autre part, toutes les tâches sont couvertes par ces chemins car toutes les tâches sont affectées aux employés durant chaque intervalle. Donc la contrainte 1 est satisfaite.

(\Leftarrow) Si les contraintes 1,2 et 5 sont satisfaites, chaque tâche est couverte par un et un seul employé à tout instant. Donc, il existe une affectation des tâches aux employés dans chaque intervalle de temps et chaque tâche est affectée au même employé dans tous les intervalles.

Proposition 3.4 : $(PO \Rightarrow PO_1) \wedge (PO_1 \not\Rightarrow PO)$

Preuve :

Comme il a été cité dans la proposition 3.1 ($Offre_1 \geq OFFRE$) et démontré dans l'exemple de la page 38, $Offre_1$ peut surestimer le nombre de rotations pouvant être couvertes par les employés et rendre par conséquent le problème irréalisable

Proposition 3.5 : (PO \Leftrightarrow PO₁) si on permet la préemption des tâches.

Preuve :

(\Rightarrow) Par la proposition 3.1.

(\Leftarrow) Par définition, chaque employé potentiel ou disponible n'effectue pas de tâche fixe durant l'intervalle. Il suffit alors, pour un intervalle donné, de simplement assigner de façon séquentielle les portions de rotations aux employés. Ainsi, un employé possédant une activité GND de 8h00 à 12h00 pourrait par exemple effectuer les 3 premières heures d'une rotation qui dure de 5h00 à 13h00. Selon cette méthode, chaque portion de rotation est alors assurément couverte par un employé résiduel.

Proposition 3.6 : (PO \Rightarrow PO₂) \wedge (PO₂ $\not\Leftarrow$ PO)

Preuve :

Comme il a été cité dans la proposition 3.2 (Offre₂ \geq OFFRE) et démontré dans l'exemple de la page 41, Offre₂ peut surestimer le nombre de rotations pouvant être couvertes par les employés et rendre par conséquent le problème irréalisable .

Proposition 3.7 : (PO \Leftrightarrow PO₂) s'il n'y a qu'un seul intervalle critique selon Offre₂ (en supposant que les autres intervalles ne sont pas critiques en utilisant OFFRE)

Preuve :

(\Rightarrow) Par la proposition 3.1.

(\Leftarrow) Soit l'intervalle critique X. Selon la définition d'Offre₂, il existe une affectation possible entre les tâches de l'intervalle et les employés potentiels. Assigner les autres tâches de l'intervalle X aux employés disponibles : toutes les tâches de l'intervalle X sont donc couvertes. Pour chaque autre intervalle, assigner des tâches aux employés selon les calculs d'OFFRE. Les contraintes 1,2 et 5 sont donc respectées. Dans le cas où il y a plusieurs intervalles critiques, on peut encore avoir ($PO \Leftrightarrow PO_2$) s'il n'y a pas d'interaction entre chaque paire d'intervalles critiques. L'étude de ces interactions fera l'objet de recherches futures.

Pour ce qui est des compteurs C_2 , ceux-ci nous permettent de savoir s'il ne reste pas trop de crédits de vols à couvrir (demande \geq Offre Max. réelle) et assez de crédits à couvrir pour donner un bloc à chaque employé résiduel qui contiendra un nombre légal de crédits. De plus, si ($\text{Offre Min.} \geq \text{demande} \geq \text{Offre Max.}$) avec $\text{FacteurC}_2\text{Min}$ et $\text{FacteurC}_2\text{Max}$ de l'ordre de 0.1 à 0.3, la moyenne des crédits par blocs est loin des valeurs limites. Dans ce cas, il existe probablement pour chaque employé un bloc avec un nombre légal de crédits.

En dernier lieu, il faut noter qu'il y a une notion relativement importante que les compteurs C_2 ne vérifient que partiellement : les patrons de congé (fait partie des contraintes 4). Ceux-ci déterminent par exemple que les employés doivent avoir 3 séries de congé dans leurs blocs : deux séries de 2 jours consécutifs et une série de 3 jours consécutifs. Toutefois les compteurs C_2 , en assurant que les employés ne travaillent pas

plus que le maximum permis, assurent qu'ils n'auront pas trop de jours de travail. Ils auront donc suffisamment de jours de congés. Il est par contre impossible de savoir si chacun de ces employés pourra avoir un patron « 2-2-3 ». Mais le choix judicieux de *FacteurC2Max* devrait théoriquement éviter tout problème à ce niveau. Lors des tests effectués, ceci n'a d'ailleurs pas été une cause d'erreur puisque la condition d'arrêt de l'heuristique était toujours : $\text{demande} \leq \text{OffreMin}$.

3.5 Open Time

Le lecteur consciencieux aura sans doute remarqué que les compteurs C_1 tels que décrit dans les sections précédentes ne tiennent aucunement compte de l'open time et des employés en réserve. Des recherches à ce sujet sont présentement en cours par une étudiante de maîtrise. Il s'agit en fait d'un problème très semblable à celui des employés qui possèdent plus de deux attributs « spécifiques ». En effet, un bloc d'open time peut être représenté par un employé qui possède tous les attributs du problème.

Pour ce qui est des compteurs C_2 , nous savons déjà qu'elle modification apporter à l'attribut global. Si par exemple le nombre limite de bloc d'open time est fixé à 3, il faut alors ajouter à $OffreMax$ la valeur suivante: $MINIMUM \{ 3 * MaximumBlockTime, MaximumOpenTime \}$. En effet, la demande résiduelle à la suite de la résolution des employés résiduels ne doit pas dépasser $3 * MaximumBlockTime$ puisque les blocs qui seront assignés aux employés en réserve doivent respecter les contraintes locales. Voici une figure qui illustre l'allure qu'aurait le graphique du compteur C_2 global en ajoutant l'open time :

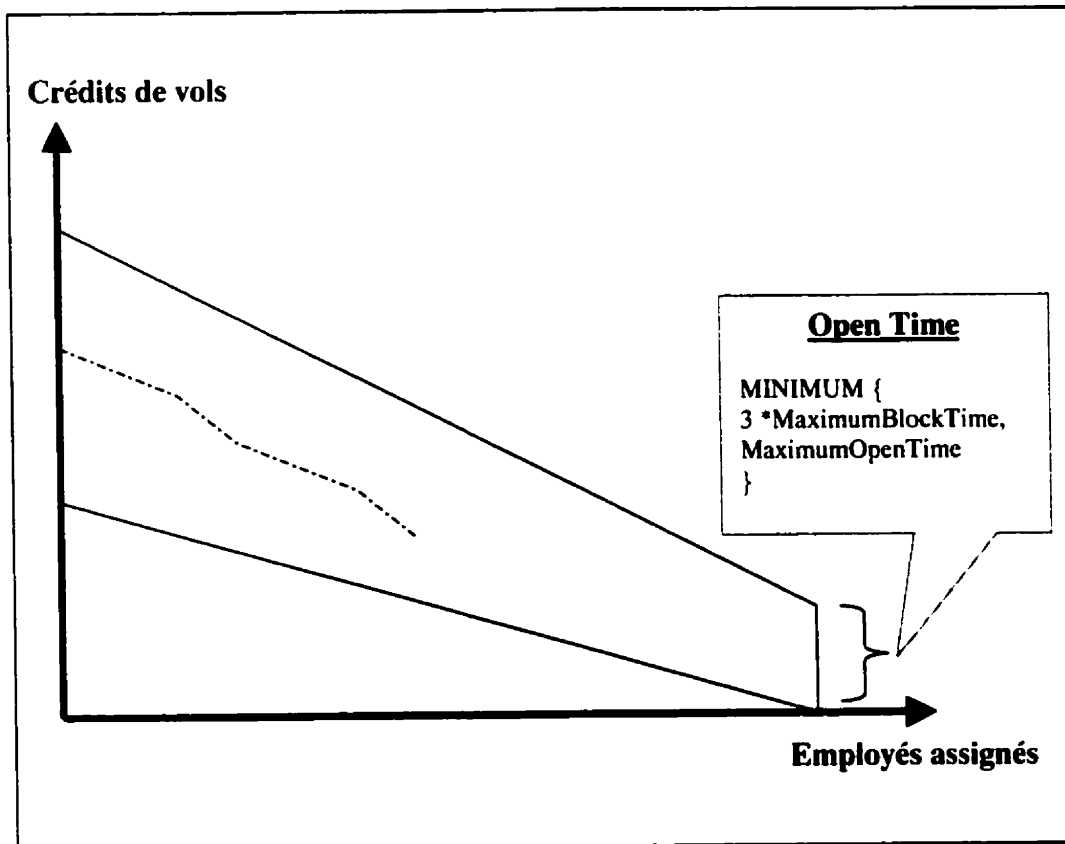


Figure 3.11 – Compteur C₂ global avec Open Time.

CHAPITRE 4 : TESTS.

Comme il a été cité au tout début du mémoire, les tests ont été effectués à l'aide de jeux de données d'Air Transat de 1998. Il s'agit des problèmes d'agents de bords de Juin (Montréal), Juillet (Montréal) et Août (Vancouver).

Pour comparer les résultats obtenus avec ceux de Gencol, nous avons résolu les deux en « parallèle », i.e, à chaque résolution d'employé, Gencol est utilisé en premier pour déterminer la solution optimale et l'heuristique est utilisée par la suite. Il y a trois choses à noter à ce sujet :

- 1- À chaque itération, c'est la solution de l'heuristique qui est fixée.
- 2- Au moment d'utiliser Gencol pour comparer les solutions, toutes les colonnes du problème maître sont enlevées (pour ne pas biaiser les résultats puisque ce sont les solutions de l'heuristique qui sont fixées.)
- 3- Les temps d'exécution de Gencol correspondent à ceux obtenus lorsque la méthode standard est utilisée pour résoudre le problème dans sa totalité, i.e, les solutions sont fixées par Gencol et les temps de résolution sont par conséquent un peu plus courts (les colonnes générées peuvent être conservées du début à la fin).

Ce plan d'expérience permet de vérifier pour chaque problème k si la méthode heuristique donne la même solution que Gencol lorsqu'appliquée exactement dans les mêmes conditions.

4.1 Juin 1998.

Ce problème a été qualifié par les employés d'AD OPT comme ayant été un des problèmes les plus difficiles à résoudre depuis quelques années. La complexité vient du fait qu'une grande partie des rotations ont lieu durant les fins de semaine et que la majorité des employés préfèrent ne pas travailler durant ces périodes. Ceci a pour effet que les compteurs C_1 sont grandement actifs durant les 4 fins de semaine du mois.

Le cadre, calculé par Gencol, contient 222 employés assignés et l'heuristique a continué la résolution jusqu'au 190^{ième}. En effet, ce n'est qu'au 191^{ième} employé que le compteur C_2 global a été actif (demande = OffreMin, Facteur C_2 Min = 0.9). Toutefois, vu la complexité du problème, celui-ci est devenu irréalisable au 191^{ième} (alors qu'il était réalisable au 190^{ième}). D'ailleurs, l'utilisation de la méthode standard du début à la fin s'est aussi arrêtée au 190^{ième} employé puisqu'aucune solution entière n'a été trouvée au 191^{ième} employé. Ceci est dû au fait que lors de la résolution d'un problème k , la méthode standard n'exige pas l'intégrité pour les employés $k+1 \dots n$. Le problème peut s'avérer non réalisable quand on résout les employés suivants et que l'on exige l'intégrité pour l'employé courant.

Il est important de noter que la fonction de plus court chemin possède un paramètre « SIZE » qui représente une borne sur la quantité de mémoire vive que peut utiliser la fonction. Ceci veut donc dire que plus cette valeur est élevée, plus les solutions trouvées seront exactes et plus le temps d'exécution sera grand. Pour les

résultats qui suivent, ce paramètre a été fixé à **300 megs**. Un deuxième test suivra en utilisant cette fois 500 megs.

Temps total requis par l'heuristique (jusqu'au 190^{ième}) : 1h 25

Temps total requis par Gencol (jusqu'au 190^{ième}) : 10 h 39

6 cas où : solution (heuristique) ≠ solution (Gencol) :

Employés # 129, 157, 169 : solutions (Gencol) meilleures que l'heuristique
(taux d'erreur < 2%)
(voir rangées en **caractères gras** du tableau)

Employés # 86, 163, 180 : solutions (heuristique) meilleures que Gencol.
(voir rangées en **caractères gras** du tableau)

Paramètres : FacteurC2Min = 0.9
FacteurC2Max = 0.9

Attributs : Bilingue : bil.
« Unianglais » : uni.
Portugais : por.
Grec : gre.
Italien : ita.
Allemand : all.
Espagnol : esp.
Polonais : pol.

Tableau 4.1 – Test (juin 1998)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution</u> <u>(Gencol)</u>	<u>Solution</u> <u>(Heuristique)</u>	<u>Temps (sec.)</u> <u>(Gencol)</u>	<u>Temps (sec.)</u> <u>(Heuristique)</u>	<u>Nombre</u> <u>d'interv.</u> <u>Ci actifs.</u>
1	140	bil.	-79382	-79382	473.58	22.47	0
2	600	bil.	-24442	-24442	627.22	25.13	0
3	737	bil.	-23178	-23178	261.29	45.98	0
4	743	bil.	-64079	-64079	156.30	26.06	0
5	735	bil.	-5159	-5159	304.08	26.75	0

Tableau 4.1 – Test (juin 1998)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
6	738	bil.	-15240	-15240	619.40	29.67	0
7	794	bil.	-146490	-146490	305.07	23.22	0
8	871	bil.	-8663	-8663	141.81	20.52	0
9	894	bil.	-190818	-190818	264.83	23.80	0
10	903	bil.	-509	-509	422.77	20.83	0
11	862	bil.	-154912	-154912	146.05	24.52	0
12	881	bil.	-55743	-55743	255.12	27.49	0
13	899	por.bil.	-18720	-18720	342.96	35.12	0
14	895	bil.	-142056	-142056	347.01	21.01	0
15	876	bil.	-99876	-99876	127.49	25.90	0
16	1113	bil.	-4015	-4015	274.04	34.05	0
17	1109	bil.	-26396	-26396	274.07	28.72	0
18	1088	bil.	-63883	-63883	259.60	19.28	0
19	1084	bil.	83744	83744	210.34	28.55	0
20	1111	bil.	-57226	-57226	565.70	27.08	0
21	1083	bil.	-3213	-3213	172.18	25.74	0
22	1092	por.bil.	-10900	-10900	150.12	29.46	0
23	1115	bil.	-94368	-94368	231.77	27.18	0
24	1116	bil.	-26456	-26456	190.51	22.12	0
25	1129	bil.	-226887	-226887	325.76	20.12	0
26	1123	bil.	-19463	-19463	237.19	25.82	0
27	1128	bil.	-14149	-14149	160.50	22.97	0
28	1124	bil.	-10729	-10729	155.71	30.15	0
29	1119	bil.	-25754	-25754	179.57	17.23	0
30	1141	bil.	-49507	-49507	403.19	36.52	0
31	1171	bil.	16294	16294	157.88	12.30	0
32	1158	bil.	-28042	-28042	199.75	35.53	0
33	1170	bil.	-18380	-18380	89.50	22.25	0
34	1147	bil.	-193036	-193036	159.53	20.58	0
35	1167	bil.	-3139	-3139	107.68	23.81	0
36	1143	esp.por.bil.	-10163	-10163	94.06	115.99	0
37	1166	bil.	-120112	-120112	224.68	22.85	0
38	1148	bil.	-11079	-11079	267.33	24.66	0
39	1156	bil.	-55130	-55130	194.78	24.81	0
40	1198	bil.	-8493	-8493	187.34	34.96	0
41	1201	all.bil.	-10676	-10676	208.35	59.50	0
42	1188	bil.	-134626	-134626	214.93	32.82	0
43	1195	uni.	-12436	-12436	131.43	5.74	0
44	1209	bil.	-9356	-9356	217.16	23.04	0
45	1205	gre.bil.	-168516	-168516	62.26	25.54	0
46	1211	uni.	-22957	-22957	245.26	28.35	0
47	1213	bil.	-31633	-31633	235.35	12.78	0
48	1192	uni.	-16958	-16958	139.03	2.00	0
49	1199	uni.	-13383	-13383	164.98	20.62	0
50	1204	gre.uni.	-9160	-9160	1487.88	39.66	2

Tableau 4.1 – Test (juin 1998)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
51	1404	bil.	-2668	-2668	55.44	19.61	0
52	1410	bil.	-12776	-12776	220.82	16.74	0
53	1403	bil.	-34857	-34857	172.12	20.34	0
54	1398	bil.	-11364	-11364	119.08	31.09	0
55	1393	bil.	-85868	-85868	210.35	20.04	0
56	1416	bil.	-12582	-12582	62.90	3.39	0
57	1430	bil.	-8748	-8748	135.12	32.62	0
58	1432	bil.	-8399	-8399	108.69	33.20	0
59	1163	bil.	-56406	-56406	164.95	35.25	0
60	1435	por.bil.	-56906	-56906	122.23	100.56	0
61	1422	uni.	812.53	812.53	90.44	23.98	0
62	1433	bil.	29037	29037	225.44	26.89	0
63	1428	uni.	-3551	-3551	201.53	34.63	0
64	1423	bil.	-24726	-24726	183.75	36.62	0
65	1444	bil.	-29329	-29329	126.00	15.73	0
66	1426	bil.	-56889	-56889	174.68	41.32	0
67	1434	uni.	-7761	-7761	162.68	18.16	0
68	1483	bil.	-20699	-20699	464.61	26.18	0
69	1509	bil.	-15375	-15375	45.65	29.17	1
70	1500	bil.	-119292	-119292	207.60	13.37	1
71	1448	bil.	-4300	-4300	546.13	15.83	1
72	1502	uni.	-3832	-3832	115.27	15.06	1
73	1481	por.bil.	-1418	-1418	33.87	78.93	1
74	1504	bil.	-27179	-27179	64.46	20.69	1
75	1499	bil.	-8861	-8861	78.14	21.88	1
76	1533	bil.	-44876	-44876	188.51	30.76	1
77	1545	bil.	-35203	-35203	181.47	12.58	1
78	1541	bil.	-40133	-40133	90.35	16.81	1
79	1543	uni.	-40389	-40389	109.97	23.61	1
80	1547	bil.	-103475	-103475	55.28	38.41	1
81	1548	uni.	-30692	-30692	185.44	15.38	1
82	1567	bil.	-19550	-19550	53.49	21.97	1
83	1570	bil.	-5331	-5331	259.62	14.54	1
84	1505	bil.	-19428	-19428	243.90	29.46	1
85	1600	bil.	35248	35248	212.93	22.08	1
86	1579	bil.	-21226	-21232	302.64	19.06	1
87	1615	por.bil.	-32103	-32103	58.07	79.84	1
88	1591	bil.	-9592	-9592	77.66	24.88	1
89	1666	bil.	-12308	-12308	462.69	31.35	1
90	1596	bil.	-13660	-13660	96.68	10.45	0
91	1576	bil.	-20045	-20045	137.26	26.71	1
92	1637	bil.	-4129	-4129	92.79	23.58	1
93	1606	bil.	-17382	-17382	64.08	21.21	1
94	1667	bil.	-52868	-52868	27.70	20.38	1
95	1635	bil.	-20594	-20594	139.07	17.55	1

Emp.#	Attribut(s)	Solution (Gencol)	Solution (Heurtsique)	Temps (sec.) (Gencol)	Temps (sec.) (Heurtsique)	Nombre d'interv. Clactis.
96	por.bil.	-43770	-43770	349.99	172.75	1
97	bil.	-40385	-40385	88.59	12.84	1
98	bil.	-21459	-21459	84.92	30.22	1
99	ta.bil.	-31293	-31293	75.18	17.95	1
100	bil.	-59955	-59955	197.08	10.20	1
101	bil.	-8547	-8547	133.89	17.30	1
102	bil.	-14796	-14796	77.91	18.87	1
103	bil.	35538	35538	123.79	25.90	1
104	bil.	-184335	-184335	391.29	18.41	1
105	unt.	-128532	-128532	81.24	24.71	1
106	bil.	-18637	-18637	176.70	23.29	1
107	bil.	-6569	-6569	37.72	16.86	1
108	bil.	-1676	-1676	351.32	16.30	1
109	bil.	-1434	-1434	133.18	26.73	1
110	unt.	-4670	-4670	327.61	29.48	1
111	unt.	-14116	-14116	130.46	22.48	1
112	unt.	-38731	-38731	185.54	12.89	1
113	unt.	-6342	-6342	56.06	9.72	1
114	unt.	-7442	-7442	129.75	16.02	1
115	bil.	-19229	-19229	60.29	15.65	1
116	esp.bil.	-21608	-21608	182.65	20.89	1
117	bil.	40645	40645	239.29	21.76	1
118	pol.unt.	-9768	-9768	82.29	15.99	1
119	unt.	-7656	-7656	83.35	14.42	1
120	bil.	-145759	-145759	49.08	20.68	1
121	bil.	-38493	-38493	314.13	11.80	1
122	bil.	-65271	-65271	96.94	17.97	1
123	unt.	-111873	-111873	204.46	12.35	1
124	gpc.unt.	-38984	-38984	367.52	33.92	3
125	bil.	-68634	-68634	228.78	15.04	1
126	bil.	-25318	-25318	31.43	12.16	1
127	unt.	-64518	-64518	17.06	21.93	2
128	unt.	-8872	-8872	23.94	15.67	2
129	ta.por.unt.	-10939	-10884	249.02	153.10	5
130	unt.	-3726	-3726	78.28	12.85	2
131	bil.	-3840	-3840	51.54	14.23	3
132	unt.	-5577	-5577	46.42	19.69	3
133	bil.	-78642	-78642	90.08	22.36	3
134	unt.	-8317	-8317	24.18	11.22	3
135	esp.unt.	-60836	-60836	124.33	18.87	3
136	bil.	-12681	-12681	122.58	23.07	3
137	gpc.bil.	-6525	-6525	303.29	2.05	4
138	unt.	-9011	-9011	100.01	9.94	3
139	bil.	-15917	-15917	113.61	15.60	3
140	unt.	-52660	-52660	112.67	23.43	3

Tableau 4.1 - Test (juin 1998)

Emp.#	Attribut(s)	Solution (Gencol)	Solution (Heuristique)	Temps (sec.) (Gencol)	Temps (sec.) (Heuristique)	Nombre d'interv. Cl actifs
141	uni.	-85425	-85425	199.72	9.06	3
142	bil.	-13131	-13131	120.99	23.17	6
143	uni.	42655	42655	195.21	17.58	6
144	bil.	-8275	-8275	98.13	9.06	7
145	bil.	-33759	-33759	35.89	30.34	7
146	bil.	-24168	-24168	73.66	14.02	7
147	bil.	-39019	-39019	285.84	17.05	7
148	bil.	-98168	-98168	31.80	17.79	8
149	uni.	-14323	-14323	107.48	9.71	8
150	lia.bil.	-1281	-1281	154.10	11.60	8
151	bil.	-30387	-30387	61.02	11.81	8
152	bil.	-16373	-16373	20.90	12.96	8
153	bil.	-11364	-11364	114.05	14.12	8
154	bil.	-15993	-15993	108.79	16.42	8
155	bil.	-2872	-2872	56.81	14.92	8
156	bil.	-22108	-22108	26.90	11.10	8
157	bil.	5809	5977	26.13	17.23	8
158	bil.	-23462	-23462	23.24	16.09	8
159	bil.	-56831	-56831	31.98	16.77	8
160	bil.	-79270	-79270	21.15	13.33	8
161	esp.bil.	-5241	-5241	27.44	10.81	8
162	bil.	-54827	-54827	161.73	11.96	11
163	bil.	-21067	-21075	13.75	8.14	11
164	bil.	-71128	-71128	75.55	18.01	11
165	bil.	-15960	-15960	107.49	17.84	11
166	bil.	-45006	-45006	81.44	17.64	11
167	pol.bil.	-55083	-55083	11.55	13.66	11
168	bil.	-20626	-20626	27.45	16.39	11
169	bil.	-62904	-62858	358.41	15.92	17
170	bil.	-19489	-19489	116.57	15.76	17
171	bil.	-75949	-75949	35.66	8.97	17
172	bil.	-67830	-67830	158.35	22.95	20
173	bil.	-40820	-40820	192.66	12.71	22
174	bil.	-8167	-8167	24.01	16.02	27
175	bil.	-42175	-42175	182.06	9.37	29
176	bil.	-30113	-30113	17.06	20.95	29
177	bil.	-30113	-30113	27.56	14.59	29
178	bil.	-23795	-23795	128.49	16.67	29
179	bil.	-3027	-3027	90.12	14.01	27
180	bil.	-66838	-66871	116.07	14.76	29
181	por.bil.	-5899	-5899	285.84	30.97	10
182	bil.	-93573	-93573	247.71	11.29	31
183	bil.	-16660	-16660	442.17	13.19	31
184	bil.	-38397	-38397	797.13	10.87	31
185	bil.	-7026	-7026	22.47	9.59	31

Tableau 4.1 - Test (juin 1998)

Tableau 4.1 – Test (juin 1998)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
186	2894	bil.	-59382	-59382	1025.58	10.06	31
187	2868	bil.	-2146	-2146	898.12	9.76	31
188	2885	ita.bil.	-11527	-11527	1418.83	7.46	31
189	2873	bil.	-27405	-27405	814.04	17.64	31
190	2876	bil.	-24784	-24784	805.81	9.43	31

Lors des tests effectués en utilisant 500 megs de mémoire vive pour le plus court chemin, il n'y a eu aucun cas où une solution de Gencol était meilleure que celle de l'heuristique. Cependant, le temps d'exécution a été plus élevé : 2h15. Ces petites erreurs sont dues au fait que l'algorithme de plus court chemin n'est pas complètement optimal. En effet, différentes techniques heuristiques sont utilisées par l'algorithme afin d'accélérer les temps d'exécution et de diminuer la demande en mémoire vive.

Voici maintenant un graphique montrant l'évolution du compteur C2 global tout au long de la résolution. Ce graphique est quelque peu différent des graphiques qui ont été montrés dans la section 3.2. Pour améliorer la lisibilité du graphique, celui-ci a été « normalisé » en soustrayant, à chaque résolution d'employé, une même valeur à OffreMax, OffreMin et Demande. Cette valeur, appelée « DemandeThéorique » correspond à la Demande initiale à laquelle on soustrait à chaque résolution le nombre de crédits d'un bloc « moyen », i.e., bloc possédant $[(0.5 * \text{MaximumBlockTime}) + (0.5 * \text{MinimumBlockTime}) - (\text{nombre de crédits pré-assignés})]$ crédits.

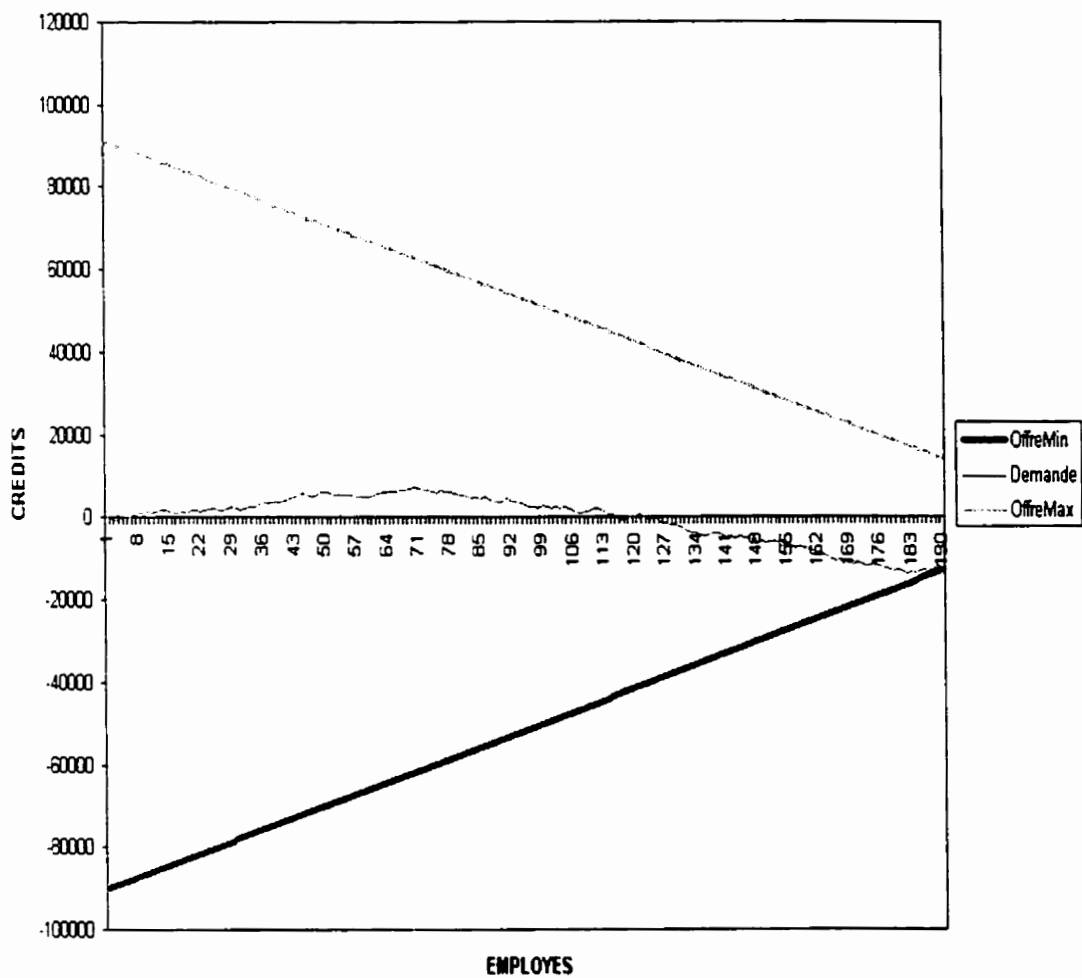


Figure 4.2 – Compteur C₂ Global (Juin 1998, 1^{er} au 190^{ième} employé)

4.2 Juillet 1998.

Ce problème a été aussi considéré comme étant ardu à résoudre, mais pour des raisons différentes du problème précédent.

Le cadre, calculé par Gencol, contient 271 employés assignés. Cette fois-ci, les compteurs C_1 n'ont aucunement été critiques pour l'attribut global (et 2 fois seulement pour l'attribut « Grec ») mais le compteur C_2 global l'est devenu très rapidement en cours de résolution. En effet, les premiers employés à être résolus ont reçu des blocs contenant beaucoup de crédits de vol et dès le 184^{ième} employé, le compteur C_2 global s'est alors trouvé dans la situation critique : $\text{Demande} \leq \text{OffreMin}$. Il n'y a cependant pas à s'en alarmer puisque la méthode standard n'a pas fait beaucoup mieux : celle-ci s'est arrêtée au 194^{ième} employé par la suite. L'heuristique n'est donc pas responsable de cette difficulté. La cause principale est que le cadre initial contenait beaucoup trop d'employés pour un problème où les employés voulaient des blocs avec beaucoup de crédits de vol. De plus, il n'y a aucune flexibilité en ne permettant aucun bloc d'open time car les blocs des employés assignés doivent contenir le nombre minimum de crédits de vols.

Pour palier à ce problème, un nouveau cadre a été calculé suite à la résolution du 194^{ième} employé. Effectivement, Gencol a pu aboutir à une bien meilleure solution en mettant 261 employés assignés dans le cadre au lieu de 271.

Note : le paramètre « SIZE » a été fixé à **250 megs** dans les tests. (*les 183 solutions trouvées avec l'heuristique sont identiques à celles de Gencol*)

Temps total requis par l'heuristique (jusqu'au 183) : 1h 05

Temps total requis par Gencol (jusqu'au 183) : 15 h 10

Paramètres : FacteurC2Min = 0.9

FacteurC2Max = 0.9

Attributs : Bilingue : bil.
 « Unianglais » : uni.
 Portugais : por.
 Grec : gre.
 Italien : ita.
 Allemand : all.
 Espagnol : esp.
 Polonais : pol.

Tableau 4.2 – Test (juillet 1998 , 183 premiers employés)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
1	140	bil.	-174294	-174294	853.63	27.43	0
2	600	bil.	-138705	-138705	637.20	26.49	0
3	737	bil.	-211674	-211674	651.90	23.63	0
4	743	bil.	-222899	-222899	476.72	36.12	0
5	735	bil.	-46124	-46124	956.22	23.08	0
6	738	bil.	-75797	-75797	788.57	17.89	0
7	794	bil.	84666	84666	339.01	13.87	0
8	871	bil.	-71803	-71803	502.27	17.48	0
9	894	bil.	-11627	-11627	483.31	17.11	0
10	903	bil.	-184582	-184582	403.32	22.28	0
11	862	bil.	-257317	-257317	353.76	16.67	0
12	881	bil.	-51178	-51178	339.06	25.40	0
13	899	por.bil.	-36695	-36695	278.94	41.52	0
14	895	bil.	-178944	-178944	854.65	23.04	0
15	876	bil.	-81277	-81277	356.75	20.48	0
16	1113	bil.	-3871	-3871	325.72	5.85	0

Tableau 4.2 – Test (juillet 1998 , 183 premiers employés)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
17	1109	bil.	-26138	-26138	286.97	19.68	0
18	1088	bil.	-246166	-246166	191.47	14.77	0
19	1084	bil.	-10956	-10956	565.39	24.07	0
20	1111	bil.	-262140	-262140	475.56	9.36	0
21	1083	bil.	-9116	-9116	325.67	22.93	0
22	1092	por.bil.	-26073	-26073	310.64	34.61	0
23	1115	bil.	-73395	-73395	517.15	26.94	0
24	1116	bil.	-44761	-44761	235.09	21.26	0
25	1129	bil.	-254953	-254953	316.69	31.41	0
26	1123	bil.	-56270	-56270	176.06	33.81	0
27	1128	bil.	-99348	-99348	474.01	35.61	0
28	1124	bil.	-103300	-103300	208.85	25.88	0
29	1119	bil.	-103900	-103900	296.29	13.67	0
30	1141	bil.	-62657	-62657	279.11	13.79	0
31	1171	bil.	-44745	-44745	410.88	17.13	0
32	1158	bil.	-206541	-206541	378.90	22.62	0
33	1163	bil.	-152416	-152416	449.47	19.71	0
34	1170	bil.	-54778	-54778	245.04	19.97	0
35	1147	bil.	-74816	-74816	232.91	21.40	0
36	1167	bil.	-14943	-14943	90.92	24.26	0
37	1143	esp.por.bil.	-79538	-79538	371.34	20.84	0
38	1166	bil.	-248548	-248548	165.03	3.44	0
39	1156	bil.	-65254	-65254	374.97	3.78	0
40	1198	bil.	-14861	-14861	425.92	42.69	0
41	1201	all.bil.	-15212	-15212	220.67	105.88	0
42	1188	bil.	-254319	-254319	482.52	16.41	0
43	1193	uni.	-27868	-27868	246.58	23.83	0
44	1209	bil.	-57224	-57224	753.16	25.92	0
45	1205	gre.bil.	-14404	-14404	484.54	105.10	0
46	1211	uni.	-22658	-22658	92.64	19.07	0
47	1213	bil.	104681	104681	140.99	20.16	0
48	1192	uni.	-7507	-7507	133.93	23.02	0
49	1199	uni.	-7635	-7635	522.77	32.41	0
50	1204	gre.uni.	-37888	-37888	357.84	99.32	0
51	1404	bil.	-15633	-15633	372.51	17.86	0
52	1410	bil.	-16227	-16227	234.76	20.36	0
53	1403	bil.	-21445	-21445	294.03	16.89	0
54	1398	bil.	-31060	-31060	424.73	18.11	0
55	1407	bil.	-16173	-16173	121.44	2.52	0
56	1430	bil.	-27968	-27968	248.90	16.58	0
57	1432	bil.	-50713	-50713	456.57	19.17	0
58	1435	por.bil.	-8254	-8254	233.54	34.80	0
59	1422	uni.	-61984	-61984	60.81	17.87	0
60	1433	bil.	-61523	-61523	176.41	19.09	0
61	1438	bil.	-43895	-43895	232.67	23.55	0

Tableau 4.2 – Test (juillet 1998 , 183 premiers employés)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
62	1428	uni.	-28073	-28073	270.14	29.64	0
63	1423	bil.	-10983	-10983	357.39	21.25	0
64	1444	bil.	-32596	-32596	99.78	20.82	0
65	1426	bil.	-52078	-52078	281.28	24.44	0
66	1434	uni.	-16912	-16912	126.13	19.99	0
67	1484	bil.	-33451	-33451	186.74	14.65	0
68	1509	bil.	-7714	-7714	114.56	43.79	0
69	1500	bil.	-180567	-180567	202.10	30.88	0
70	1448	bil.	-14042	-14042	195.10	18.62	0
71	1502	uni.	-14047	-14047	235.14	21.90	0
72	1481	por.bil.	-6722	-6722	214.64	36.79	0
73	1504	bil.	-36476	-36476	89.91	30.58	0
74	1499	bil.	-158471	-158471	227.42	20.82	0
75	1533	bil.	-57123	-57123	286.23	16.47	0
76	1545	bil.	-65111	-65111	113.82	35.80	0
77	1541	bil.	-15627	-15627	168.36	17.35	0
78	1543	uni.	-15627	-15627	286.37	34.23	0
79	1547	bil.	-105551	-105551	54.75	30.97	0
80	1548	uni.	-40197	-40197	418.75	20.87	0
81	1657	bil.	-41876	-41876	211.21	34.95	0
82	1570	bil.	2394	2394	156.14	19.49	0
83	1600	bil.	57051	57051	136.03	20.93	0
84	1579	bil.	-51292	-51292	128.27	19.88	0
85	1615	por.bil.	-62868	-62868	207.19	16.84	0
86	1591	bil.	-17867	-17867	131.59	41.22	0
87	1666	bil.	-17867	-17867	94.82	39.94	0
88	1595	bil.	-138059	-138059	155.90	19.99	0
89	1576	bil.	-138059	-138059	219.76	18.18	0
90	1637	bil.	-14700	-14700	181.57	22.03	0
91	1606	bil.	-28386	-28386	119.49	15.69	0
92	1667	bil.	-48447	-48447	78.56	30.66	0
93	1635	bil.	-46014	-46014	235.32	13.94	0
94	1607	por.bil.	-30286	-30286	46.39	16.30	0
95	1561	bil.	40374	40374	231.91	22.20	0
96	1633	bil.	-24198	-24198	128.85	18.05	0
97	1583	ita.bil.	-62251	-62251	183.85	17.52	0
98	1560	bil.	-91150	-91150	848.79	16.05	0
99	1851	bil.	-40025	-40025	251.92	15.06	0
100	2005	bil.	0	0	275.08	12.98	0
101	1988	uni.	172620	172620	200.20	21.40	0
102	1970	bil.	-24156	-24156	141.38	20.34	0
103	1998	bil.	-85280	-85280	178.99	16.29	0
104	1957	bil.	-2724	-2724	91.20	23.91	0
105	2004	bil.	-1731	-1731	124.24	21.40	0
106	2012	uni.	-9666	-9666	79.14	22.51	0

Tableau 4.2 – Test (juillet 1998 , 183 premiers employés)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
107	2042	uni.	-15084	-15084	212.55	32.72	0
108	1952	uni.	-51616	-51616	88.37	15.84	0
109	1961	uni.	-10371	-10371	104.03	23.00	0
110	1951	uni.	-15069	-15069	133.28	26.34	0
111	1964	bil.	-71837	-71837	168.85	18.96	0
112	2002	esp.bil.	-55562	-55562	110.85	19.99	0
113	2040	pol.uni.	-19721	-19721	40.15	21.00	0
114	2057	uni.	-13539	-13539	95.44	17.73	0
115	2029	bil.	-153819	-153819	225.38	14.06	0
116	1972	bil.	-83107	-83107	101.89	24.45	0
117	1989	uni.	-33842	-33842	147.38	21.49	0
118	1968	gre.uni.	-96538	-96538	468.46	248.83	2
119	2024	bil.	-26837	-26837	143.71	25.61	0
120	2003	bil.	-34119	-34119	174.00	13.40	0
121	1950	uni.	-130967	-130967	37.32	18.85	0
122	1955	por.bil.	-16950	-16950	62.03	19.88	0
123	2049	ita.por.uni.	-45430	-45430	134.12	31.63	0
124	1984	uni.	-12505	-12505	87.98	18.43	0
125	1990	uni.	-19335	-19335	204.17	17.40	0
126	2017	bil.	145976	145976	146.54	19.87	0
127	1954	uni.	-15938	-15938	53.44	15.80	0
128	2070	esp.uni.	-67757	-67757	28.87	23.37	0
129	2076	bil.	-14566	-14566	174.90	29.80	0
130	2079	gre.bil.	-61782	-61782	82.46	69.99	4
131	2084	uni.	-18325	-18325	75.44	32.17	0
132	2081	bil.	-45451	-45451	151.99	5.87	0
133	2090	bil.	-47519	-47519	56.15	19.51	0
134	2080	uni.	-6166	-6166	69.09	19.52	0
135	2063	uni.	-52114	-52114	141.29	16.16	0
136	2069	bil.	-48656	-48656	63.11	16.83	0
137	2077	uni.	16311	16311	88.92	18.84	0
138	2067	bil.	-20145	-20145	75.90	14.45	0
139	2088	bil.	-35049	-35049	47.92	19.78	0
140	2091	bil.	-52555	-52555	109.81	27.33	0
141	2073	bil.	-42947	-42947	69.62	29.36	0
142	2126	uni.	-78603	-78603	59.43	20.67	0
143	2223	ita.bil.	-29018	-29018	78.34	18.82	0
144	2432	bil.	-42438	-42438	44.32	18.88	0
145	2434	bil.	-13863	-13863	102.15	28.58	0
146	2436	bil.	-10663	-10663	42.39	17.24	0
147	2486	bil.	-32818	-32818	95.59	23.83	0
148	2488	bil.	-648	-648	110.00	14.78	0
149	2592	bil.	-65974	-65974	60.69	30.23	0
150	2576	bil.	-39446	-39446	96.59	18.69	0
151	2594	bil.	-95205	-95205	85.37	25.23	0

Tableau 4.2 – Test (juillet 1998 , 183 premiers employés)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
152	2848	bil.	-56720	-56720	68.90	15.59	0
153	2883	esp.bil.	-11261	-11261	82.01	15.68	0
154	2843	bil.	-81011	-81011	62.85	11.40	0
155	2853	bil.	-45193	-45193	95.87	32.77	0
156	2896	bil.	-40602	-40602	48.54	22.27	0
157	2865	bil.	-45499	-45499	61.42	17.56	0
158	2891	pol.bil.	-39449	-39449	97.43	9.06	0
159	2845	bil.	-58353	-58353	81.85	18.88	0
160	2880	bil.	-75591	-75591	202.25	11.50	0
161	2854	bil.	-15754	-15754	82.57	18.29	0
162	2844	bil.	-100601	-100601	76.99	23.41	0
163	2895	bil.	-105353	-105353	110.81	17.42	0
164	2861	bil.	-74952	-74952	106.51	25.78	0
165	2859	bil.	-41683	-41683	33.88	16.76	0
166	2820	bil.	-39878	-39878	80.13	16.84	0
167	2852	bil.	-113672	-113672	61.53	21.52	0
168	2840	bil.	-37555	-37555	86.95	22.18	0
169	2807	bil.	-55105	-55105	64.54	13.77	0
170	2882	bil.	-32299	-32299	153.67	34.27	0
171	2889	bil.	-98982	-98982	72.57	20.37	0
172	2869	bil.	-107725	-107725	92.61	30.88	0
173	2856	por.bil.	-87873	-87873	88.75	29.51	0
174	2814	bil.	59417	59417	31.26	17.33	0
175	2866	bil.	-12536	-12536	72.46	14.83	0
176	2875	bil.	-66549	-66549	157.21	17.09	0
177	2864	bil.	-64298	-64298	4464.07	34.26	0
178	2894	bil.	90462	90462	151.34	17.15	0
179	2868	bil.	-81213	-81213	3973.58	19.80	0
180	2885	ita.bil.	-53626	-53626	1721.44	18.01	0
181	2873	bil.	-69678	-69678	2454.68	19.30	0
182	2876	bil.	-20155	-20155	1747.04	6.67	0
183	2941	bil.	-14144	-14144	830.38	16.04	0

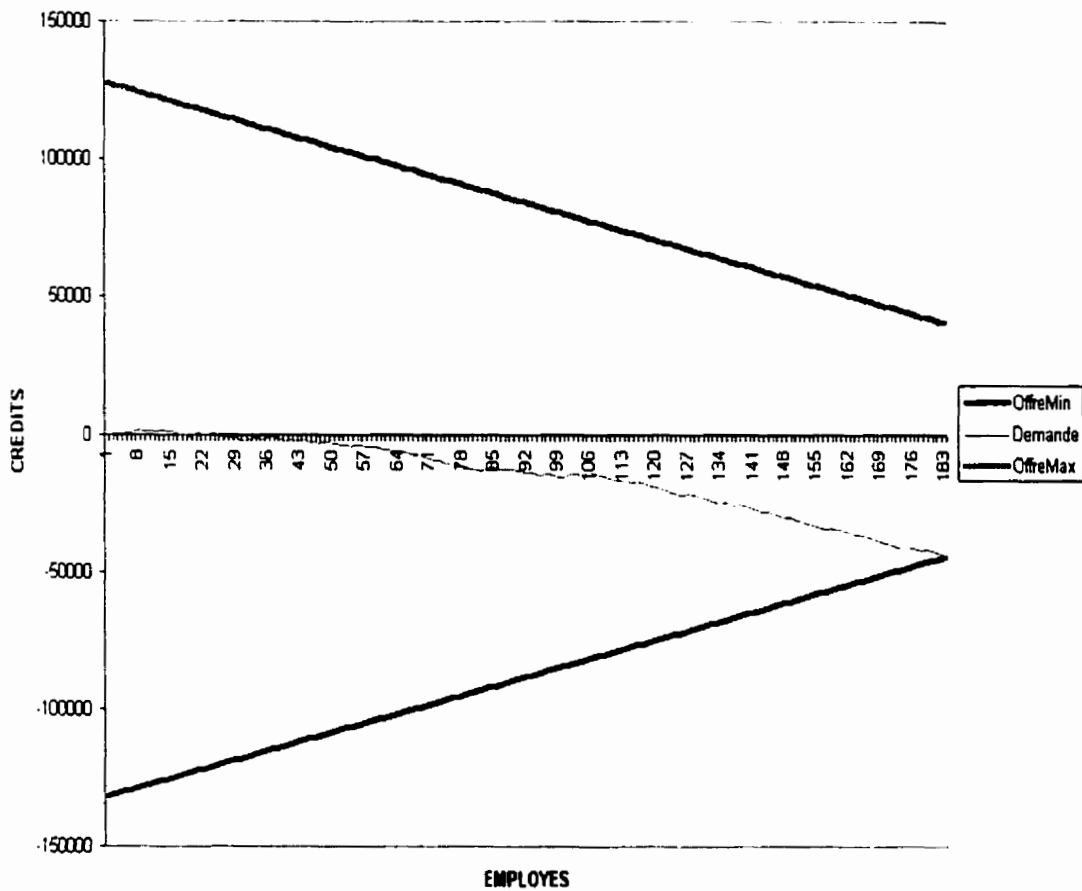


Figure 4.3 – Compteur C2 Global (Juillet 1998, 1^{er} au 183^{ième} employé)

4.3 Août 1998.

Ce « petit » problème, contrairement aux deux précédents, n'a pas été considéré comme étant difficile à résoudre.

Le cadre, calculé par Gencol, contient 100 employés assignés. Les compteurs C₁ n'ont été critiques que pour seulement 2 employés et le compteur C₂ global est devenu critique (demande = OffreMin) au 60^{ième} employé. Par la suite, la résolution s'est continuée à l'aide de Gencol jusqu'au 100^{ième} employé. Évidemment, on remarque que la différence de temps d'exécution entre Gencol et l'Heuristique est beaucoup moins frappante et cela est tout à fait normal vu la taille du problème.

Le tableau qui suit correspond à la résolution par l'heuristique jusqu'au 59^{ième} employé assigné.

Note : le paramètre « SIZE » a été fixé à **250 megs** dans les tests.

Temps total requis par l'heuristique (jusqu'au 59) : 2 minutes 45 sec.

Temps total requis par Gencol (jusqu'au 59) : 8 minutes 30 sec.

Paramètres : FacteurC₂Min = 0.7

FacteurC₂Max = 0.7

Attributs :	Bilingue :	bil.
	« Unianglais » :	uni.
	Portugais :	por.
	Grec :	gre.
	Italien :	ita.

Allemand : all.
 Espagnol : esp.
 Polonais : pol.
 Hollandais : hol.

Tableau 4.3 – Test (août 98 , 1^{er} au 59^{ème} employé)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
1	882	bil.	171728	171728	21.97	2.30	0
2	1216	por.bil.	-76543	-76543	13.41	3.53	0
3	1375	bil.	-246720	-246720	12.20	4.04	0
4	1478	bil.	-219936	-219936	7.87	0.39	0
5	1882	bil.	-76563	-76563	12.75	5.24	0
6	2033	bil.	-63199	-63199	16.94	4.83	0
7	1953	uni.	-21204	-21204	9.24	2.89	0
8	2115	uni.	-133805	-133805	16.29	4.49	0
9	2304	bil.	-41517	-41517	8.84	2.47	0
10	2329	bil.	-69998	-69998	8.54	2.33	0
11	2371	bil.	-69046	-69046	8.41	1.62	0
12	2441	bil.	-154871	-154871	15.19	3.78	0
13	2481	por.bil.	-50954	-50954	5.75	0.93	0
14	2490	bil.	-28705	-28705	3.55	0.88	0
15	2483	bil.	-24920	-24920	4.39	0.99	0
16	2470	all.bil.	-55921	-55921	17.81	6.86	0
17	2598	bil.	-232848	-232848	7.16	1.73	0
18	2635	bil.	-13255	-13255	5.54	0.60	0
19	2582	bil.	-114621	-114621	10.86	3.46	0
20	2553	bil.	-50662	-50662	8.18	2.36	0
21	2578	bil.	-91674	-91674	7.25	2.26	0
22	2641	bil.	157284	157284	3.02	0.23	0
23	2600	bil.	-24170	-24170	4.83	0.49	0
24	2589	bil.	-49567	-49567	8.32	2.17	0
25	2599	bil.	-35183	-35183	7.68	2.25	0
26	2665	bil.	-4057	-4057	6.29	2.60	0
27	2585	bil.	-30824	-30824	9.35	1.65	0
28	2571	bil.	-39564	-39564	5.82	1.76	0
29	2562	bil.	-134503	-134503	2.54	0.42	0
30	2587	bil.	-49904	-49904	5.42	1.11	0
31	2552	bil.	-74224	-74224	3.47	0.58	0
32	2637	bil.	-68629	-68629	2.43	0.58	0
33	2625	bil.	-229628	-229628	5.61	2.50	0
34	2569	ita.bil.	-229628	-229628	4.34	2.43	0
35	2564	bil.	-53421	-53421	6.17	1.27	0
36	2614	bil.	-9532	-9532	3.55	0.48	0
37	2819	bil.	-13206	-13206	3.49	1.38	0
38	2831	bil.	-61356	-61356	2.84	2.06	0

Tableau 4.3 – Test (août 98 , 1^{er} au 59^{ème} employé)

	<u>Emp.#</u>	<u>Attribut(s)</u>	<u>Solution (Gencol)</u>	<u>Solution (Heuristique)</u>	<u>Temps (sec.) (Gencol)</u>	<u>Temps (sec.) (Heuristique)</u>	<u>Nombre d'interv. C1 actifs.</u>
39	2825	all.bil.	-86221	-86221	14.53	1.37	2
40	2824	bil.	-127947	-127947	3.53	1.29	0
41	2818	bil.	-123127	-123127	2.28	0.30	0
42	2827	bil.	-65289	-65289	4.56	1.92	0
43	2816	bil.	-62724	-62724	3.86	1.31	0
44	2850	bil.	-14115	-14115	3.82	1.16	0
45	2826	ita.bil.	-16732	-16732	4.03	0.37	0
46	2822	bil.	-143842	-143842	2.91	1.92	0
47	2821	bil.	-154111	-154111	5.34	1.31	0
48	2872	bil.	-221475	-221475	3.60	1.25	0
49	2830	bil.	-91520	-91520	2.08	0.42	0
50	2836	bil.	-213349	-213349	3.07	1.08	0
51	2871	ita.bil.	-224694	-224694	3.05	0.65	0
52	2829	bil.	56074	56074	2.20	0.89	0
53	2812	bil.	-110739	-110739	2.05	0.30	0
54	2838	bil.	-34512	-34512	67.72	0.30	0
55	3022	hol.bil.	22874	22874	5.48	0.09	4
56	3035	bil.	-7582	-7582	12.57	0.11	0
57	3021	bil.	-6337	-6337	10.97	0.76	0
58	2999	bil.	-190635	-190635	11.27	0.72	0
59	2960	bil.	-194333	-194333	6.22	0.44	0

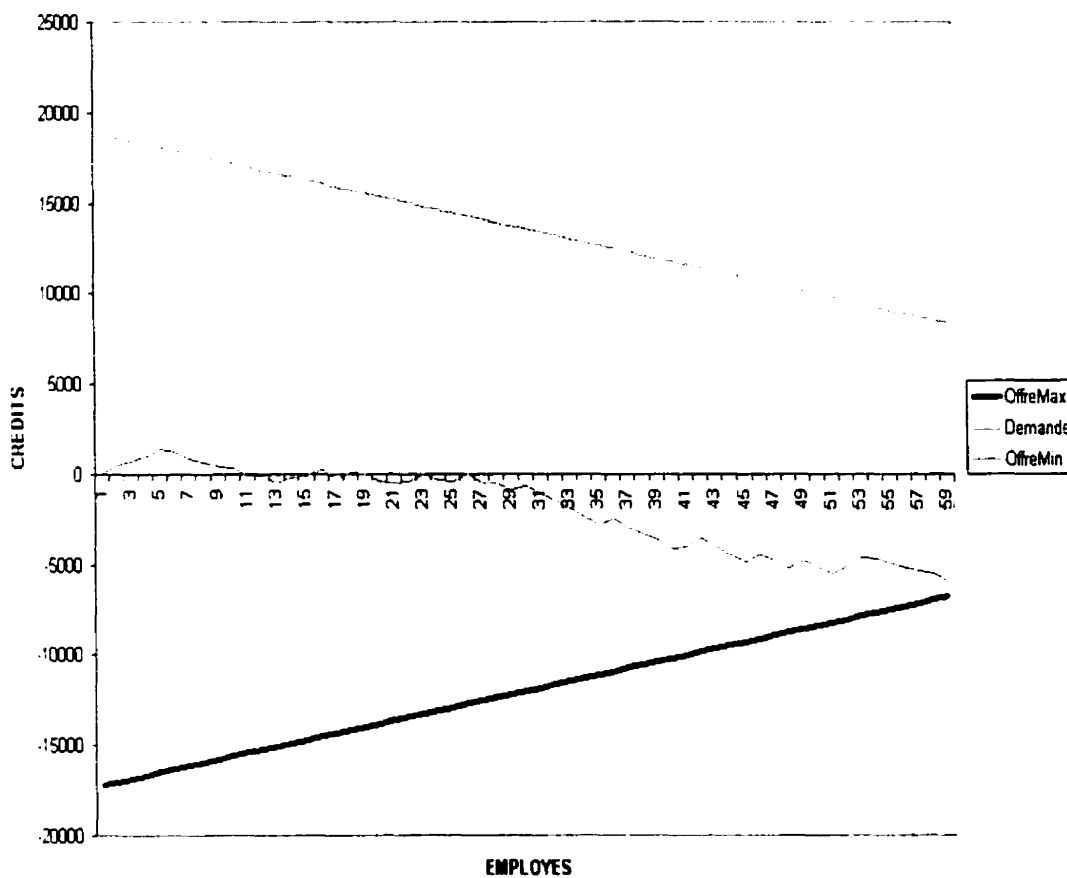


Figure 4.4 – Compteur C₂ Global (Août 1998, 1^{er} au 59^{ième} employé)

CONCLUSION

Ce projet de maîtrise a porté sur un nouvel algorithme heuristique permettant la construction de blocs mensuels personnalisés pour des pilotes ou agents de bord. Le système commercialisé actuellement résout une formulation optimale du problème mais son temps de résolution est beaucoup trop élevé. Celle-ci peut atteindre plus de 24 heures pour les plus grands problèmes. L'accroissement constant du nombre de pilotes dans les compagnies aériennes va amener des temps de résolution encore plus grands. Le nouveau système est également nécessaire pour résoudre les problèmes d'agents de bords. L'objectif est donc de développer un système capable de traiter des problèmes plus grands dans des temps raisonnables.

L'heuristique qui est présentée dans ce mémoire permet d'évaluer la faisabilité du problème global de manière très rapide grâce à des compteurs dynamiques. Les compteurs C_1 permettent d'enlever certains arcs du réseau d'un employé courant afin de s'assurer que les employés résiduels puissent tous avoir un bloc tel que l'ensemble des rotations résiduelles soient couvertes. Ce sont les compteurs C_2 qui cumulent des données pour vérifier que les blocs résiduels à être construits obtiendront une quantité légale de crédits de vols. Ces données permettent d'indiquer le moment où la résolution du problème devient trop complexe pour l'heuristique et doit être effectuée à l'aide de l'optimiseur Gencol.

Le temps de calcul est grandement diminué puisqu'un seul plus court chemin est effectué pour chaque problème k à résoudre. Ce qui fait donc que la complexité est pseudo-polynomiale au lieu d'être exponentielle comme dans le cas de Gencol. En effet, les tests effectués jusqu'à maintenant sont convaincants : le taux d'erreur est pratiquement nul et le temps de calcul est réduit par un facteur de 10 à 15 pour les plus gros problèmes. De plus, ce facteur augmente avec la taille du problème. Par conséquent, il est tout à fait raisonnable de prévoir que les plus gros problèmes à être résolus pourront passer de 300 employés à quelques milliers. Des plus gros jeux de données de l'ordre de 500 employés seront créés dans les prochains mois à venir afin de tester à nouveau la méthode. Le seul problème qui semblerait peut-être pouvoir assombrir les résultats est le fait que le réseau généré sera évidemment plus volumineux et que le plus court chemin sera alors plus complexe et nécessitera une grande quantité de mémoire vive. Mais une solution a déjà été trouvée pour remédier à cela : il s'agit d'utiliser les tâches mères abordées au chapitre 2. Ceci permettra de réduire le temps de calcul et la mémoire utilisée par un facteur de l'ordre de 5 puisqu'il y a moyenne 5 agents de bords dans un avion.

Parmi les nouvelles recherches reliées à l'heuristique, deux seront débutées d'ici peu. Un étudiant de doctorat étudiera une nouvelle méthode de branchement pour Gencol qui utilisera les valeurs des compteurs. Cette méthode a pour but d'améliorer les solutions trouvées pour les derniers problèmes k . Pour sa part, une étudiante de maîtrise étudiera la possibilité d'améliorer les compteurs C_2 à fin de repousser au plus

tard possible l'utilisation de Gencol. De plus, elle travaillera à l'implantation des tâches mères dans le réseau pour améliorer l'efficacité et la polyvalence de l'heuristique.

L'heuristique est déjà transférée chez AD OPT et certaines grandes compagnies aériennes telles qu'American Airlines ont signifié leur intérêt pour ce nouveau système.

RÉFÉRENCES BIBLIOGRAPHIQUES

DESROCHERS, M. (1986). La fabrication d'Horaires de Travail pour les Conducteurs d'Autobus par une méthode de Génération de colonnes, Thèse de doctorat, Université de Montréal, Centre de recherche sur les transports, Canada.

DESROCHERS, M., SOUMIS, F., (1989). A Column Generation Approach to the Urban Transit Crew Scheduling Problem. Transportation Science, 23 :1-13.

DESROCHERS, J., DUMAS, Y., SOLOMON, M.M., SOUMIS, F., (1995). Time Constrained Routing and Scheduling. In M.O. Ball et al. (eds.), Network Routing, Handbooks in Operations Research and Management Science 8. Elsevier Science, 35-139.

DESAULNIERS, G., DESROSIERS, J., IOACHIM, I., SOLOMON, M.M., SOUMIS, F., VILLENEUVE, D., (1998). A Unified Framework for Deterministic Time Constrained Vehicle Routing and Crew Scheduling Problems, Fleet Management and Logistics, T. G. Crainic and G. Laporte (eds.), Kluwer, Norwell, MA, 57-93

GAMACHE, M., SOUMIS, F., VILLENEUVE, D., DESROSIERS, J., GÉLINAS, E., (1998), The Preferential Bidding System at Air Canada, Transportation Science, 32(3), 246-255

GAMACHE, M., SOUMIS, F., VILLENEUVE, D., DESROSIERS, J., GÉLINAS, E.,
(1998), Le système d'horaires mensuels personnalisés d'Air Canada, Les Cahiers du
GERAD, G-98-56, 20 pages.