UNIVERSITÉ LAVAL

Travail pratique

Travail présenté à Claude-Guy Quimper
IFT-7020 Optimisation combinatoire
Session d'hiver 2018

Réalisé par
François Bérubé et François Pelletier ;
900226407, 908144032

Dernière version produite le 12 février 2018 à 22:08

Table des matières

Table des figures

1 Problème 1

Modélisation et résolution du problème des quatre cubes avec le solveur Choco 4 [1].

1.1 Constantes

Deux constantes sont définies dans l'énoncé du problème. Il s'agit du nombre de cubes N ainsi que du nombre de faces F sur le prisme rectangulaire. Ces deux constantes sont fixées à 4.

Afin de faciliter la création des contraintes dans le solveur Choco, on définit une application : l'ensemble de nombres $\{1, 2, 3, 4\}$ sert à représenter l'ensemble de quatre couleurs $\{rouge, vert, bleu, jaune\}$, dans cet ordre (1).

$$\left\{ \begin{array}{ccc} 1 & \mapsto & \text{Rouge} \\ 2 & \mapsto & \text{Vert} \\ 3 & \mapsto & \text{Bleu} \\ 4 & \mapsto & \text{Jaune} \end{array} \right\} \tag{1}$$

1.2 Variables

On définit une matrice FC de taille $N \times F$ où chaque variable fc_{ij} correspond à la couleur que présente le cube i sur la face j du prisme rectangulaire.

$$\begin{aligned} fc_{ij} &\in \{1, 2, 3, 4\} && \forall 1 \leq i \leq N, 1 \leq j \leq F \\ dom(fc_{ij}) = \{1, 2, 3, 4\} &\in \theta(1) && \forall 1 \leq i \leq N, 1 \leq j \leq F \end{aligned} \tag{2}$$

1.2.1 Nombre total de variables et de valeurs

Le nombre total de variables pour la définition (2) :

$$NF \in \theta(NF) \tag{3}$$

Le nombre total de valeurs possibles pour ces mêmes définitions est :

$$NF \times 4 = 4NF \in \theta(NF) \tag{4}$$

1.3 Contraintes

1.3.1 Tableaux des couleurs

La contrainte du tableau des couleurs s'énonce comme suit : pour chaque cube i, un tableau doit être construit dans lequel seront énumérées toutes les séquences possibles des couleurs apparaissant sur les 4 faces du prisme. Pour chacun des 4 cubes, il y a 24 combinaisons (6 faces pointant dans une direction $\times$ 4 orientations par face) possibles. Si certaines combinaisons se répètent, ces doublons seront pris en charge par le solveur.

Il s'agit d'une contrainte de type Tableau qui s'écrit de la façon suivante :

$$\textsc{Tableau}(\{fc_{i1}, \ldots, fc_{iF}\}, T) \qquad \forall 1 \leq i \leq N \tag{5}$$

1.3.2 Couleurs différentes sur les 4 faces

Le problème des 4 cubes consiste à trouver un agencement de ces cubes permettant d'obtenir chacune des 4 couleurs sur les 4 faces rectangulaires du prisme créé. Ce qui équivaut à ne pas avoir une paire de couleurs identiques sur une face latérale du prisme (6).

$$fc_{ij_1} \neq fc_{ij_2} \qquad \forall 1 \leq i \leq N, 1 \leq j_1 < j_2 \leq F \tag{6}$$

$$\tag{7}$$

Il s'agit d'une contrainte de type AllDifferent (8) pour chaque face latérale du prisme.

$$\textsc{AllDifferent}(\{fc_{i1}, \ldots, fc_{iF}\}) \forall 1 \leq i \leq N \tag{8}$$

1.3.3 Nombre total de contraintes

Le nombre total de contraintes pour les définitions (5) et (8) est :

$$N\text{contraintes «tableau»} + \frac{4 * N(N-1)}{2}\text{contraintes de différence} = 2N^2 - N \in \theta(N^2) \tag{9}$$

De plus, chaque tableau contient au plus $24 \in O(24)$ entrées.

1.4 Résultats

Nous avons demandé au solveur Choco de trouver une solution au problème des 4 cubes à l'aide de la méthode `findSolution`. Nous avons utilisé les euristiques par défaut, car elles permettaient d'obtenir un résultat en moins d'une seconde. Le solveur trouve une seule solution en n'effectuant aucun retour arrière, et ce en 0,038s.

1.4.1 Solution

La solution retournée par le solveur est représentée à la figure 1.

	Cube 0	Cube 1	Cube 2	Cube 3
Face 0	R	B	V	J
Face 1	V	J	B	R
Face 2	V	R	J	B
Face 3	J	B	R	V

Figure 1 – Solution du problème 1

2 Question 2

L'objectif de ce problème est de concevoir un horaire de travail avec minimisation de la perte de profit en utilisant le solveur Choco 4 [1].

2.1 Constantes

Plusieurs constantes sont définies par l'énoncé du problème. On définit les périodes $t = 1, \ldots, 16$ de la journée comme étant des périodes de 30 minutes débutant à 9h00 et se terminant à 16h59.

— N = Nombre d'employés disponibles
— MIN_H = Nombre minimal de périodes travaillées par l'employé, incluant la pause
— MAX_H = Nombre maximal de périodes travaillées par l'employé, incluant la pause
— MIN_P = Nombre minimal de périodes dans le bloc de travail précédant et suivant la pause
— e_j^{req} = Nombre d'employés requis à la période $j \in \{1, 16\}$.
— e_j^{souh} = Nombre d'employés souhaités à la période $j \in \{1, 16\}$.
— P = Nombre de périodes dans l'horaire. Constante initialisée à 16.
— V_{PERTE} = Valeur de la perte liée à un écart d'un employé par rapport au nombre souhaité. Constante initialisée à 20.

On définit aussi ces constantes supplémentaires pour simplifier la création de contraintes, car Choco 4 ne permet pas d'effectuer d'opérations arithmétiques dans une contrainte autre que la contrainte arithmétique. Autrement il aurait fallu ajouter des contraintes redondantes à notre modèle.

— $MAX_P = MAX_H - MIN_P - 1$ = Nombre maximal de périodes dans le bloc de travail précédant et suivant la pause.
— $MAX_{PR} = MAX_P - MIN_H - 1$ = Nombre maximal de périodes dans le bloc de repos débutant ou terminant la journée.
— $MIN_{HT} = MIN_H - 1$ = Nombre minimal de périodes travaillées par l'employé, excluant la pause
— $MAX_{HT} = MAX_H - 1$ = Nombre maximal de périodes travaillées par l'employé, excluant la pause

2.2 Variables

On définit une matrice de variables E_{NP} où e_{ij} correspondant à une valeur booléenne prenant la valeur 1 si l'employé i travaille à la période j et 0 sinon.

$$\begin{aligned} e_{ij} &= 1_{\text{Employé } i \text{ travaille à la période j}} \forall 1 \leq i \leq N, 1 \leq j \leq P \qquad (10) \\ dom(e_{ij}) = \{0,1\} \in \theta(1) & \qquad \forall 1 \leq i \leq N, 1 \leq j \leq P \end{aligned}$$

Afin de résoudre le problème d'optimisation, on doit définir des variables supplémentaires, car les composantes de la fonction d'optimisation sont aussi exprimées sous forme de contraintes.

e_j^{eff}, le nombre d'employés effectif à la période j, et e_j^{dist} la différence absolue entre le nombre d'employés effectif et souhaité à la période j.

$$e_j^{eff} = \sum_{i=1}^{N} e_{ij} \qquad \forall 1 \leq j \leq P \tag{11}$$
$$dom(e_j^{eff}) = \{0, N\} \in \theta(N) \qquad \forall 1 \leq j \leq P$$
$$e_j^{dist} = |e_j^{eff} - e_j^{souh}| \qquad \forall 1 \leq j \leq P \tag{12}$$
$$dom(e_j^{dist}) = \{0, N\} \in \theta(N) \qquad \forall 1 \leq j \leq P$$

Enfin, nous créons une variable N_{PERTE} représentant le total des unités de perte, équivalant à la somme des différences absolues. Le nombre total d'unités de perte prend une valeur entre 0 et NP.

$$N_{PERTE} = \sum_{j=1}^{P} e_j^{dist} \tag{13}$$
$$dom(N_{PERTE}) = \{0, \ldots, NP\} \qquad \in \theta(NP)$$

2.2.1 Nombre total de variables et de valeurs

Le nombre total de variables pour les définitions (10), (11), (12) et (13) :

$$NP + P + P + 1 = (N + 2) \times N_p + 1 \in \theta(NP) \tag{14}$$

Le nombre total de valeurs possibles pour ces mêmes définitions est :

$$NP \times 2 + NP + NP + NP = 5NP \in \theta(NP) \tag{15}$$

2.3 Contraintes

2.3.1 Nombre d'employés requis

La contrainte du nombre d'employés requis s'énonce comme suit : pour chaque période i, le nombre d'employés travaillant doit être supérieur ou égal au nombre d'employés minimum requis.

$$\sum_{i=1}^{N} e_{ij} \geq e_j^{req} \qquad \forall 1 \leq j \leq P \tag{16}$$

Il s'agit d'une contrainte de type Sum :

$$\textsc{Sum}(\{e_{1j}, \ldots, e_{Nj}\}, \geq, e_j^{req}) \qquad \forall 1 \leq j \leq P \tag{17}$$

2.3.2 Motif de la journée de travail

Afin de représenter le motif correspondant à une journée de travail alternant les périodes de travail et celles de repos, on utilisera une contrainte de type Regular qui est paramétrée par un automate fini, c'est-à-dire une chaine de caractère formant une expression régulière qui valide plusieurs contraintes définies par l'énoncé du problème.

Dans Choco 4, la contrainte `model.regular` prend en paramètre une séquence de variables entières à valider, et un automate fini défini par la classe `FiniteAutomaton`. Pour construire une instance de cette classe, on doit construire l'expression régulière en n'utilisant que des nombres comme valeurs. Comme nous avons choisi de représenter les différents états d'un employé par une variable boolienne, cette condition est satisfaite.

Voici comment nous avons construit l'expression :

— On débute par un bloc de repos d'au moins 0 et au plus MAX_{PR} périodes.

— On a ensuite un bloc de travail d'au moins MIN_P et au plus MAX_P périodes.

— On a ensuite un bloc de repos d'une durée d'une période.

— On a ensuite un bloc de travail d'au moins MIN_P et au plus MAX_P périodes.

— On termine par un bloc de repos d'au moins 0 et au plus MAX_{PR} périodes.

On obtient ainsi un automate fini défini par l'expression régulière suivante :

$$\begin{aligned} FA = & 0\{0, MAX_{PR}\} \\ & 1\{MIN_P, MAX_P\} \\ & 0 \\ & 1\{MIN_P, MAX_P\} \\ & 0\{0, MAX_{PR}\} \end{aligned} \tag{18}$$

L'ensemble des contraintes pour le motif d'horaire peut donc s'écrire

$$\textsc{Regular}(\{e_{i1}, \ldots, e_{ij}\}, FA) \qquad \forall 1 \leq i \leq N \tag{19}$$

2.3.3 Nombre de périodes travaillées par employé

On définit l'inégalité suivante pour définir la contrainte du nombre minimum et maximum de périodes travaillées par employé.

$$MIN_{HT} \leq \sum_{j=1}^{P} e_{ij} \leq MAX_{HT} \qquad \forall 1 \leq i \leq N \tag{20}$$

On utilise deux contraintes Sum pour représenter dans Choco le nombre de périodes minimum et maximum devant être travaillées, pour chaque employé.

$$\begin{aligned} &\textsc{Sum}(\{e_{i1}, \ldots, e_{iP}\}, \geq, MIN_{HT}) && \forall 1 \leq i \leq N \\ &\textsc{Sum}(\{e_{i1}, \ldots, e_{iP}\}, \leq, MAX_{HT}) && \forall 1 \leq i \leq N \end{aligned} \tag{21}$$

2.3.4 Nombre total de contraintes

Le nombre total de contraintes pour les définitions (17), (19) et (21) est :

$$P + N + 2N = P + 3N \in \theta(N) \tag{22}$$

2.4 Optimisation

Dans Choco, les contraintes d'optimisation sont créées de la même façon que les contraintes de satisfaction. Cependant, une des variables ne sera pas une constante, mais un intervalle dans lequel la fonction d'optimisation va effectuer sa recherche.

Nous implémenterons le nombre de travailleurs effectifs e_j^{eff} à la période j en utilisant une contrainte SUM :

$$\text{SUM}(\{e_{1j}, \ldots, e_{Nj}\}, =, e_j^{eff}) \qquad \forall 1 \leq j \leq P \tag{23}$$

Nous implémenterons ensuite la différence entre le nombre de travailleurs effectifs et souhaités e_j^{dist} à la période j en utilisant une contrainte DISTANCE :

$$\text{DISTANCE}(e_j^{souh}, e_j^{eff}, =, e_j^{dist}) \qquad \forall 1 \leq j \leq P \tag{24}$$

Enfin, la variable N_{PERTE} est implémentée par une contrainte SUM :

$$\text{SUM}(\{e_1^{dist}, \ldots, e_P^{dist}\}, =, N_{PERTE}) \tag{25}$$

Nous n'incluons pas la valeur en dollars des unités de perte dans le modèle, car elle n'est pas pertinente dans l'élicitation des contraintes. Nous n'aurons qu'à multiplier le résultat obtenu par 20$.

Nous avons donc ici $P + P + 1 \in \theta(P)$ contraintes d'optimisation.

2.5 Résultats

Nous avons demandé au solveur Choco de minimiser la perte N_{PERTE} à l'aide de la méthode `findOptimalSolution`. Nous avons utilisé les euristiques par défaut, car elles permettaient d'obtenir un résultat en moins d'une seconde. Le solveur trouve deux solutions optimales en effectuant 6397 retours arrière en 0,553s. La valeur optimale est de 4 unités de perte, pour une valeur de 80$.

2.5.1 Solution

La solution retournée par le solveur est représentée à la figure 2.

```
Employé 0: 0  0  0  1  1  1  1  1  0  1  1  1  1  1  1  1
Employé 1: 0  1  1  1  1  1  0  1  1  1  1  1  1  0  0  0
Employé 2: 1  1  1  1  1  0  1  1  1  1  1  1  0  0  0  0
Employé 3: 0  0  0  0  1  1  1  0  1  1  1  1  1  1  1  1
Employé 4: 0  0  1  1  1  1  0  1  1  1  1  1  1  1  1  1
```

FIGURE 2 – Solution du problème 2

Bibliographie

[1] Charles Prud'homme, Jean-Guillaume Fages, and Xavier Lorca. *Choco Solver Documentation*. TASC, INRIA Rennes, LINA CNRS UMR 6241, COSLING S.A.S., 2016. URL `http://www.choco-solver.org`.