



# INFO - F - 310

---

Projet d'Algorithmique et recherche opérationnelle  
Année académique 2017-2018

---

Auteurs :

**Jacobs** Alexandre  
Matricule ULB : 408850

# 1 Sens de la fonction objective

$$\max \sum_{i \in \{1,2\}} ( \sum_{j \in \{s,c\}} (10p_i^j + 2e_i^j) + 50b_i^s + 120a_i^s ) - 1200$$

Cette fonction objective représente le fait qu' Han Solo et Chewbacca veulent maximiser les revenus de chaque transport avec les différents crédits des cargaisons transportées entre Tatooine et Coruscant. Aussi, les  $-1200$  représentent le fait qu' Han solo utilise par trajet en moyenne 120 pièces (10 pièces à remplacer en moyenne par parsec et le trajet comptent 12 parsecs donc 120 pièces au total en moyenne) pour maintenir son vaisseau en ordre de marche, il doit les déduire des revenus, donc 120 pièces à 10 crédits font donc bien les 1200 qu'il retire.

Voici le sens des différentes variables utilisées :

- $\forall i \in \{1, 2\}, a_i^s$  représente le nombre de caisses d'armes convoyées dans le sas  $i$ .
- $\forall i \in \{1, 2\}, b_i^s$  représente le nombre de caisses de bâtons de la mort convoyées dans le sas  $i$ .
- $\forall i \in \{1, 2\}, e_i^s$  représente le nombre de kilos d'épices convoyés dans le sas  $i$ .
- $\forall i \in \{1, 2\}, e_i^c$  représente le nombre de kilos d'épices convoyés dans la cale  $i$  (en dehors du sas).
- $\forall i \in \{1, 2\}, p_i^s$  représente le nombre de pièces détachées convoyées dans le sas  $i$ .
- $\forall i \in \{1, 2\}, p_i^c$  représente le nombre de pièces détachées convoyées dans la cale  $i$  (en dehors du sas).

# 2 Sens des contraintes

$$1. \sum_{i \in \{1,2\}} (50a_i^s + 10b_i^s + \sum_{j \in \{s,c\}} (4p_i^j + e_i^j)) \leq 80000$$

Cette contrainte représente le fait que chaque cargaison chargé dans les différents sas et cales sommées ensemble avec leurs poids respectifs ne peuvent dépasser le poids maximal que le vaisseau ne peut accepter qui est de 80000 kilos.

$$2. \sum_{i \in \{1,2\}} \sum_{j \in \{s,c\}} p_i^j \geq 120$$

Cette contrainte représente le fait que les pièces utilisées pour réparer le vaisseau peut être supérieur ou égale à 120 pièces utilisées, 120 pièces est le minimum de pièces remplacées lors d'un voyage de 12 parsecs, sachant que pour 1 parsec en moyenne 10 pièces sont à remplacer.

$$3. \sum_{i \in \{1,2\}} \sum_{j \in \{s,c\}} p_i^j \leq 7500$$

Cette contrainte représente le fait que le nombre total de pièces chargées dans le vaisseau ne peut dépasser le nombre de pièces total, 7500 pièces, à transporter.

$$4. \sum_{i \in \{1,2\}} a_i^s \leq 50$$

Cette contrainte représente le fait que le nombre total de caisses d'armes chargées dans le vaisseau ne peut dépasser le nombre de caisses total, 50 caisses, à transporter.

$$5. \sum_{i \in \{1,2\}} b_i^s \leq 100$$

Cette contrainte représente le fait que le nombre total de caisses de bâtons de la mort chargées dans le vaisseau ne peut dépasser le nombre de caisses total, 100 caisses, à transporter.

$$6. 0,3a_i^s + 0,1b_i^s + 0,001e_i^s + 0,1p_i^s \leq 5 \quad \forall i \in \{1, 2\}$$

Cette contrainte représente le fait que le volume de chargement total de chacun des sas 1 et 2 ne peuvent excéder les  $5m^3$ , en prenant compte des volumes des différentes cargaisons qui peuvent y être entreposées et de leurs nombres totaux.

$$7. 0,001e_i^c + 0,1p_i^c \leq 95 \quad \forall i \in \{1, 2\}$$

Cette contrainte représente le fait que le volume de chargement total de chacune des cales 1 et 2 ne peuvent excéder les  $95m^3$ , en prenant compte des volumes des différentes cargaisons qui peuvent y être entreposées et de leurs nombres totaux.

$$8. 50(a_1^s - a_2^s) + 10(b_1^s - b_2^s) + \sum_{j \in \{s, c\}} (e_1^j - e_2^j + 4(p_1^j - p_2^j)) = 0$$

Cette contrainte représente le fait que le vaisseau doit être équilibré, donc que le poids total de chaque sas et cales soit identique.

$$9. a_i^s \geq 0 \quad \forall i \in \{1, 2\}$$

$$10. b_i^s \geq 0 \quad \forall i \in \{1, 2\}$$

$$11. e_i^j \geq 0 \quad \forall i \in \{1, 2\}, \forall j \in \{s, c\}$$

$$12. p_i^j \geq 0 \quad \forall i \in \{1, 2\}, \forall j \in \{s, c\}$$

Ces 4 dernières contraintes représentent le fait que les cargaisons à charger peuvent être nulles, mais pas négatives.

### 3 Contraintes Superflues

Les contraintes superflues sont :

$$— I3 \sum_{i \in \{1, 2\}} \sum_{j \in \{s, c\}} p_i^j \geq 120$$

Cette contrainte est superflue du fait que la fonction objective avec les contraintes *I7* et *I8* suffisent afin de dire qu'il y aura au minimum 120 pièces de charger dans les cales. Le fait de charger des pièces un maximum permet d'augmenter le revenu d'un voyage par rapport à un voyage ne prenant aucune pièce dans son chargement. Exemple : si on l'on charge le vaisseau avec seulement 80000 kilos d'épices, le revenu ne serait que de 158800 crédits selon la fonction objective, et à contrario si on chargeait 120 pièces et le reste d'épices, 79520 kilos d'épices, on aurait un revenu de 159040 crédits qui est bien supérieur au revenu sans les pièces dans son chargement. En conclusion, on peut donc retirer la contrainte de la modélisation du problème, car on essayera presque tout le temps de charger au maximum des pièces qui seront supérieures aux 120 pièces décrites dans la contrainte.

$$— I5 \sum_{i \in \{1, 2\}} a_i^s \leq 50$$

Cette contrainte est superflue du fait que le volume des caisses d'armes dans les cales 1 et 2 occupe  $0.3m^3$ , comme le volume total des cales 1 et 2 est de  $10m^3$  on ne peut au maximum y charger que 33 caisses d'armes, ce qui est moins que les 50 spécifié dans la contrainte qui à un volume plus élevé ( $15m^3$ ) que le volume disponible ( $10m^3$ ) pour les cales. En conclusion, on peut donc retirer la contrainte de la modélisation du problème.

$$— I6 \sum_{i \in \{1, 2\}} b_i^s \leq 100$$

Cette contrainte est superflue du même fait que celle de la contrainte *I5* que le volume des caisses de bâtons de la mort dans les cales 1 et 2 occupe  $0.1m^3$ , comme le volume total des cales 1 et 2 est de  $10m^3$  on ne peut au maximum y charger que les 100 caisses de disponibles. En conclusion, on peut donc retirer la contrainte de la modélisation du problème, car le volume maximal égale le volume de disponible pour les cales.

On peut donc remarquer avec les résultats de la résolution avec GLPK en excluant les 3 contraintes superflues et avec que cela reste totalement pareil au niveau des solutions obtenues.

```

Problem:
Rows: 10
Columns: 13
Non-zeros: 48
Status: OPTIMAL
Objective: obj = 176800 (MAXimum)

No. Row name St Activity Lower bound Upper bound Marginal
-----
1 I 2 NU 80000 80000 2
2 I 4 NU 7500 7500 2
3 I 7 1 NU 5 5 300
4 I 7 2 NU 5 5 300
5 I 8 1 B 84.5 95
6 I 8 2 B 39.5 95
7 I 9 NS 0 0 = < eps
8 I 3 B 7500 120 50
9 I 5 B 0 50
10 I 6 B 100 100

No. Column name St Activity Lower bound Upper bound Marginal
-----
1 p 1 s NL 0 0 -3
2 p 2 s NL 0 0 -3
3 p 1 c B 7500 0
4 p 2 c NL 0 0 < eps
5 e 1 s NL 0 0 -0.3
6 e 2 s NL 0 0 -0.3
7 e 1 c B 9500 0
8 e 2 c B 39500 0
9 b 1 s B 50 0
10 b 2 s B 50 0
11 a 1 s NL 0 0 -70
12 a 2 s NL 0 0 -70
13 cout maintenance NS 1200 1200 = -1

Karush-Kuhn-Tucker optimality conditions:
KKT-PE: max.abs.err = 1.46e-11 on row 1
max.rel.err = 9.09e-17 on row 1
High quality
KKT-PB: max.abs.err = 0.00e+00 on row 0
max.rel.err = 0.00e+00 on row 0
High quality
KKT-DE: max.abs.err = 4.97e-14 on column 9
max.rel.err = 4.92e-16 on column 9
High quality
KKT-DB: max.abs.err = 0.00e+00 on row 0
max.rel.err = 0.00e+00 on row 0
High quality
End of output

```

FIGURE 1 – Solution du problème à l'aide la modélisation d'un PL avec toutes les contraintes.

```

Problem:
Rows: 7
Columns: 13
Non-zeros: 40
Status: OPTIMAL
Objective: obj = 176800 (MAXimum)

No. Row name St Activity Lower bound Upper bound Marginal
-----
1 I 2 NU 80000 80000 2
2 I 4 NU 7500 7500 2
3 I 7 1 NU 5 5 300
4 I 7 2 NU 5 5 300
5 I 8 1 B 84.5 95
6 I 8 2 B 39.5 95
7 I 9 NS 0 0 = < eps

No. Column name St Activity Lower bound Upper bound Marginal
-----
1 p 1 s NL 0 0 -3
2 p 2 s NL 0 0 -3
3 p 1 c B 7500 0
4 p 2 c NL 0 0 < eps
5 e 1 s NL 0 0 -0.3
6 e 2 s NL 0 0 -0.3
7 e 1 c B 9500 0
8 e 2 c B 39500 0
9 b 1 s B 50 0
10 b 2 s B 50 0
11 a 1 s NL 0 0 -70
12 a 2 s NL 0 0 -70
13 cout maintenance NS 1200 1200 = -1

Karush-Kuhn-Tucker optimality conditions:
KKT-PE: max.abs.err = 7.28e-12 on row 7
max.rel.err = 9.09e-17 on row 7
High quality
KKT-PB: max.abs.err = 0.00e+00 on row 0
max.rel.err = 0.00e+00 on row 0
High quality
KKT-DE: max.abs.err = 2.84e-14 on column 12
max.rel.err = 2.54e-16 on column 3
High quality
KKT-DB: max.abs.err = 0.00e+00 on row 0
max.rel.err = 0.00e+00 on row 0
High quality
End of output

```

FIGURE 2 – Solution du problème à l'aide la modélisation d'un PL sans les contraintes superflues

## 4 Modélisation et Résolution du PL à l'aide de GLPK.

Ce PL est modélisé dans un fichier CPLEX LP "projet.lp", celui-ci est modélisé sans tenir compte des contraintes inutiles qui y sont mis en commentaires. Les variables utilisées dans cette modélisation du PL signifient :

- $a\_1\_s$  représente le nombre de caisses d'armes convoyées dans le sas 1.
- $a\_2\_s$  représente le nombre de caisses d'armes convoyées dans le sas 2.
- $b\_1\_s$  représente le nombre de caisses de bâtons de la mort convoyées dans le sas 1.
- $b\_2\_s$  représente le nombre de caisses de bâtons de la mort convoyées dans le sas 2.
- $e\_1\_s$  représente le nombre de kilos d'épices convoyés dans le sas 1.

- $e\_2\_s$  représente le nombre de kilos d'épices convoyés dans le sas 2.
- $e\_1\_c$  représente le nombre de kilos d'épices convoyés dans la cale 1 (en dehors du sas).
- $e\_2\_c$  représente le nombre de kilos d'épices convoyés dans la cale 2 (en dehors du sas).
- $p\_1\_s$  représente le nombre de pièces détachées convoyées dans le sas 1.
- $p\_2\_s$  représente le nombre de pièces détachées convoyées dans le sas 2.
- $p\_1\_c$  représente le nombre de pièces détachées convoyées dans la cale 1 (en dehors du sas).
- $p\_2\_c$  représente le nombre de pièces détachées convoyées dans la cale 2 (en dehors du sas).