

Rapport Recherche Opérationnelle

Tom Humbertclaude, Ethan Mocellin, Robin Rouet

Mai 2023



Projet de Recherche opérationnelle Département Informatique
S6 - Année 2022/2023

1 Première modélisation

Programme linéaire

Le premier modèle consiste en une optimisation des coûts de production d'une entreprise. Pour ce faire, on dispose des données suivantes :

- n le nombre de périodes
- d_i la demande sur la période i
- c_i le coût de production d'un seul produit sur la période i
- f_i le coût d'installation des machines payé au début de la période i si la production est lancée sur la période i
- h le coût de stockage des produits en surplus à la fin de la période i

La modélisation a été faite en utilisant les variables et la constante (en lettres capitales) suivantes :

- y_i une variable binaire qui prend la valeur 1 si la production est lancée sur la période i et 0 sinon
- x_i une variable désignant la quantité produite pendant la période i
- s_i une variable représentant la quantité de produits stockée pendant la période i

dans ce PLNE, la constante M correspond aux demandes pour toutes les périodes restantes, comme le définit la première ligne des contraintes. La deuxième ligne indique que le nombre de produits fabriqués ne doit pas dépasser la demande restante M . La troisième ligne indique que le stock de la période passée et la production de la période courante doit être supérieure ou égale à la demande du mois en cours. Enfin la quatrième contrainte dit que le stock de la fin de la période courante est égal au stock avant additionné à la production moins la demande.

Voici donc le PLNE associé à ce modèle :

$$\begin{aligned} \min \quad & \sum_{i=1}^n (c_i \cdot x_i + f_i \cdot y_i + s_i \cdot h) \\ & x_i \leq y_i * M \\ & M = \sum_{j=i}^n d_j \\ & x_i + s_{i-1} \geq d_i \\ & s_i = x_i + s_{i-1} - d_i \\ & \forall i \in \llbracket 1, n \rrbracket, x_i, s_i, d_i, c_i, f_i, h \in \mathbb{N}, y_i \in \{0, 1\} \end{aligned}$$

Résultat de l'exécution

Avec l'instances des jouets, on obtient donc une valeur de la fonction objectif de 1788. Voici les données qui ont été calculées sur la production prévue :

Production	Besoin	Stock	Bool	Coût	
1	70	30	40	1	5
2	0	25	15	0	3
3	0	15	0	0	4
4	106	47	59	1	5
5	0	34	25	0	6
6	0	10	15	0	3
7	0	15	0	0	4

Calculs

Les calculs sur les instances ont été limités à 10 secondes par instance car étrangement le processus s'arrêtait automatiquement au bout d'un certain temps sinon. Voici donc les résultats :

instance	relax. lin.	statut	solution	ecart	nœuds	temps
60.4	31131	FEASIBLE	31131	0	16091	10
120.8	74724	FEASIBLE	74678	0.06	1038	10
60.8	27962	FEASIBLE	28049	0.31	22628	10
120.10	86160	FEASIBLE	92014	6.36	843	10
120.1	77204	FEASIBLE	77865	0.84	859	10
60.1	29739	FEASIBLE	29739	0	35895	10
120.7	83335	FEASIBLE	84789	1.71	700	10
120.5	97306	FEASIBLE	101110	3.76	832	10
120.3	87748	FEASIBLE	99830	12.10	861	10
60.7	30927	FEASIBLE	31037	0.35	36515	10
90.1	51426	FEASIBLE	52276	1.62	942	10
90.8	50181	FEASIBLE	51792	3.11	810	10
90.5	64980	FEASIBLE	67655	3.95	1065	10
90.7	55601	FEASIBLE	58154	4.39	1208	10
60.9	35492	FEASIBLE	35658	0.46	24233	10
Toy_Instance	1788	OPTIMAL	1788	0	0	0
90.6	41967	FEASIBLE	42793	1.93	703	10
60.10	31809	FEASIBLE	31809	0	28455	10
90.2	46703	FEASIBLE	46948	0.52	647	10
90.4	54205	FEASIBLE	54063	0.26	1146	10
120.6	70111	FEASIBLE	78947	11.19	757	10
60.2	27572	OPTIMAL	27572	0	591	7
90.10	57435	FEASIBLE	59178	2.94	863	10
90.3	59612	FEASIBLE	59347	0.44	963	10
60.3	34081	FEASIBLE	34081	0	41539	10
120.4	89088	FEASIBLE	88328	0.86	630	10
120.2	75486	FEASIBLE	69104	9.23	857	10
90.9	60072	FEASIBLE	59722	0.58	748	10
60.5	35693	FEASIBLE	35693	0	26404	10
60.6	25186	FEASIBLE	25186	0	16923	10
120.9	90799	FEASIBLE	95067	4.48	815	10
21.1	13068	OPTIMAL	13068	0	4	0

2 Seconde modélisation

Programme linéaire

Le deuxième modèle porte sur une suppression de la nécessité du stock à la fin du mois. Ainsi, on peut modéliser avec le PLNE suivant :

$$\min \sum_{i=1}^n \left(\sum_{j=i}^n ((c_i + h \cdot (j - i)) \cdot x_{ij} * d_j) + f_i \cdot y_i \right)$$

$$\begin{aligned}
c.s. \quad & \sum_{i=1}^n x_{ij} = 1 \\
& \sum_{j=1}^n (x_{ij}) \leq n - \sum_{k=0}^i \left(\sum_{j=0}^n (x_{kj}) \right) \\
& x_{ij} \leq y_i \\
& x_{ij} = 0, i > j \\
& \forall i, j \in \llbracket 1, n \rrbracket, d_i, c_i, f_i \in \mathbb{N}, y_i, x_{ij} \in \{0, 1\}
\end{aligned}$$

Le modèle est différent dans ce cas, où il n'y a plus de stock. De plus, la production de produits durant la période i a été changée et dépend maintenant de la demande en j .

La première contrainte indique que x_{ij} vaut une fois 1 et le reste du temps 0. En effet, on produit tous les produits pour la période j en une seule fois. Les deuxième et troisième contraintes sont les mêmes que dans le modèle 1.

Résultat de l'exécution

Toujours avec l'instance des jouets, on obtient également une valeur optimale de la fonction objectif de 1788. Voici les données qui ont été calculées sur la production prévue :

	Besoin	Stock	Bool	Coût
1	30	40	1	5
2	25	15	0	3
3	15	0	0	4
4	47	59	1	5
5	34	25	0	6
6	10	15	0	3
7	15	0	0	4

Production :

Mois	Période 1	Période 2	Période 3	Période 4	Période 5	Période 6	Période 7
1	1	1	1	0	0	0	0
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	0	0	0	1	1	1	1
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0

On remarque ainsi que la production a eu lieu seulement pendant le jour 1 et 4.

Calculs

On effectue les mêmes calculs sur toutes les instances comme pour le modèle 1, tout en limitant le temps de calcul à 10 secondes. Voici les résultats obtenus :

instance	relax. lin.	statut	solution	ecart	nœuds	temps (en s)
90.6	41811.0	OPTIMAL	41811.0	0	0	1.1062734127044678
90.2	46518.0	OPTIMAL	46518.0	0	0	1.0507445335388184
120.3	86778.0	OPTIMAL	86778.0	0	0	2.9003145694732666
120.5	96316.0	OPTIMAL	96316.0	0	0	2.9048855304718018
60.5	35693.0	OPTIMAL	35693.0	0	0	0.4016075134277344
60.7	30853.0	OPTIMAL	30853.0	0	0	0.39008545875549316
120.4	82367.0	OPTIMAL	82367.0	0	0	2.7695417404174805
90.8	49010.0	OPTIMAL	49010.0	0	0	1.155306100845337
60.2	27572.0	OPTIMAL	27572.0	0	0	0.36879587173461914
120.7	81866.0	OPTIMAL	81866.0	0	0	2.476663589477539
90.1	50943.0	OPTIMAL	50943.0	0	0	1.2033770084381104
120.8	70734.0	OPTIMAL	70734.0	0	0	2.6091115474700928
60.10	31809.0	OPTIMAL	31809.0	0	0	0.37151336669921875
60.4	31131.0	OPTIMAL	31131.0	0	0	0.39945268630981445
120.2	67630.0	OPTIMAL	67630.0	0	0	2.416717529296875
120.1	75417.0	OPTIMAL	75417.0	0	0	2.497011423110962
90.10	56514.0	OPTIMAL	56514.0	0	0	1.0547146797180176
60.6	25186.0	OPTIMAL	25186.0	0	0	0.3529515266418457
120.9	87909.0	OPTIMAL	87909.0	0	0	2.77036452293396
90.9	59424.0	OPTIMAL	59424.0	0	0	1.1066889762878418
21.1	13068.0	OPTIMAL	13068.0	0	0	0.0623321533203125
60.8	27962.0	OPTIMAL	27962.0	0	0	0.37633347511291504
120.10	85103.0	OPTIMAL	85103.0	0	0	2.4737844467163086
120.6	65704.0	OPTIMAL	65704.0	0	0	2.5305581092834473
90.4	53897.0	OPTIMAL	53897.0	0	0	1.0663700103759766
90.3	57613.0	OPTIMAL	57613.0	0	0	1.116053581237793
90.5	64123.0	OPTIMAL	64123.0	0	0	1.1001217365264893
Toy_Instance	1788.0	OPTIMAL	1788.0	0	0	0.007040977478027344
60.9	35492.0	OPTIMAL	35492.0	0	0	0.35979413986206055
60.1	29739.0	OPTIMAL	29739.0	0	0	0.3415348529815674
90.7	54913.0	OPTIMAL	54913.0	0	0	1.0832581520080566
60.3	34081.0	OPTIMAL	34081.0	0	0	0.38583993911743164

On remarque clairement qu'aucun des calculs des instances n'a pris plus de 3 secondes, contrairement aux résultats précédents qui eux, dépassent les 10 secondes.

3 Analyse des résultats

Pour ce projet, deux modélisation du problème ont été effectuées. La première utilise 3 variables, dont une gérant le stockage des produits. La seconde quant à elle ne comporte que 2 variables binaires, et suppose qu'il n'y a pas de limite sur la quantité produite chaque mois, permettant ainsi de supprimer la variable précédente. Chacune est supposée donner une solution optimale.

Dans le cadre de ce projet, le temps d'exécution est limité à 10 secondes pour des soucis liés au système des machines utilisées.

Pour la première modélisation, on constate qu'elle ne donne que très rarement une solution optimale dans l'intervalle de temps donné, et affiche un écart entre le résultat avec relaxation linéaire. De surcroît, ne nombreux nœuds de l'arbre de branchements sont impliqués dans les calculs.

La seconde modélisation cependant, donne pour chaque instance une solution optimale et ce dans un temps relativement court (moins de 3 secondes). De plus, elle ne sollicite pas de nœud de l'arbre de branchement et ne présente aucun écart entre la durée d'exécution avec relaxation linéaire ou sans grâce aux variables binaires.

Ainsi, il apparaît donc que la modélisation 2 est bien plus efficace que la 1, que ce soit en terme de temps d'exécution ou bien pour obtenir une solution optimale.