

Analyse des exercices

25/03/20

Principe général

- Lire 2 fois l'énoncé pour être sûr d'avoir tout saisi.
(La deuxième fois, on sait qu'elle est la question sur ce que l'on optimise et l'on note donc au passage les éléments importants)
- Relever les différents concepts sous jacents à l'énoncé. Que manipule t'on ?
Que quantifie t'on (les **données** chiffrées sont souvent des indices données des proportions aux concepts manipulés) ? Quelles sont les bornes ?
- Construire à la main une solution pragmatiquement et naïvement (on ne cherche pas l'optimum mais une solution réalisable) avec une stratégie déterministe et réexplicable (par exemple je prend le premier et je remplis au maximum, j'affecte dans l'ordre croissant, ...). Analyser quelles **décisions** sont prises pour construire cette solution.
- Corriger au fur et à mesure quand l'algorithme ne marche pas (il existe quelquepart des **contraintes** qui empêche l'algorithme de faire ce qu'il veut).
- Comptabiliser le score de la solution (prémisse de la fonction **objectif** à optimiser)
- Désormais, tout est prêt pour appliquer le protocole :
 - Données/Variables-Domains/Contraintes/Objectif

Le Stade (question 1)

- Données :
 - Tâche d'indice i
 - D_i Durée de la tâche i
 - $P_{i,j}$ lien de précédence entre la tâche i et la tâche j
- On nous demande la durée minimale du chantier
- Une tâche ne peut commencer que lorsque *toutes* celles qui le précèdent sont terminées
- A priori, un graphe orienté représentant les précédences devrait nous aider à modéliser le problème
- Indice : le chantier sera fini quand TOUTES les tâches seront finies

Le Stade (question 2)

- Données supplémentaires :
 - P Prime de 30K€/semaine
 - R_i Réduction maximale (en semaines) de la durée d'une tâche en cas d'embauche d'ouvriers supplémentaire
 - C_i Coût supplémentaire PAR SEMAINE en € si embauche d'ouvriers supp.
- Attention, l'objectif n'est plus uniquement de réduire la durée du projet. Comment le réécrire ?
- Il ne sert à rien de recruter des ouvriers supplémentaires, si ceux ci coûtent plus chers que la prime FINALE.
- Il ne sert à rien de réduire une durée de tâche si celle ci n'a pas d'impact sur la tâche suivante et donc n'induit pas une réduction de la durée du projet.

Rail Route

- Données :
 - Q 180 tonnes de produits
 - E l'ensemble des Entrepôts (d'indice i)
 - C l'ensemble des Centres de retraitement (d'indice j)
 - K_i la capacité de l'entrepôt i
 - $RA_{i,j}$ le coût de transport entre l'entrepôt i et le centre j via le Rail
 - $RO_{i,j}$ le coût de transport entre l'entrepôt i et le centre j via la Route
 - MinRA, MaxRA les quantités minimales et maximales autorisées sur le Rail par la SCNF
- On nous demande COMMENT acheminer les produits en minimisant les coûts.
- Tous les chemins entre i et j ne sont pas possibles. Comment modifier les données pour *guider* le solveur (qui minimise les coûts) en l'empêchant de trouver des solutions interdites.
- A priori, un graphe en deux couches (Entrepôts, Centres) devrait suffire à visualiser la problématique
- Indice : 180 tonnes de produits à livrer alors que $50+40+35+65=190$ tonnes sont disponibles dans les entrepôts. Il ne faut pas utiliser des contraintes '=' mais plutôt ' \leq ' car tout ne doit pas être transporté.

Constitution d'équipage (question 1)

- Données :
 - Pilote d'indice i
 - Langue d'indice j
 - Appareil d'indice k
 - $N_{i,j}$ Niveau de langue j du pilote i
 - $M_{i,k}$ Maîtrise de l'appareil k du pilote i
- On nous demande si tous les pilotes peuvent voler, c'est à dire est ce qu'il est possible de faire 4 couples de pilotes compatibles.
Deux pilotes sont compatibles si ils ont tous les deux la moyenne dans au moins une langue commune.
- On doit pouvoir construire un graphe (dit de compatibilité) connectant des nœuds (pilotes) si ceux-ci ont la moyenne sur une des langues.
- On cherche un *couplage* sur ce graphe, c'est à dire que chaque nœud (pilote) doit être relié à UN nœud. L'arête ainsi sélectionnée représentera l'équipage.
- Qu'est ce que l'on maximise sur ce graphe ?

Constitution d'équipage (question 2)

- Données supplémentaires :
 - S_{i_1, i_2} Score entre le pilote i_1 et le pilote i_2
- On cherche désormais le couplage de score total maximum
- Notre graphe possède désormais un poids sur ses arêtes, c'est le score de l'équipage.
- Attention, le score se calcule avec un MAX. Celui ci n'est pas LINEAIRE ... va t'on pouvoir exprimer ce problème en Programmation Linéaire ?