Rapport technique

AGUSTO Rold PAUL Hugo TRANG Hoang Phong Vu
SUGANATHASIVAM Vathanalakshan

November 9, 2020

1 Intoduction

Dans le cadre de ce projet,

2 Problème du voyageur de commerce déterministe

Afin de pouvoir traiter le problème du voyageur de commerce stochastique, nous partons de l'origine avec l'approche déterministe. Dans cette approche, les distances entre les villes sont connues avant que le commerçant commence le voyage. Plus concrètement, soit:

- G = (V, E) un graphe orienté ou non-orienté complet avec :
  - V un ensemble de sommets
  - $E \subseteq \{(x, y)|(x, y) \in V^2 \wedge x \neq y\}$ un ensemble d'arêtes, qui sont des couples de sommets distincts
- c_{ij} le coût pour aller du sommet v_i au sommet v_j
- x_{ij} la variable de décision. elle vaut 1 s'il existe un arc (v_i, v_j), 0 sinon. On peut l'interpréter comme une matrice de passage.

Le problème du voyageur de commerce peut être écrit sous forme problème d'optimisation :

$$\min_{x} \quad \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{1a}$$

s.t.

$$\sum_{j=1, i\neq j}^{n} x_{ij} = 1, \qquad i = 1, ..., \tag{1b}$$

$$\sum_{i=1, i\neq j}^{n} x_{ij} = 1, \qquad j = 1, ..., \tag{1c}$$

$$\sum_{i|v_i \in S}\sum_{j|v_j \in S} x_{ij} \leq |S| - 1 \quad S \subset V \mathrm{et} S \neq \varnothing, \tag{1d}$$

$$x_{ij} \in \{0, 1\} \qquad 1 \leq i, j \leq n \tag{1e}$$

Littéralement, le voyageur doit visiter toutes les villes (connues dans le graphe) tel que la distance qu'il parcourt est minimale et que pour chaque ville il ne peut visiter qu'une seule fois. Plus concrètement, la fonction objectif sert à trouver la distance minimale. La contrainte 1b veut dire : après avoir visté une ville v_i, le voyageur ne peut visiter qu'une ville ensuite. Avec la contrainte 1c, lors d'une visite d'une ville v_j, le voyageur doit provenir d'une seule ville.
En comparant avec les conditions littérales qu'on a prises en compte précédémment, les deux premières contraintes sont suffisantes. Néanmoins, les algorithmes d'optimisation des solveurs peuvent génerer des subtours (i.e. sous chemins). Toutes les villes sont visitées une seule fois et la distance totale est minimale, mais le fait d'avoir plusieurs chemins discrètes veut dire que le voyageur a le droit de téléporter d'une ville en une autre ville et ce n'est pas le cas. La contrainte 1d est donc rajoutée au modèle pour les éliminer, elle est appelée la contrainte d'élimination des subtours.

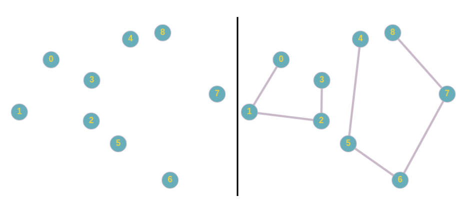

Figure 1: Une solution où existe plusieurs subtours (sous-chemins) (Attention, il s'agit d'un exemple pour montrer la conséquence d'une manque de la contrainte 1e, elle n'est donc pas une solution optimale)

Une approche possible pour ce modèle est de lancer l'optimisation avec les contraintes 1b, 1c, 1e plusieurs fois jusqu'à ce qu'on trouve la solution optimale.

C'est le cas où il n'existe qu'un seul subtour. La démarche est exprimée dans le pseudo-code ci-dessous :

```
Data: graphe G=(V, E)
Result: La solution optimale du problème du voyageur de commerce
while il existe plusieurs subtours dans la solution do
    lancer l'optimisation avec les contrainte 1b 1c 1e;
    if understand then
        go to next section;
        current section becomes this one;
    else
        go back to the beginning of current section;
    end
end
```

Algorithm 1: Trouver la solution optimale du problème du voyageur de commerce