

# DM

## Rapport Optimisation

### 2.2 Petites instances du problème : recherche explicite

- Comment représentez-vous une solution au problème ?
- Supposons que l'on ait  $N$  ingrédients disponibles au total. Quelle est la taille de l'espace des solutions (recettes de pizza) possibles ?
- Sachant que  $N < 10$  pour les problèmes A, B et C, proposez une méthode exacte.
- Donnez une estimation du temps que cette méthode mettrait pour s'exécuter sur l'instance E<sup>4</sup>.

#### Espace de solution:

La taille de l'espace de solution est de  $2^N - 1$ .

#### Méthode exacte:

On calcule le nombre de recettes possibles.

Pour chaque recette possible, on regarde pour chaque client s'il est satisfait par la pizza. S'il est satisfait, on augmente le nombre de clients satisfaits de cette recette de 1. Puis, on compare si la recette est meilleure que celle en mémoire.

A la fin de la 2nd boucle, on renvoie donc la meilleure solution.

## Algorithme:

N = nombre d'ingrédients

X = nombre de recettes possibles ( $2^N - 1$ ).

Y = nombre de clients

resX = nombre de clients satisfaits par X.

R = Recette optimale

RNbClient = Nombre de client de satisfait pour la recette optimale

RNbClient = 0

Pour chaque X faire:

    resX = 0

    Pour chaque Y faire:

        C = 1

        nb = 0

        Pour chaque ingrédient i dans Y.like:

            Si i est dans X.ingrédient:

                nb = nb + 1

        FinSi

    FinPour

    Si taille de nb = taille de Y.like alors:

        C = 1

    FinSi

    Pour chaque ingrédient i dans Y.dislike faire:

        Si i est dans X.ingrédients alors:

            C = 0

    FinSi

    FinPour

    Si taille X.ingrédients < taille Y.like:

        C = 0

    FinSi

    resX += C

FinPour

Si RNbClient < resX alors:

    R = X

    RNbClient = resX

FinSi

FinPour

Retourner R

### Estimation du temps pour E:

En reprenant les mêmes variables que précédemment on a:

$$Y = 4\,986$$

$$X = 2^{10\,000} - 1$$

$$N = 10\,000$$

On boucle sur X et Y, puis on effectue les vérifications.

Les deux boucles représentent donc  $X * Y$  opération.

Les vérifications sont constituées de 2 boucles pour de taille N et de 3 opérations.

On aurait donc un total de:  $(2^{10000} - 1 \times 4986) \times (N + N + 3)$  opérations.

En partant du principe qu'un ordinateur effectue  $10^9$  opérations par seconde, il faudrait environ:

$$1,98 \times 10^{3009} \text{ secondes pour effectuer cette méthode sur E.}$$

### 2.3.1 Algorithme Génétique

- Quelle est la taille de votre population ?
- Comment définissez-vous l'opérateur de croisement ?
- Comment définissez-vous l'opérateur de mutation ? Quelle probabilité de mutation donne les meilleurs résultats ?
- Que choisissez-vous comme critère d'arrêt ?
- Précisez toute information au sujet de votre algorithme qui vous paraîtrait pertinente dans le rapport

**Taille de la population:** On génère XXX recettes de départ.

**Opérateur de croisement:** On prend X% d'une recette et Y% d'une autre

**Opérateur de mutation:** On prends une recette et on ajoute ou retire aléatoire un ingrédient

**Opérateur de sélection:** Garder les N meilleures recettes

**Critère d'arrêt:** Le nombre de clients satisfaits des N meilleures recettes n'a pas augmenté depuis un certain nombre de générations.

#### Information diverses:

On peut ajouter un critère de mutation:

Calculer la fréquence d'apparition des ingrédients dans les recettes aimées et détestées et modifier les mutations en conséquence. On peut se baser sur les mutations génétiques: Les gènes qui permettent à l'espèce de mieux survivre/se reproduire ont plus de chance de traverser les générations.

Ici un ingrédient qui est souvent aimé et peu détesté a donc plus de chance de donner un meilleur résultat. Et à l'inverse, un ingrédient peu aimé et beaucoup détesté devrait disparaître.

#### Les paramètres:

Il faut jouer avec les paramètres (Taille de population, %de chance de mutation, %de sélection) pour trouver les valeurs optimales. Une idée serait de lancer plusieurs fois l'algorithme en ne changeant qu'un seul critère et de garder celui avec le meilleur résultat. Si on effectue cela pour la Taille, le %de mutation et de sélection on obtiendra de meilleur résultat.

Cependant l'implémentation de cet algorithme étant extrêmement lente, il est difficile de le mettre en place. On peut cependant remarquer que plus la population de base est élevée, meilleure est la solution mais l'exécution devient également plus lente. En ce qui concerne les mutations, il faut qu'elle soit assez élevée (En dessous de 0.5, nous n'avons quasiment aucun changement).

### 2.3.2 Recuit Simulé

- Quels mouvements entre solutions autorisez-vous ?
- Quelle est la valeur de départ de la température ?
- Comment décroît la température au cours du temps ?
- Quel est le critère d'arrêt ?
- Précisez toute information au sujet de votre algorithme qui vous paraîtrait pertinente dans le rapport

On autorisera des mouvements entre solutions en calculant le critère d'acceptation de Metropolis. On accepte la variation si elle augmente l'objectif ou si elle ne le diminue pas trop.

conditions d'acceptation :  $\text{rand}() < \exp(|\Delta| / t_{\text{current}})$  ou  $\Delta = \text{candidate\_eval} - \text{curr\_eval} < 0$

La première étape consiste à calculer la différence entre l'évaluation par la fonction objective de la solution courante et la solution voisine sur laquelle on travaille actuellement.

- valeur de départ de la Température : 100 (arbitraire)
- la température décroît au cours du temps
- critère d'arrêt : la température courante est  $< 1$ . On évite ainsi les valeurs négatives.

Le recuit simulé ne peut garantir en pratique de trouver la solution globalement optimale, mais il produit une "bonne" solution à un problème d'optimisation.

- scores :
  - D -> 1533 < 1750      en 03:41
  - E -> 856 << 2000      en 1h08:54

### 2.3.3 Recherche Tabou

- Quels mouvements entre solutions autorisez-vous ?
- Quel temps choisir pour les éléments dans la liste tabou ?
- Quel est le critère d'arrêt ?
- Précisez toute information au sujet de votre algorithme qui vous paraîtrait pertinente dans le rapport

**Mouvements autorisé:** Ajouter ou retirer un ingrédient

**Temps choisi:**

**Critère d'arrêt:** Le nombre de client satisfait n'augmente plus (ou que très peu) sur les X dernières itérations