Exercice d'algorithmique

nicolas.audebert@onera.fr

Mercredi 28 février 2018

Exposé du problème

Considérons un *speed meeting* durant lequel 10 personnes doivent échanger chacune avec les autres. Un échange dure 5 minutes et se réalise en tête à tête. Chaque personne doit avec l'opportunité de discuter **seule** pendant 5 minutes avec chaque autre personne du groupe. On cherche à déterminer un arrangement minimisant la durée totale de l'événement.

Par exemple, l'arrangement suivant :

- t+0 à t+5 minutes : se rencontrent 1 et 9, 2 et 4, 3 et 7, 5 et 8, 6 et 10.
- t+5 à t+10 minutes : se rencontrent 1 et 2, 3 et 10, 4 et 5, 6 et 7, 8 et 9.
- t+10 à t+15 minutes : se rencontrent 1 et 5, 2 et 6, 4 et 9, 8 et 10.
- t+15 à t+20 minutes : se rencontrent 1 et 8, 2 et 7, 3 et 6, 9 et 10.
- t+20 à t+25 minutes : se rencontrent 1 et 4, 2 et 3, 5 et 9, 6 et 8, 7 et 10.
- t+25 à t+30 minutes : se rencontrent 1 et 10, 4 et 8, 5 et 6, 7 et 9.
- t+30 à t+35 minutes : se rencontrent 1 et 7, 2 et 9, 3 et 8, 4 et 6, 5 et 10.
- t+35 à t+40 minutes : se rencontrent 1 et 6, 2 et 8, 3 et 9, 4 et 10, 5 et 7.
- t+40 à t+45 minutes : se rencontrent 1 et 3, 2 et 10, 4 et 7, 6 et 9.
- t+45 à t+50 minutes : se rencontrent 2 et 5, 3 et 4, 7 et 8.
- t+50 à t+55 minutes : se rencontrent 3 et 5.

nécessite 11 étapes, soit **55 minutes**.

Question En pratique, il existe **45 binômes** et il est possible de réaliser **5 binômes** durant chaque intervalle de 5 minutes. Au mieux, un arrangement prend donc **9 étapes**. Exhiber un tel arrangement.

Formalisation

Dans le cas général, on considère un ensemble E à n éléments, que l'on assimilera à $\{1 \dots n\}$. On $C = \{(p,q) \in E^2/p > q\}$ l'ensemble des binômes non-redondants, de cardinal $|C| = \frac{(n-1)n}{2}$.

Il s'agit ici de trouver le nombre minimal k de sous-ensembles C_i tels que :

$$\bigcup_{i=1}^{k} C_i = C$$

et

$$\forall i \in \{1 \dots k\} \quad (p,q) \in C_i \Rightarrow \forall (p',q') \in C_i \;\; p' \neq p, p' \neq q, q' \neq p, q' \neq q \quad .$$

Autrement dit, on cherche une collection de sous-ensembles C_i tels que :

- leur union est égal à l'ensemble des couples possibles ("tous les participants ont échangé avec tous les autres"),
- et si un couple apparaît dans un des sous-ensembles, alors aucun de ses membres n'apparaît dans un autre binôme de ce même sous-ensemble ("on ne peut participer qu'à une seule discussion à la fois").

Note 1 On se rend bien compte que pour que k soit minimal, il faut interdire les redondances : un participant ne doit discuter qu'une et une seule fois avec chacun des autres. Formellement dit, $(p,q) \in C_i \Rightarrow (p,q) \notin C_j \;\; \forall j \neq i$.

Si n est pair, alors on peut réaliser au maximum $\frac{n}{2}$ couples dans chaque C_i. Comme :

$$|C| \leq \sum_{i=1}^{k} |C_i|$$

et que

$$|C_i| \leq n/2 \quad ,$$

il vient :

$$k\frac{n}{2} \geq \frac{n(n-1)}{2} \quad ,$$

c'est-à-dire

$$k \geq n-1 \quad .$$

La question est donc de trouver une méthode constructive qui permet d'atteindre cet optimal de $n-1$ étapes. On suppose n pair pour le reste du problème. On peut facilement résoudre le problème pour n impair en ajoutant un élément "fantôme" et en l'enlevant *a posteriori*. Dans le cas n impair, l'optimal est donc de n.

Question (facile) Pour se forger une intuition, il est recommandé de commencer par trouver des arrangements optimaux pour des petites valeurs de n, par exemple 4, 6 et 8.

Question (facile) Proposer un algorithme qui, en partant d'un arrangement quelconque, vérifie que la solution est répond bien au problème posé. Quel est sa complexité (polynomial ou non-polynomial) ?

Question (facile) Proposer un algorithme permettant d'obtenir un arrangement quelconque.

Indice : l'arrangement n'a pas à être optimal.

Question (modérée) Proposer un algorithme permettant d'obtenir un arrangement optimal, pour n relativement faible ($n \leq 20$).

Indice : on ne se préoccupe pas de complexité ici, mais simplement de trouver un résultat !

Question (difficile) Proposer un algorithme permettant d'obtenir un arrangement optimal, même pour n très grand. Prouver son exactitude.

Question (difficile) Même question en utilisant un algorithme récursif.

Indice : on ne peut pas directement résoudre le problème à n éléments à partir de la solution à n éléments. En revanche, si on connaît une solution $n/2$ éléments, on peut chercher à construire une solution n éléments. On peut commencer par les cas où n est une puissance de 2.