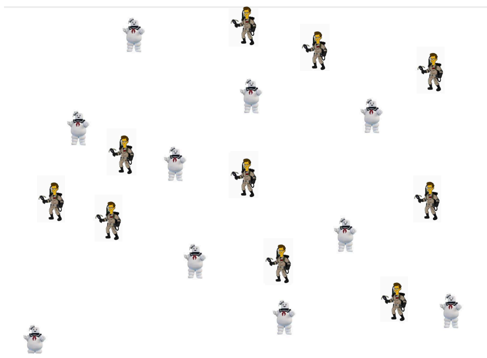
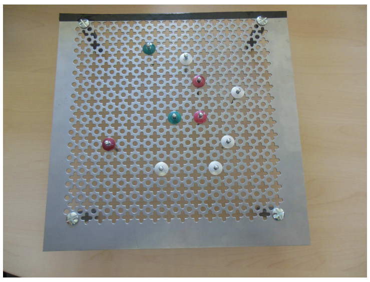
Auteur : Jean-Marc.Vincent@imag.fr
Université Grenoble-Alpes, UFR IM²AG, Inria Rhône-Alpes, IREM-Grenoble

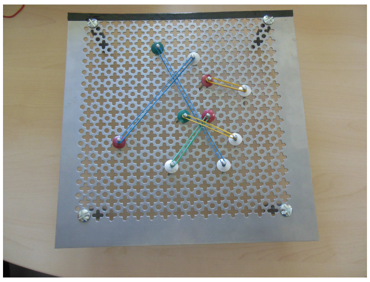
> *There's something very important I forgot to tell you ! Don't cross the streams. . . It would be bad. . .Try to imagine all life as you know it stopping instantaneously and every molecule in your body exploding at the speed of light.* Dr. Egon Spengler

Les chasseurs de fantômes Ghostbusters utilisent des rayons pour neutraliser les fantômes. Comme l'indique la citation ci-dessus, il est primordial que ces rayons ne se croisent jamais. Imaginons dix fantômes chassés par dix Ghostbusters. Ces derniers peuvent-ils se répartir les fantômes afin que chacun tire sur un fantôme sans qu'aucun rayon ne se croise. [1]

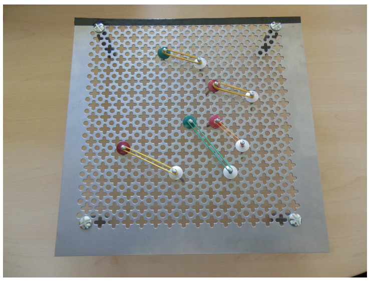




Ronds blancs : chasseurs,
Ronds colorés : fantômes.



Une mauvaise solution des lasers se croisent.



Une bonne solution, le monde ne va pas exploser.

1. Merci à Romain Joly et Math en Jeans

Existence d'une solution

Il y a autant de fantômes que de chasseurs. Peut-on toujours trouver une association de chaque chasseur avec un fantôme sans que les *lasers* s'entrecroisent ?

Activité 1 : on bricole pour dégrossir le problème
On note n le nombre de chasseurs, n est aussi le nombre de fantômes. Dans un premier temps on peut chercher avec un n significatif ($n = 10$ par exemple). Avec un peu d'efforts on trouve une solution. Peut-on bouger des fantômes/chasseurs de manière à *empêcher* la solution ?
On peut restreindre le problème, que se passe-t-il pour des petites valeurs de n ? On pourra essayer avec $n = 1$, $n = 2$, $n = 3$...

Indication 1
Le problème survient lorsque les points fantômes et les chasseurs sont alignés.

Heuristiques et contre-exemples

L'objectif est de trouver une méthode qui permette de résoudre le problème.

Activité 2 : contre-exemple
On joue à deux, un propose un algorithme et l'autre cherche un contre-exemple réalisable tel que l'algorithme ne donne pas la solution.

Indication 2
Exemples d'idées d'heuristiques : on choisit le fantôme et le chasseur les plus proches, on joint un couple fantôme/chasseur sur la périphérie (sur l'enveloppe convexe) et on recommence avec le reste, on choisit le fantôme/chasseur les plus bas, ...

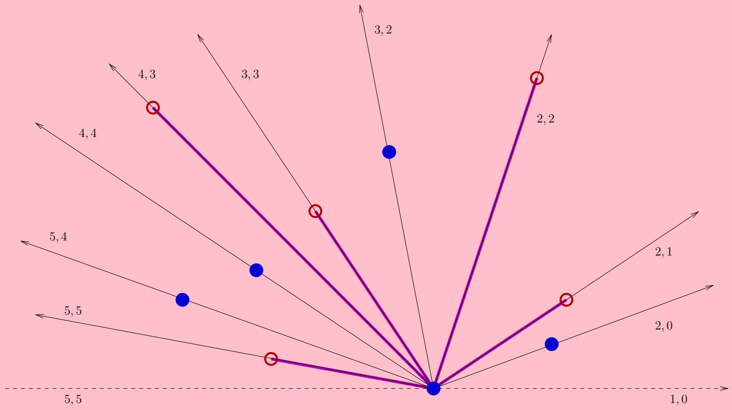
Diviser pour régner

À partir des heuristiques, on sent bien que si on réussit à se ramener à un problème plus petit on pourra raisonner par induction. Or le passage de n à $n-1$ parait difficile cf contre-exemples, il faudrait donc découper en **plusieurs** sous-problèmes. Pour éviter les cas spéciaux on suppose que les points sont en position générale (pas d'alignements), cette hypothèse pourra être affaiblie par la suite.

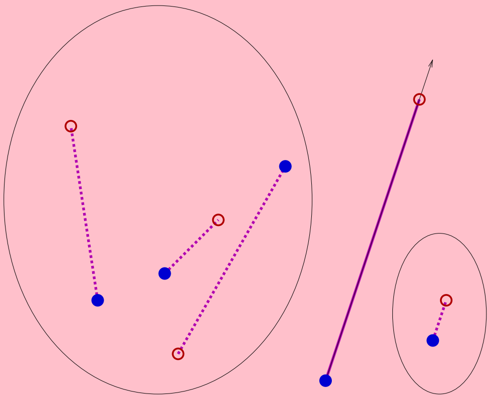
Activité 3 : un algorithme magistral
Trouver un couple fantôme/chasseur qui sépare convenablement le nuage des $2(n-1)$ points restants.

Indication 3
On choisit un point arbitrairement, un fantôme par exemple, et on parcourt tous les chasseurs. Pour chaque chasseur on trace la droite entre le chasseur et le fantôme et on compte le nombre de fantômes et de chasseurs en dessous de la droite (au passage définir le "dessous"). Quel point peut-on choisir pour simplifier l'analyse ?

Éléments de solution 3



Un point d'ordonnée minimale étant choisi, on suppose que c'est un chasseur (si ce n'est pas le cas échanger le rôle chasseur et fantôme). On trie l'ensemble des autres points (n fantômes et $n-1$ chasseurs) par angle croissant, l'angle étant défini entre le segment et l'axe horizontal orienté. On parcourt les points par angle croissant et on calcule le nombre de fantômes/chasseurs dans le secteur angulaire entre la droite et l'axe horizontal (l'orientation de la droite étant fixée par le segment).
Le nombre de chasseurs/fantômes va passer de $1, 0$ à n, n lorsque tout le demi-plan au-dessus de l'axe horizontal sera balayé. À chaque étapes du parcourt, on ajoute 1 soit au compte de chasseurs, soit au compte de fantômes. Il y aura donc une première étape pour laquelle le compte de chasseurs et de fantômes sera égal. À cette étape on sera sur un segment associant un chasseur à un fantôme, sinon ce ne serait pas la première fois.
La droite contenant le segment sépare le nuage de points en 3 parties. Les 2 points (chassuer/fantôme sur la droite), un nombre égal de chasseurs/fantômes dans le secteur angulaire entre la droite et l'axe horizontal et un nombre égal de chasseurs/fantômes dans le secteur complémentaire.
Le problème est donc réduit à deux sous-problèmes de même nature dans les 2 secteurs, ces 2 sous-problèmes sont de taille strictement plus petite, on applique donc notre algorithme récursivement sur ces deux sous-problèmes. La droite étant séparatrice, on est certain que les segments construits des deux cotés ne pourront pas s'intersecter, donc leur réunion va former une solution au problème global.



Le principe utilisé dans cet algorithme est celui du **diviser pour régner**. On divise le problème en sous-problèmes de même nature mais de taille plus petite, on résout les sous-problèmes et on compose les solutions obtenues pour obtenir la solution au problème initial. Ici la taille des sous-problèmes dépend de la valeur des données et donc sont de taille qui n'est pas connue à l'avance. L'algorithme est très proche de l'algorithme de tri rapide (ou tri par segmentation, *Quicksort*). Cette approche a été extraite d'un exercice de Cormen, Leiserson, Rivest & Stein (2010).

Variant d'itération

Comme l'objectif est de n'avoir aucun segments qui se croisent, une idée intuitive est, lorsque l'on observe un croisement de décroiser les segments qui se croisent.

Activité 4 : un algorithme magique
Choisir une association chasseurs/fantômes arbitrairement. Tant qu'il existe des segments qui se croisent, les décroiser. Essayer sur des exemples, essayer de trouver des contre-exemples, observer que décroiser ne diminue pas forcément le nombre de croisements (au fait combien peut-on en avoir ?). Trouver un argument pour prouver que l'algorithme termine en un nombre fini d'étapes.

Indication 4
Dans un quadrilatère convexe non dégénéré, les diagonales sont toujours plus grandes que les cotés.

Éléments de solution 4
Après un choix arbitraire d'affectation, l'algorithme consiste en une itération simple : tant qu'il existe un croisement entre 2 segments, décroiser les segments.
Pour prouver la correction de cet algorithme on va trouver un *variant* de l'itération, c'est à dire une fonction des valeurs des variables qui décroisse strictement. Le nombre d'intersections n'est pas un variant de l'itération car il peut augmenter (exemples). Par contre la somme des longueur des segments diminue. En effet, la somme des longueurs des 2 diagonales d'un quadrilatère non dégénéré est strictement supérieure à la somme des longueurs de 2 cotés opposés.
La somme des longueurs est donc une fonction strictement décroissante à chaque décroisement. Comme le nombre d'associations est fini, le nombre d'itérations sera fini donc l'algorithme se termine et il n'y a plus de croisements.

Réduction

Ce problème d'association pourrait-il simplement être un cas particulier d'un problème général que l'on saurait résoudre ? Pour cela on peut utiliser l'idée de l'algorithme magique en remarquant que la somme des longueurs est minimale en fin d'algorithme.

Activité 5 : un algorithme universel
Sur des exemples étudier le minimum obtenu. Est-ce un minimum global ? Et si on avait une méthode pour trouver le minimum global, qu'observerait-on ? Transformer le problème initial en un problème d'optimisation.

Indication 5
Se donner une représentation algébrique du problème. Quelles sont les variables à utiliser ? Quelles sont les contraintes ?

Éléments de solution 5
On note $D = ((d_{i,j}))$ la matrice $n \times n$ des distances entre chasseurs et fantômes, $d_{i,j}$ est la distance entre le chasseur i et le fantôme j. On cherche alors à minimiser la somme des distances entre les chasseurs et les fantômes. Ce problème est connu sous le nom de *problème d'affectation* ou problème de couplage parfait de poids minimal.
La méthode (algorithme) *hongroise* proposé par Kuhn (1955), permet de résoudre ce problème par transformations successives de la matrice D.
Pour généraliser encore le problème, on peut le modéliser par ce que l'on appelle un *programme linéaire en nombres entiers*.
On note $x_{i,j}$ les variables du problème, $x_{i,j} = 1$ si le chasseur i tire sur le fantôme j, $x_{i,j} = 0$ sinon. On note X la matrice des $x_{i,j}$.
Le problème est donc de minimiser

$$F(X) = \sum_{i,j} x_{i,j} d_{i,j}, \text{ somme des longueurs de segments ;}$$

sous l'ensemble des contraintes

$$\sum_{j} x_{i,j} = 1 \text{ pour tout chasseur } i \text{ (un chasseur ne tire que sur un seul fantôme);}$$

et

$$\sum_{i} x_{i,j} = 1 \text{ pour tout fantôme } j \text{ (un fantôme n'est la cible que d'un seul chasseur).}$$

Le problème a donc été réduit à une programme linéaire en nombres entiers.

Références

Cormen, T., Leiserson, C., Rivest, R. & Stein, C. (2010), *Algorithmique*, 3e edn, Dunod.

Kuhn, H. W. (1955), 'The hungarian method for the assignment problem', *Naval Research Logistic Quarterly* **2**, 83–97.