par Nicolas K. Blanchard

Même le hasard peut créer des certitudes

Si les algorithmes probabilistes sont à l'honneur pour leur efficacité, on peut leur reprocher cette intervention du hasard. Il existe cependant des cas où l'on peut passer par des méthodes probabilistes pour obtenir des solutions déterministes.

Un problème classique de combinatoire est celui des amis et des inconnus. Un groupe de personnes se réunit à une fête et on se demande si l'on peut trouver ou bien un trio de personnes qui se connaissent toutes mutuellement, ou bien un trio de convives qui ne se connaissent pas du tout (au sens que deux quelconques de ces convives ne se connaissent pas). Combien de personnes doivent être présentes pour garantir cette propriété ? Dans ce cadre simple, la réponse est immédiate, il suffit d'avoir six personnes (et six sont nécessaires). Le problème devient beaucoup plus difficile si on le généralise pour demander combien de personnes sont nécessaires pour qu'il y ait un groupe de n personnes qui se connaissent parfaitement ou pas du tout.

Les amis et les inconnus

Le problème peut être posé de manière plus formelle : soit $R(i, j)$ le nombre de personnes nécessaires pour garantir un groupe de i convives se connaissant ou bien un groupe de j personnes ne se connaissant pas. On a alors un majorant : $R(i, j) \leq R(i-1, j) + R(i, j-1)$. En effet, prenons un convive quelconque. On peut séparer les autres personnes en deux groupes : ceux qui le connaissent, et les autres. Si le premier groupe est plus grand que $R(i-1, j)$ ou le deuxième plus grand que $R(i, j-1)$, alors on peut continuer par récurrence,

et comme on a au moins $R(i-1, j) + R(i, j-1) - 1$ personnes, au moins un de ces deux cas arrive.

Calculer une borne inférieure pour $R(i, j)$ est plus difficile, mais peut se faire en utilisant un outil développé au départ par Paul Erdös : la méthode probabiliste. Pour cela, on va prendre un objet de grande taille (ici un graphe représentant les convives) et dont la structure est aléatoire (les liens représentant les connaissances). On borne alors la probabilité d'avoir une propriété – comme l'existence d'un groupe de i personnes se connaissant – en fonction de la taille du graphe et on montre que si cette probabilité n'est pas égale à 1, alors il doit nécessairement y avoir au moins un cas où la propriété n'est pas vérifiée.

Appliquons cette démarche sur notre exemple. Prenons donc un groupe de n convives, et disons que chaque couple possible (x, y) se connaît indépendamment avec probabilité 1 / 2. Tout d'abord, il existe $\binom{n}{i} = \frac{n!}{i!(n-i)!}$ manières de prendre i personnes dans un groupe de n. Comme il y a $i(i+1)/2$ liens de connaissances dans un groupe de i convives, la probabilité qu'ils se connaissent tous mutuellement est exactement $1 / 2^{i(i+1)/2}$. Or, la probabilité d'un ensemble d'évènements est inférieure à la somme des probabilités (peu importe qu'ils soient ou non indépendants), donc la probabilité d'avoir un groupe de i personnes se connaissant est au plus $2^{-i(i+1)/2}\binom{n}{i}$. Si $i = j$ (pour faire simple), la probabilité d'avoir un groupe de i convives se connaissant parfaitement ou ne se connaissant pas du tout est au plus égal à $2 \times 2^{-i(i+1)/2}\binom{n}{i}$. En majorant largement $\binom{n}{i}$ par $2^{i \ln n}$, on obtient que la probabilité d'avoir un tel groupe est inférieure à 1 lorsque $n < 2^{(i+1)/2}$. Si la probabilité d'avoir un tel groupe dans un graphe de taille n est strictement inférieure à 1, il doit exister au moins un graphe de cette taille ne possédant aucun groupe de i personnes se connaissant parfaitement ou ne se connaissant pas du tout. On a donc montré l'existence de contre-exemples, et donc une borne inférieure, de manière complètement non constructive, à coup de hasard.

C'est la force de la méthode probabiliste : prouver des bornes réelles et ne dépendant pas du hasard en passant par des outils venant des probabilités.

La dérandomisation

Si un algorithme probabiliste fonctionne bien, il peut être tentant de faire disparaître le côté aléatoire. En pratique, on fait tourner peu d'algorithmes aléatoires dans les ordinateurs, car créer des *bits* vraiment aléatoires est difficile (on peut utiliser des sondes thermiques ou des variations dans l'horloge de l'ordinateur, mais plus souvent on exploite des générateurs pseudo-aléatoires).

En pratique, des algorithmes déterministes créent des séquences de nombres « ressemblant fortement » à des séquences aléatoires et de nombreuses conjectures correspondent à l'existence de générateurs satisfaisant certaines propriétés. Ainsi, la classe BPP des problèmes que l'on peut résoudre en temps polynomial probabiliste peut se retrouver égale à la classe P si l'on trouve un jour un générateur pseudo-aléatoire d'« assez bonne qualité ». C'est tout l'enjeu de la *dérandomisation*.

MAX-CUT et la dérandomisation

Déjà, la taille de la coupe est au moins égale à $m / 2$, car il doit exister une solution de taille au moins égale à l'espérance de l'algorithme probabiliste. On va construire de manière gloutonne les ensembles A et B. Prenons un sommet u quelconque. Si parmi les sommets déjà classés il possède plus de voisins dans A, alors on met u dans B, et réciproquement. Pour montrer que cet algorithme produit une solution de taille au moins la moitié de l'optimale, considérons l'espérance de la taille de la solution qui serait produite si l'on procédait au hasard à partir d'un certain sommet u. Dans ce cas, on prend avec probabilité 1 / 2 chaque arête qui n'est pas déjà décidée (dont les deux sommets ne sont pas déjà dans A ou B). L'espérance de la taille de la coupe est donc $k + m' / 2$, où m' est le nombre d'arêtes restantes et k est le nombre d'arêtes déjà prises entre A et B. Cette espérance est aussi égale à la probabilité :

$$P(u \in B) \times (k + k_1 + m'' / 2) + P(u \in A) \times (k + k_2 + m'' / 2),$$

où k_1 est le nombre d'arêtes entre u et A, k_2 entre u et B, et $m'' / 2$ le nombre d'arêtes restantes une fois u placé dans A ou dans B. Comme $P(u \in B) = Pr(u \in A) = 1 / 2$ et que la moyenne de $k + k_1 + m'' / 2$ et de $k + k_2 + m'' / 2$ vaut $k + m' / 2$, le plus grand des deux vaut au moins $k + m' / 2$. En mettant u dans l'ensemble maximisant le nombre d'arêtes coupées jusqu'ici, on ne réduit pas l'espérance. Si l'on s'en tient à cette stratégie du début à la fin, on part d'une espérance égale à $m / 2$ et on ne la réduit jamais (par induction), donc on arrive avec une solution dont la taille vaut au moins $m / 2$, tout cela de manière déterministe.

Parfois, cependant, on peut se contenter de méthodes plus simples. Prenons le problème suivant (connu sous le nom MAX-CUT) : ayant un graphe G avec m arêtes, il faut trouver une partition des sommets en deux ensembles A et B maximisant le nombre d'arêtes entre A et B (ce nombre est la *taille de la coupe*). C'est un problème difficile (il est NP-complet), mais il existe un algorithme élémentaire donnant une (1 / 2)-approximation (ce qui signifie que la solution produite par l'algorithme possède au moins la moitié du nombre d'arêtes de la solution maximale). L'idée est intuitive : il suffit de prendre n'importe quelle partition aléatoire (ou chaque sommet est dans A avec probabilité 1 / 2). Chaque arête est prise dans la solution si les deux sommets correspondants sont dans des ensembles différents, ce qui arrive avec probabilité 1 / 2. Chaque arête de la solution maximale est dans la solution aléatoire avec probabilité 1 / 2, et l'espérance du nombre d'arêtes dans la solution est donc au moins égale à la moitié de la solution maximale.

On peut dérandomiser cet algorithme en utilisant la méthode des espérances conditionnelles (voir en encadré).

Si cette méthode est puissante, elle ne peut cependant être appliquée à tous les algorithmes probabilistes. Il y a toutefois une chance que tout algorithme puisse être dérandomisé d'une manière ou d'une autre, et cela est une conséquence de plusieurs conjectures actuellement étudiées. Pour le problème de Ramsey, cependant, citons Paul Erdös : *« Imaginez une puissance extraterrestre beaucoup plus puissante que nous qui atterrit et demande la valeur de R(5, 5), sans quoi ils détruiront notre planète. Dans ce cas, nous devrions regrouper l'ensemble des ordinateurs et tous nos mathématiciens pour trouver la valeur. Mais supposons, par contre, qu'ils demandent la valeur R(6, 6) ; nous devrions alors tenter de détruire les extraterrestres. »*

N.K.B.