2. Mathématiques

Thèse idiote
le hasard fabrique des certitudes

La sécurité d'un avion repose sur le hasard... L'accès aux données protégées utilise aussi l'aléatoire. La théorie du « zéro savoir » ouvre de nouvelles perspectives. En science, l'heure est aux ruptures, totales, iconoclastes.

Imaginez-vous aux commandes d'un avion. Soudain un signal retentit : un avion non identifié se trouve sur la même trajectoire que la vôtre... et vole en sens inverse, droit sur vous ! Heureusement, tous les appareils sont équipés d'un système anticollision et le drame sera évité. En tirant simultanément à pile ou face et en échangeant leurs résultats jusqu'à ce qu'ils diffèrent, les deux avions vont réussir à s'accorder pour savoir qui ira vers le haut (pile, par exemple) et qui ira vers le bas (face). La collision n'aura pas lieu.

Pour prendre cette décision, l'usage du hasard a été fondamental. Pourquoi ? Car un ordinateur ne fait que ce qu'on lui dit de faire, et dans les mêmes conditions, il réagit de la même façon. Ainsi, si les deux avions exécutaient le même programme déterministe, ils prendraient la même décision et la collision serait inévitable. L'appel à l'aléatoire donne au programme anti-collision ce trait de génie qui lui permet de rompre la symétrie d'exécution des programmes de pilotage et donc d'aboutir à une décision différente pour chacun des deux avions, garantissant ainsi l'évitement.

Ainsi, l'usage de décisions aléatoires en informatique, qui renforce, paradoxalement, la fiabilité des programmes et leur sécurité, devient de plus en plus courant. Il permet de résoudre très efficacement de nombreuses difficultés, parfois même d'accélérer considérablement le temps des calculs. Quelques exemples : augmenter la fiabilité de calculs numériques, réduire le temps de calculs, rompre la symétrie des calculs (comme dans le cas d'un système anticollision) et surtout restreindre l'accès à des données informatisées et garantir l'authentification d'une personne (via la vérification de la connaissance du code secret pour les cartes bleues, par exemple).

Le problème de la vérification de connaissances pour identifier une personne est un grand classique de l'enseignement : comment détermine-t-on qu'un élève a bien compris une leçon ? En lui posant une question à laquelle il devrait savoir répondre, par exemple calculer l'aire d'un carré. Pour être sûr que l'élève n'a pas appris la réponse par cœur, le professeur choisit les données numériques du problème, la longueur du côté, « au hasard » dans un grand ensemble de valeurs. Si l'élève sait résoudre le problème pour cette donnée numérique aléatoire, le professeur conclut que l'élève saurait le résoudre pour toutes les valeurs, et sait donc calculer l'aire d'un carré quelconque. Pour le suc-

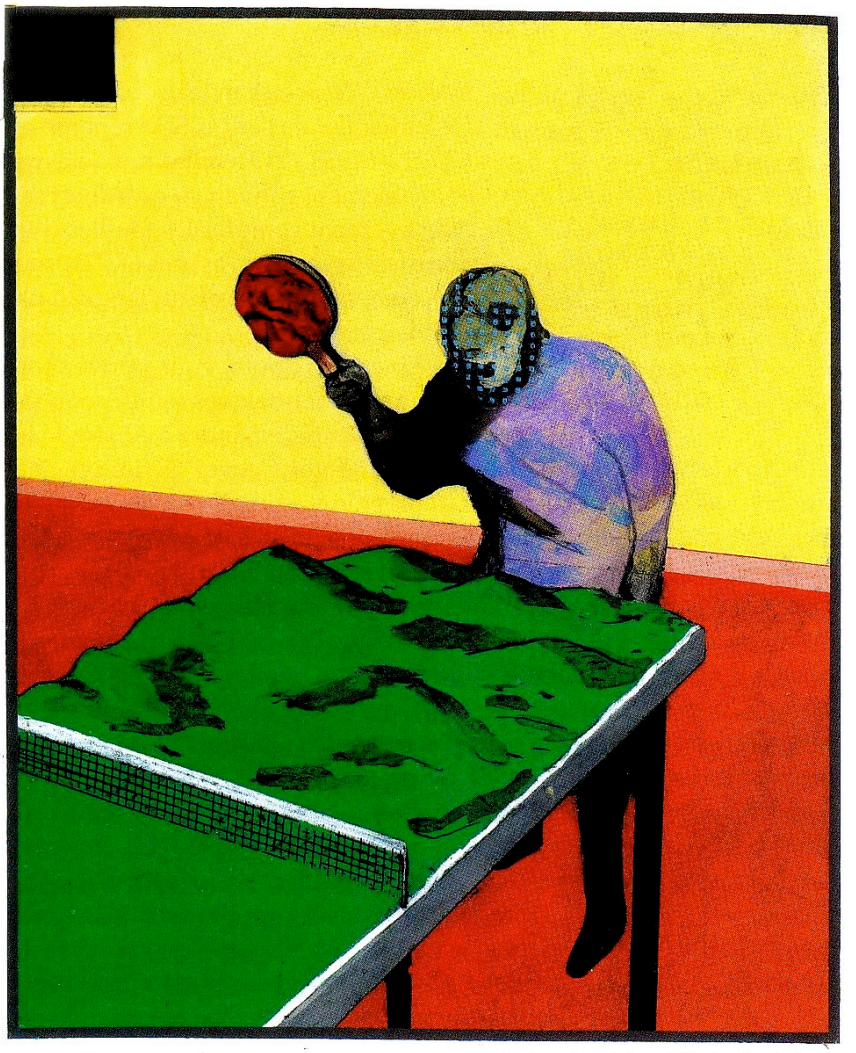


Illustration Michel Galvin

Si près de vous, un quidam se met à parler de métastase, avec votre science de la rupture et du hasard, vous saurez qu'il évoque le poète et dramaturge homonyme italien du XVIIe siècle. Et vous saurez que ce Métastase-là s'écrit avec un M majuscule.

cès de ce test, il est primordial que les données soient choisies de façon aléatoire : si le professeur pose chaque année les mêmes problèmes, les élèves le remarqueront très vite et se passeront les devoirs d'une année sur l'autre, le test perdra alors rapidement sa fiabilité. Ce type de procédé est à la base de tous les systèmes de sécurité informatique évolués actuels : vérifier l'identité d'une personne en lui posant une question *aléatoire* à laquelle elle seule saura répondre.

L'utilité de l'aléatoire est reconnu depuis longtemps. Les sondages en sont un bon exemple : il s'agit de sélectionner « le plus aléatoirement possible » un petit nombre de personnes pour en extrapoler l'avis général de la population nationale. Un autre exemple est celui du comptage : imaginons que je souhaite connaître approximativement le nombre de pièces défectueuses produites par une usine qui sort dix millions de pièces par jour. Si je sélectionne 10 000 = 10 000 000 / 1 000 pièces au hasard dans ma chaîne de production, et que je compte combien sont défectueuses dans ma sélection, par exemple 37, en multipliant par 1 000 le résultat, soit 37 000 ou encore 0,37%, j'aurai une excellente approximation du nombre exact de pièces défectueuses produites chaque jour par mon usine.

Lors de l'identification d'une personne ou d'un calcul, nous luttons contre un « adversaire » théorique : dans le premier cas, l'adversaire est simplement la personne à identifier qui cherche à nous tromper ; dans le second cas, l'adversaire cherche à nous donner les plus mauvaises données pour nous faire perdre le plus de temps ou de précision dans le calcul. Le recours à des décisions aléatoires permet, à l'instar du dribble ou de la feinte en football, de maintenir une incertitude qui interdit à l'adversaire de deviner nos intentions.

Reprenons l'exemple de l'identification d'une personne qui souhaite accéder à des données sensibles. Dans ce cas, notre but est de déjouer un adversaire qui cherche à passer le test d'identification par tous les moyens. Les systèmes de sécurité classiques identifient les personnes par leurs capacités à répondre correctement à une question. Par exemple, l'identification du propriétaire d'une carte bleue repose sur sa capacité à taper le bon code secret. Mais le système comporte un danger : pour répondre correctement à la question « Quel est votre code secret ? », vous devez donner votre code à la machine, qui peut alors usurper très facilement votre identité. C'est l'arnaque classique des fausses machines de retrait qui sévissait il y a quelques années : une fois votre carte insérée, la fausse machine vous demandait de taper votre code, pendant que s'effectuait une copie de votre carte ; celle-ci était restituée tandis qu'on vous faisait croire à un problème réseau. Le malfaiteur disposait alors de toutes les informations nécessaires pour débiter votre compte en banque grâce à cette copie. Cette arnaque serait impossible si vous ne donniez pas au système d'identification la réponse à sa question, mais si vous réussissiez à le *convaincre* que vous la connaissez, sans jamais la lui donner.

C'est toute la magie des protocoles dits «zero-knowledge» («zéro savoir»): convaincre que l'on sait, sans rien dévoiler de ce que l'on sait.

Voici comment nous pourrions procéder. Imaginons que nous cherchions à identifier une personne — que nous appellerons Albert —, qui sache faire quelque chose d'extraordinaire : compter les feuilles d'un arbre. Voici une méthode qui permettra de déterminer si Albert sait exécuter cette opération tout en préservant son secret, c'est-à-dire sans nous apprendre pour autant comment il s'y prend, ni même combien l'arbre compte de feuilles. Vous, le système d'identification, montrez un arbre choisi au hasard à Albert et lui demandez combien il compte de feuilles. Il vous répond, par exemple « 67 938 ». Étant donné que vous ne savez pas compter les feuilles d'un arbre sans vous tromper (c'est le secret qu'Albert ne souhaite pas dévoiler), vous êtes incapables de vérifier si cette réponse est exacte. Pour vous convaincre qu'Albert, lui, sait, procédez comme suit : vous demandez à Albert de se bander les yeux et vous arrachez de l'arbre un nombre aléatoire de feuilles (par exemple 217) ; puis vous demandez à Albert d'ôter son bandeau et de recompter les feuilles de l'arbre. Si sa réponse est 67 721 = 67938 – 217 feuilles, vous êtes raisonnablement convaincu qu'Albert sait effectivement compter les feuilles d'un arbre (si sa réponse n'est pas 67 721, vous êtes en revanche certain qu'il ne sait pas les compter).

En cas de doute, vous pouvez toujours arracher quelques feuilles supplémentaires et redemander à Albert de les compter jusqu'à ce que vous soyez définitivement convaincu. Vous aurez alors acquis la conviction qu'Albert sait bien compter les feuilles d'un arbre et pourtant vous ne saurez toujours pas comment il s'y est pris. Vous ne saurez d'ailleurs peut-être même pas combien il y avait de feuilles sur l'arbre ! En effet, Albert, de son côté aura pris soin de choisir un nombre aléatoire au début (par exemple 131), qu'il aura ajouté à chacun de ses comptages : la première fois, l'arbre comptait en fait 67 807 feuilles, mais il a répondu 67 938 = 67 807 + 131 ; et la seconde fois, l'arbre comptait 67 590 = 67807-217, mais il vous a répondu 67 721 = 67 590 + 131. Vous êtes convaincu qu'Albert sait bien compter les feuilles d'un arbre, mais vous n'avez appris rien d'autre… pas même le nombre de feuilles de votre arbre. Le secret d'Albert est donc bien gardé ! C'est toute la magie des protocoles dits *« zero-knowledge »* (*« zéro savoir »*) : convaincre que l'on sait, sans rien dévoiler de ce que l'on sait.

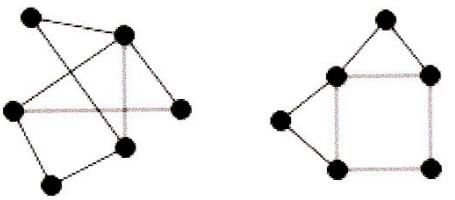
« Coquetteries de l'âme et artifices de l'esprit que tout cela », poétisait, à propos des ruses d'Ulysse, le cher Nikos Kazantzaki.

Remarquons que l'usage de l'aléatoire a été fondamental pour le bon déroulement de l'identification et la préservation du secret. Si vous aviez arraché un nombre déterminé de feuilles (mettons 13) au lieu d'un nombre aléatoire et qu'Albert le connaissait, alors il lui aurait suffi de répondre n'importe quel nombre, par exemple 66 932, la première fois, puis ce nombre moins 13 la seconde fois, 66 919. De même, si Albert n'avait pas ajouté un nombre aléatoire au nombre de feuilles de l'arbre, vous connaîtriez le nombre de feuilles de l'arbre.

Bien entendu pour réaliser de tels protocoles d'identification dans des systèmes de sécurité, il ne s'agit pas de demander à quelqu'un de compter les feuilles d'un arbre. Pour utiliser le principe *zero-knowledge* d'identification, on s'appuie sur un problème très difficile à résoudre (compter les feuilles d'un arbre !) et l'objet du

protocole est de convaincre le système d'identification que l'on sait le résoudre sur des cas précis, sans dire comment on a procédé.

Déterminer si deux graphes sont identiques est un autre exemple (voir graphique ci-contre). Un graphe, en informatique, est un ensemble de points (appelés sommets) reliés entre eux par des traits (appelés arêtes). On dit que deux graphes sont identiques si l'on peut passer de l'un à l'autre en changeant uniquement la position des sommets. On ne connaît pas actuellement de solution efficace pour déterminer si deux graphes sont identiques, mais on peut résoudre ce problème très rapidement si l'on a la « clé », les mouvements des sommets qui permettent de passer d'un graphe à l'autre. Aussi, si Albert possède cette clé, il pourra convaincre avec un petit nombre de questions le système d'identification qu'il possède bien cette clé, sans la lui révéler. Sans cette clé, Albert sera déjoué très rapidement par le système d'identification. C'est là tout l'intérêt des protocoles *zero knowledge*. Ils offrent une garantie de sécurité maximale.



Deux graphes identiques

L'usage de l'aléatoire en informatique a d'autres applications surprenantes. Dans les années 1990, Blum, Luby et Rubinfeld, chercheurs en informatique à Berkeley, eurent l'intuition géniale que l'aléatoire peut permettre de corriger les bugs (inconnus) d'un programme. Les cas où un programme produit une solution erronée ou très imprécise (bugs) sont rares mais ils existent et sont imprévisibles. Le problème est que souvent, du fait de sa complexité, on ne sait pas pourquoi le programme « plante ». L'adversaire est ici les « bugs » inconnus qui produisent des réponses erronées pour certaines valeurs inconnues des entrées du programme. L'aléatoire peut permettre dans certaines conditions de contourner ces bugs : l'idée est de substituer les valeurs en entrées par des valeurs aléatoires qui doivent donner le même résultat en sortie. Cette idée a valu à ces chercheurs d'importants contrats avec l'armée américaine afin de renforcer la fiabilité de ses unités de calcul en particulier.

Obtenir une sécurité absolue grâce à l'aléatoire, par définition imprévisible, quel paradoxe ! Si les valeurs de notre source aléatoire étaient prévisibles, alors tous les protocoles de sécurité, d'évitement, ou de fiabilisation présentés ci-dessus seraient déjoués très simplement. Or, tout ce que nous savons calculer est prévisible puisque calculé. Actuellement, les valeurs aléatoires sont le plus souvent générées par des suites mathématiques aux comportements « suffisamment compliqués » pour paraître aléatoires. Mais si la valeur initiale (la « graine ») de cette suite pseudo-aléatoire est connue de notre « adversaire », alors il peut calculer toutes les valeurs de la suite et prévoir toutes nos décisions, ce qui lui permet de contourner toutes nos stratégies basées sur du « vrai » aléatoire. Il existe cependant d'autres méthodes pour produire des suites aléatoires : enregistrer les dates d'émission de particules radioactives, par exemple. Ces suites semblent satisfaire assez de critères pour être qualifiées d'« aléatoires », mais le sont-elles suffisamment ? Les questions qui déterminent ce qu'est l'aléatoire, ou un comportement aléatoire, sont encore ouvertes. Il n'est même pas absolument clair pour le physicien que l'aléatoire existe dans le monde où nous vivons : tout est peut-être déterminé par des lois physiques précises, même si la mécanique quantique et ses postulats d'incertitude semblent indiquer le contraire.

Mais, plus prosaïquement, est-ce bien raisonnable que nos sécurités bancaire, physique, voire nucléaire (accès aux centrales) dépendent d'un aléatoire qu'on ne sait pas vraiment définir ? D'autant que l'aléatoire utilisé dans ces programmes n'est que du « pseudo-aléatoire » provenant de suites mathématiques bien déterminées. Peut-on essayer de se dispenser autant que possible d'accès à l'aléatoire ? Ces questions animent fortement la communauté informatique, et d'importants travaux sont menés sur ces points actuellement. Les réponses, quelles qu'elles soient, auront un très grand impact sur notre compréhension de la notion de calcul avec un ordinateur, et surtout sur la détermination de la nature même du « hasard ».

Nicolas Schabanel

MINOTAURE

La guerre est finie

L'avenir appartient aux idiots

L'espoir est dans mon corps

Un parlement des non-humains

Miles Davis contre Bourdieu

stratégi
enquêt
histoi
littératu
mathématiqu
philosoph
médeci
mo
architectu
dro
sociolog
investigati
busines
écologi
religi
gastronom
a
spectac
musiq
psychologi
linguistiq

M 07807 - 1 H - F: 9,50 € - RD