

L'oracle qui calcule l'incalculable

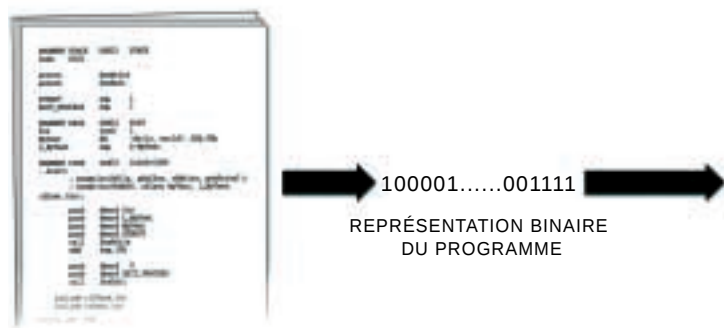
Alan Turing a démontré que sa machine universelle, et par conséquent, les ordinateurs les plus puissants, ne pourraient jamais résoudre certains problèmes. Par exemple, une machine de Turing ne peut pas toujours déterminer si un programme aboutira ou continuera à tourner indéfiniment. Dans certains cas, le mieux qu'une machine de Turing puisse faire est d'exécuter le programme et d'attendre, peut-être éternellement, qu'il se termine. Dans sa thèse de doctorat, Turing imaginait qu'une machine équipée d'un «oracle» pourrait remplir cette fonction et d'autres «calculs incalculables». L'exemple suivant illustre comment, en principe, pourrait fonctionner un oracle.

Considérons une machine hypothétique destinée à résoudre l'épineux problème de l'«arrêt des programmes». Un programme informatique peut être représenté comme une suite finie de 1 et de 0, suit qui peut s'interpréter comme la représentation binaire d'un nombre entier (par exemple 1011011 équivaut à 91). Le rôle de l'oracle serait défini ainsi : étant donné un entier qui représente un programme (pour tout ordinateur pouvant être simulé par une machine de Turing), le résultat sera 1 si le programme a une fin, sinon il sera 0.

L'oracle serait un dispositif parfait de mesure et de stockage (une mémoire) contenant une valeur précise (appelons-la τ , comme Turing) d'une grandeur physique (la mémoire pourrait ressembler à un condensateur qui emmagasine

une quantité donnée d'électricité). En fait, τ serait une grandeur représentée par une suite infinie de chiffres binaires : 0,00000001101... Le nombre τ aurait une propriété fondamentale : les chiffres qui le composeraient représenteraient précisément les programmes qui ont une fin et ceux qui n'en ont pas. Par exemple, si l'entier représentant un programme était 87 352, alors l'oracle pourrait calculer le 87 352^e chiffre de la suite binaire de τ (en comptant les chiffres décimaux de la gauche vers la droite après la

PROGRAMME INFORMATIQUE



de représentations symboliques, comme les formules dans les calculs mathématiques. Pourtant, les mécanismes cognitifs ne sont rien d'autres que des décharges de neurones. Les cogniciens sont confrontés à une question lancinante : comment faire converger ces deux approches ?

Turing avança une solution : le cortex est un réseau de neurones qui agit comme un ordinateur à usages multiples, capable d'accomplir les processus symboliques les plus abstraits. Cette affirmation, très en avance sur son temps en 1948, reste aujourd'hui encore parmi les meilleures réponses possibles à l'une des questions les plus ardues des sciences cognitives.

CALCULER L'INCALCULABLE...

En 1935, Turing invente le dispositif connu aujourd'hui sous le nom de «machine universelle de Turing». Elle est constituée d'une mémoire illimitée qui stocke à la fois le programme, les données et d'une tête de lecture-écriture qui parcourt la mémoire, symbole par symbole, lisant l'information et ajoutant de nouvelles données. Chaque instruction de base de la machine est très simple : par exemple «identifier le symbole sur lequel la tête de lecture est positionnée», «écrire le chiffre 1», «se décaler d'un cran vers la gauche». La complexité

naît de l'association d'un grand nombre de ces opérations de base. En dépit de sa simplicité, une machine de Turing peut exécuter les mêmes tâches que les plus puissants ordinateurs d'aujourd'hui. En fait, tous les ordinateurs modernes sont, par essence, des machines universelles de Turing.

En 1935, Turing voulait créer une machine, aussi simple que possible, et capable de faire les mêmes calculs qu'un mathématicien pourrait exécuter en utilisant une méthode algorithmique, sans limitation de temps, d'énergie, de papier ni de crayons et en étant parfaitement concentré. Cette capacité lui valu le qualificatif d'«universelle». Comme Turing l'écrivait lui-même, «Les ordinateurs sont conçus pour remplir n'importe quelle tâche qu'un opérateur humain discipliné, mais dépourvu d'intelligence, pourrait accomplir.»

Si puissants que soient ces ordinateurs, peut-on en concevoir de plus performants ? On peut effectivement déclore de tels «hyperordinateurs» sur le papier, mais personne ne sait si l'on pourra en construire un jour. Un nombre croissant d'informaticiens travaillent sur ces hyperordinateurs. Selon certains, le cerveau humain – le processeur le plus complexe que l'on connaisse – serait en fait un hyperordinateur naturel.

Avant que l'on s'intéresse aux hyperordinateurs, on pensait que tout problème de traitement d'informations

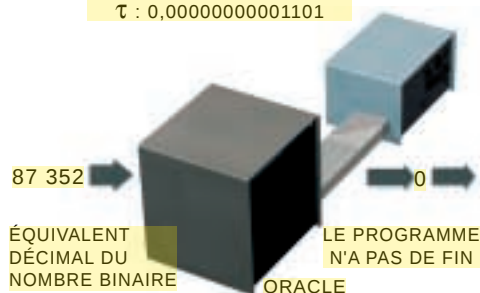
trop complexe pour une machine universelle de Turing était «incalculable». En ce sens, un hyperordinateur calcule l'incalculable.

Certaines opérations paraissent défier les ordinateurs classiques même dans les domaines les plus élémentaires des mathématiques. Par exemple, quand on donne à une machine de Turing des propositions arithmétiques choisies au hasard, elle ne sait pas toujours distinguer ce qui est un théorème ($7+5=12$) de ce qui n'en est pas un (tout nombre est la somme de deux nombres pairs). Un autre problème incalculable est de nature géométrique. Avec un jeu de carrés de tailles différentes et dont des arêtes ont des couleurs différentes, on doit paver le plan ; les carrés doivent se toucher (sans se superposer) et les arêtes adjacentes doivent être de même couleur. Les logiciens William Hanf et Dale Myers de l'Université de Hawaï ont découvert un jeu de carrés qui pave le plan, mais les motifs sont trop compliqués pour être calculés par une machine de Turing. Enfin, en informatique, une machine de Turing ne peut pas toujours prévoir si un programme aura une fin ou s'il continuera à tourner sans fin. Aucun langage de programmation standard (Pascal, BASIC, Prolog, C,...) n'a d'outil qui détecterait tous les bogues risquant de faire échouer un programme, notamment les erreurs qui aboutissent à des boucles infinies.

virgule). Si c'était un 0, l'oracle conclurait que le programme tournera sans fin.

Évidemment, sans τ , l'oracle ne servirait à rien, et l'on ne parviendrait peut-être jamais à trouver un tel modèle mathématique. Pourtant, la quête de ce concept mathématique qui éclairerait la théorie de l'incalculabilité se poursuit.

MÉMOIRE D'ORACLE AVEC
 $\tau : 0,00000000001101$



Turing a été le premier à rechercher des machines qui mèneraient à bien des tâches mathématiques trop compliquées pour la machine de Turing. En 1938, dans sa thèse de doctorat à l'Université de Princeton, il décrivait «un nouveau type de machine», la «machine O».

C'est une machine de Turing pourvue d'une boîte noire, ou «oracle», un dispositif qui exécuterait les tâches incalculables. Sinon, les machines O seraient identiques aux ordinateurs classiques. Un programme alimenterait la machine qui produirait une sortie numérique à partir de l'entrée, en utilisant un procédé itérant des opérations de base de la machine ; l'une d'entre elles consisterait à transmettre les données à l'oracle et à enregistrer sa réponse.

À LA RECHERCHE DE L'ORACLE

Turing ne donnait aucune indication sur le fonctionnement d'un oracle, pas plus qu'il n'avait expliqué, dans ses travaux antérieurs, comment une machine de Turing accomplissait ses fonctions de base, par exemple, «identifier le symbole dans la tête de lecture». Toutefois, on peut en imaginer les principes fondateurs (voir l'encadré). En principe, même un réseau de type B adéquat pourrait calculer l'incalculable, à condition que l'activité des neurones soit désynchronisée : quand une horloge centrale

impose la même cadence à tous les neurones, le fonctionnement du réseau est parfaitement simulé par une machine universelle de Turing.

En théorie, les hyperordinateurs distingueraient une proposition mathématique qui est un théorème d'une proposition qui n'en est pas un ; un correcteur de bogues pourrait même prévoir si un programme, quel que soit le langage dans lequel il est écrit, entrerait ou non dans une boucle infinie.

Si l'on parvenait un jour à construire des hyperordinateurs, on pourra résoudre d'innombrables problèmes de logique et de mathématique encore jugés insolubles. Le succès de l'entreprise tient essentiellement à une interrogation : réussira-t-on à concevoir un oracle ?

Des recherches sont menées tant en physique, qu'en chimie ou en biologie. Peut-être la solution viendra-t-elle de molécules complexes ou de structures qui s'arrangent selon des motifs compliqués. Ou bien, comme le pense Jon Doyle de l'Institut de technologie du Massachusetts, peut-être existe-t-il, dans la nature, des systèmes qui auraient un spectre discret, accompliraient des tâches incalculables, et produiraient des sorties appropriées (0 ou 1) après avoir été bombardées par le signal d'entrée.

Hormis les logiciens, nul ne s'est intéressé aux machines O de Turing qui ont sombré dans l'oubli, et un mythe est apparu. D'après un récit apocryphe, Turing aurait démontré au milieu des années 1930 que les hyperordinateurs sont irréalisables. Avec le logicien Alonzo Church, son directeur de thèse à Princeton, il aurait énoncé un principe selon lequel une machine de Turing peut simuler le comportement de toute autre machine de traitement de l'information. Selon cette proposition, nommée à tort la «thèse de Church-Turing» (bien qu'ils n'en soient pas les auteurs), aucune machine ne peut traiter des informations que ne pourrait traiter une machine de Turing. En fait, Church et Turing affirmaient seulement qu'une machine de Turing pouvait remplacer n'importe quel mathématicien humain travaillant avec du papier et un crayon et suivant une méthode algorithmique,

affirmation bien plus modérée et ne minimisant pas du tout les potentialités des hyperordinateurs.

Même ceux qui travaillent sur les hyperordinateurs n'accordent pas aux contributions théoriques innovantes de Turing la place qu'elles méritent. Les spécialistes parlent couramment de «dépasser la limite de Turing» pour le traitement des informations. On lit quelquefois que les nouvelles machines «vont bien au-delà des conceptions de Turing» et sont «des ordinateurs que jamais Turing n'avait imaginés», oubliant que le génie britannique avait élaboré le concept il y a plus de 50 ans. Il est dommage que les hyperordinateurs de Turing semblent devoir connaître le même sort que celui qu'ont connu ses idées sur le connexionisme : l'oubli.

LES DERNIÈRES ANNÉES

À la fin de sa vie, au début des années 1950, Turing ouvrit un nouveau champ de recherches, celui de la vie artificielle. Il cherchait à simuler un mécanisme chimique où les gènes d'un œuf fécondé détermineraient la structure anatomique de l'animal ou de la plante à venir. Selon lui, ces recherches n'étaient pas sans rapport avec celles qu'il menait sur les réseaux de neurones, parce que «la structure du cerveau résulte nécessairement d'un mécanisme génétique d'embryogenèse et cette théorie, à laquelle je travaille actuellement, pourrait illustrer les contraintes que le mécanisme impose.» À cette époque, Turing fut le premier à explorer sur ordinateur les systèmes dynamiques non linéaires. Pour modéliser la chimie du développement, il utilisait des équations différentielles non linéaires.

Alors qu'il s'engageait dans tous ces domaines encore inexplorés, Turing mourut d'un empoisonnement au cyanure ; il s'agissait probablement d'un suicide. Le 8 juin 1954, alors qu'il allait avoir 42 ans, on le retrouva mort dans sa chambre. Il laissait d'innombrables notes manuscrites et des programmes informatiques. Plusieurs dizaines d'années après, on ne les comprend pas encore tous.

Jack COPELAND est professeur d'informatique à l'Université de Portsmouth. Diane PROUDFOOT est professeur de philosophie à l'Université de Canterbury, en Nouvelle-Zélande, où ils dirigent tous deux un Projet Turing qui développe les idées de Turing.

Hava SIEGELMANN, *Computation beyond the Turing Limit*, in *Science*, vol. 268, pp. 545-548, 28 avril 1995.

Alan TURING, Jean-Yves GIRARD, *La machine de Turing*, collection Points Sciences, Éditions du Seuil, 1995.

J. COPELAND, Diane PROUDFOOT, *On Alan Turing's Anticipation of Connectionism*, in *Synthese*, vol. 108, n° 3, pp. 361-377, mars 1996.

Jean LASSÈGUE, *Turing*, Collection Figures du savoir, Éditions Les Belles lettres, 1998.



Arômes ou réactions ?

HERVÉ THIS

Les chimistes identifient deux façons de donner du goût aux aliments.

Les choses doivent avoir le goût de ce qu'elles sont, disait le gastronome Curnonsky dans la première moitié du siècle. L'aphorisme a été adopté par de nombreux cuisiniers qui parlent aujourd'hui de «vérité» des goûts, mais est-il pertinent ? Le rôle du cuisinier n'est-il pas la transformation des produits alimentaires en vue de la confection des mets ?

Si l'objectif est l'obtention de goûts spécifiques, comment communiquer ceux-ci aux plats ? Soit en ajoutant des arômes, soit en organisant des réactions chimiques, afin que les arômes soient formés *in situ*.

La technique la plus simple, mise en œuvre par l'industrie agro-alimentaire, est l'utilisation d'arômes, extraits naturels ou solutions de molécules de synthèse. Leur emploi en cuisine est simple (on verse quelques gouttes dans un plat), mais leur composition emploie des «nez», c'est-à-dire des artistes qui assemblent des solutions de diverses molécules aromatiques afin de reconstituer leur version de goûts connus : fraise, gingembre, romarin... Les cuisiniers hésitent à se laisser supplanter par ces «nez», d'autant que l'utilisation des produits naturels (du vrai thym et du vrai romarin dans une ratatouille, par exemple) donne souvent un résultat aromatiquement plus riche, et certainement plus varié, que l'utilisation d'un arôme industriel de thym ou de romarin (les «nez» n'assemblent généralement pas autant de molécules aromatiques qu'il en existe dans les produits naturels).

Doit-on pour autant écarter les arômes ? Ce serait perdre une occasion d'élargir la palette des goûts. Pourquoi ne renforcerait-on pas la note «verte» d'une huile d'olive avec de l'hexanal ? Pourquoi n'ajouterait-on pas du 1-octène-3-ol à un plat de viande afin de lui donner un arôme de champignons ou de sous-bois (attention au dosage : en concentration excessive, la même molécule sent un peu le mois) ? Pourquoi n'emploierait-on pas de la bêta ionone afin de donner aux desserts cet étonnant

arôme de violette que les fleurs ont bien du mal à libérer ?

D'autre part, pour donner de la saveur, et non de l'arôme, le cuisinier pourra employer le monoglutamate de sodium ou des molécules analogues, qui donnent la saveur nommée «umami», également apportée par les oignons ou par les tomates. Il pourra utiliser la réglisse ou l'acide glycyrrizique, qui communiquent une saveur spécifique, qui n'est ni salée, ni sucrée, ni acide, ni amère.

Les progrès culinaires seraient-ils limités à cet avenir de l'aromatisation, qui s'apparente à l'ajout de fines herbes ou d'épices ? Certainement pas : le cuisinier sait bien que la cuisson transforme le goût de ses produits. Le feu est son allié inséparable, et la chimie peut l'aider à l'utiliser.

Par exemple, nous avons proposé dans cette rubrique, il y a quelques mois, la confection de caramels de glucose, de fructose ou, plus généralement, d'autres sucres que le saccharose (le sucre de table). Une expérience que chacun peut réaliser montre que ces caramels sont peut-être déjà présents en cuisine.

L'expérience se fonde sur une remarque apparemment paradoxale de cuisiniers qui, pour préparer des sauces, mettent dans une casserole des échalotes hachées et du vin blanc, qu'ils font réduire «à sec». Or certains vins blancs ne laissent aucun résidu dans les casseroles. Pourquoi ? Parce qu'il leur manque du glucose, du glycérol et bien d'autres constituants. Conclusion pratique : si le vin blanc risque de s'évaporer excessivement, ajoutons-lui du glucose avant de réduire... et nous obtiendrons ce caramel de glucose qui contribue au goût des vins réduits à sec.

Cette réduction, même ainsi épaulée, reste étonnante : pourquoi évaporer l'essentiel des molécules aromatiques qui sont présentes dans les vins ? La question se pose également dans le cas des fonds, que les cuisiniers obtiennent en

concentrant des bouillons de viande par chauffage. Si la concentration élimine les molécules aromatiques, pourquoi les fonds sont-ils néanmoins savoureux et parfumés ?

Anthony Blake et François Benzi, de la Société *Firmenich*, ont comparé par chromatographie un bouillon réduit des trois quarts, puis ramené à son volume initial par ajout d'eau, au même bouillon non modifié : la concentration en certains composés aromatiques était effectivement réduite, mais d'autres composés étaient nés des réactions, favorisées par l'échauffement, entre les divers constituants du bouillon. Reste à identifier ces réactions afin de perfectionner – qui sait ? – la confection des fonds.

Les possibilités de la chimie sont innombrables : nous avons sur nos étagères de cuisine une foule de produits quasi purs : chlorure de sodium, saccharose, triglycérides (dans l'huile), éthanol, acide acétique, etc. ; et, dans nos bibliothèques, des traités de chimie décrivent parfaitement les réactions de ces molécules. Ne pourrions-nous faire aujourd'hui la synthèse ?

Ainsi les réactions dites de Maillard, entre les acides aminés et les sucres, engendrent les composés goûteux et bruns de la croûte du rôti, du pain, les arômes du café ou du chocolat... Or les chimistes savent que ces réactions diffèrent selon l'acidité du milieu réactionnel. Trempérons-nous un jour les suprêmes de volaille dans du vinaigre ou dans du bicarbonate de soude avant de les passer sous le grill, créant ainsi des goûts nouveaux ? Pour retrouver une acidité de dégustation appropriée, une fois la réaction effectuée, on neutraliserait le vinaigre au bicarbonate, ou inversement...

Prochain rendez-vous France Info et Pour la Science le 30 juin 1999, avec la chronique Info Sciences de Marie-Odile Monchicourt.