pourlascience.com

Les génies de la science

N° 29 – novembre 2006-janvier 2007

POUR LA SCIENCE

L'actualité de l'histoire des sciences

TURING

Et l'informatique fut

ET AUSSI

L'expédition de Livingstone

Les concours

Les mathématiques en sanskrit

M 05317 - 29 - F: 6,90 € - RD

FRANCE METRO. 6,90 €, DOM Surface 7,50 €, BEL 7,50 €, CAN 9,50 $, USA 9,50$, CH 12,00 FS, LUX 7,50 €, PORT. CONT. 7,50 €, MAR 75 MAD.

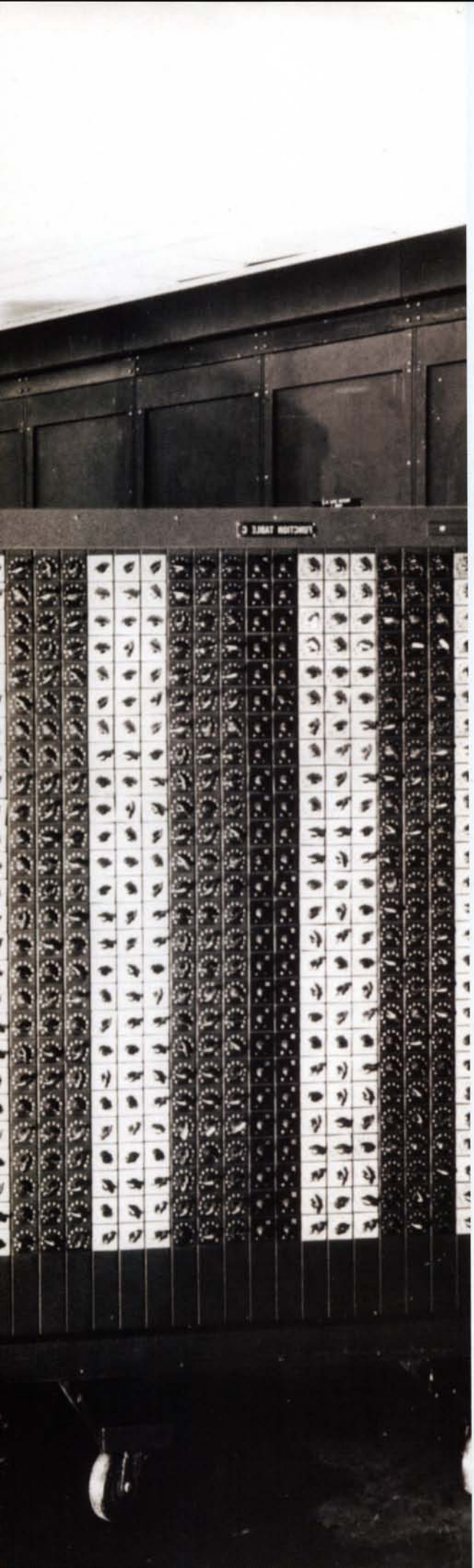
Les génies de la science

dossier

TURING

Et l'informatique fut

Jean Lassègue, chargé de recherche au CNRS, est attaché au Centre de recherche en épistémologie appliquée (CREA) de l'École polytechnique. Auteur d'un ouvrage sur Turing (*Turing*, Les Belles Lettres, Paris, 2003), il tente de saisir la portée de l'informatique au sein de l'environnement culturel qu'elle contribue à créer.



L'informatique a envahi *notre vocabulaire et nos gestes les plus quotidiens avec une force peu commune : toute machine aujourd'hui est équipée d'une « puce », d'un « capteur » ou d'un « programme ». L'informatique intervient dans les moyens de communication et de diffusion – courrier électronique, internet, téléphone portable –, de paiement et de réservation – carte de crédit, distributeurs, achat en ligne –, ou de classement et de prévision – bases de données, modélisations. C'est un instrument de pouvoir et de connaissance, aux enjeux colossaux, tant économiques que scientifiques et géo-stratégiques.*

Au cœur de cette révolution planétaire, personne n'hésiterait à nommer la cause : l'ordinateur. *Pourtant, une fois ce constat établi, nous sommes souvent bien en peine d'expliquer ce sentiment d'accélération incontrôlée des pratiques numérisées. Pour démythifier l'aura quasi magique qui entoure l'usage de l'ordinateur, plaçons-nous sur une échelle de temps plus large. À long terme, l'informatique* s'inscrit dans une longue histoire, *qui est celle de l'écriture des nombres et des langues. À plus court terme,* un homme, *plus que tout autre, a contribué à sa naissance : Alan Turing. Cet homme sut non seulement lancer les bases d'un nouveau mode de connaissance, mais aussi lui assigner des limites. Renouer les fils de l'histoire, comprendre le projet scientifique de Turing, contribueront, nous l'espérons, à lutter contre l'opacité de pratiques devenues mondiales et à apprécier pleinement ce que l'on doit à l'informatique.*

Jean Lassègue

US National Library of Medicine

La tragédie Turing

Logicien renommé, inventeur de l'ordinateur, déchiffreur du code de la marine allemande pendant la Seconde Guerre mondiale, Turing a tout pour lui… sauf d'être homosexuel dans une Angleterre conservatrice.

La vie de Turing (1912-1954) est celle d'un destin scientifique exceptionnel, dans ce qu'il peut avoir de *prométhéen* et de *tragique*. *Prométhéen* parce que Turing a fondé une nouvelle façon de voir le monde à partir d'un modèle calculatoire et en a exploré les limites. Par exemple, la notion de calcul, employée pourtant depuis l'aube des temps, n'avait jamais fait, avant lui, l'objet d'une définition rigoureuse rendant possible la distinction entre ce qui relève du calculable et ce qui n'en relève pas. *Tragique* parce que cette quête l'a anéanti. Son suicide a mis un terme à une carrière scientifique fulgurante qui dura moins de 20 ans. Ces 20 ans seront pourtant décisifs pour la techno-science qui marque si profondément notre culture contemporaine : nous en sommes les héritiers directs, nous qui vivons dans une société du « tout-numérique » où les ordinateurs ont acquis une place centrale. Turing, plus que tout autre, est le père fondateur de cette révolution.

Nous sommes en 1912, à l'apogée de l'Empire britannique. Comme Rudyard Kipling 40 ans plus tôt, Turing est, à la naissance, séparé de ses parents qui vivent à Madras, en Inde, où son père est administrateur colonial. Placé en nourrice en Angleterre, élevé à la dure dans une *grammar school*, il se découvre dès l'adolescence homosexuel et scientifique, deux traits qui en font un marginal dans un milieu imbu de culture classique et de morale victorienne. Ce n'est qu'à Cambridge, épicentre de la vie scientifique britannique, où il sera admis comme étudiant, puis deviendra enseignant et chercheur, que cette double marginalité ne posera pas de problème ; ailleurs, et tout au long de sa vie, la société britannique tâchera d'y mettre bon ordre, en s'efforçant de tenir le scientifique, utile à la nation, éloigné de l'homosexuel, indigne des bonnes mœurs. Mais Turing ne l'entend pas ainsi : c'est dans le domaine de la science qu'il s'exprime de tout son être. La capacité de détermination propre à la science doit permettre d'éclairer *qui* il est, dans sa pensée, dans son corps, dans ses rapports à autrui. Et pour lui qui pose les problèmes en termes scientifiques, si sa propre origine ressemble, dans son obscurité, à un calcul crypté, seule la science est à même d'en déchiffrer le code et d'éclairer son destin d'homme. Voici, à travers 21 tableaux inspirés de sa vie, quelques clés pour percer les mystères de ce personnage torturé.

Comment poussent les plantes ?

Noël 1922, Londres. Alan Turing, dix ans, reçoit en cadeau un livre de Edwin Tenney Brewster intitulé *Natural Wonders every child should know [Merveilles de la nature que tout enfant devrait connaître]*, et le dévore. L'ouvrage raconte comment les enfants se développent à partir d'une cellule fécondée, suivant les lois de la physique et de la chimie. Stupéfait, le jeune Alan découvre un corps présenté comme une gigantesque machine. Par conséquent, il doit être possible de déterminer, à l'aide de la physique, de la chimie et des mathématiques, les lois qui régissent sa construction.

1926, Sherborne Grammar School. Le directeur de l'établissement fait un procès à l'un de ses élèves, déjà dispensé de grec et de français pour incompétence et paresse notoires. Son nom : Alan Turing.

Library and Archive Center, King's College Cambridge, AMT/K/7/2

Jean Lassègue

Alors âgé de 14 ans, Turing est un garçon peu sociable, catalogué bizarre. Accoutré comme l'as de pique, il arrive parfois à l'école affublé d'un masque à gaz, pour éviter le rhume des foins. Le directeur lui reproche d'être incapable de rédiger correctement et proprement ses dissertations, et demande son redoublement. L'adolescent a cependant la tête ailleurs : il vient de découvrir une série infinie $x - x^3/3 + x^5/5 - x^7/7 + \ldots$ qui lui permet non seulement de calculer le nombre π, mais, de façon générale, toute valeur de la fonction tangente inverse *(arctan)*. Il a en effet démontré que, pour toute valeur de x, cette série infinie est égale à $arctan(x)$. Notamment, pour $x = 1$, $arctan(1) = \pi/4$.

Son professeur de mathématiques est étonné de cette performance inattendue : la procédure de calcul découverte par Turing a été développée par le grand savant allemand Gottfried Wilhelm Leibniz (1646-1716) deux siècles plus tôt ! Il plaide en sa faveur et obtient une dernière chance pour son surprenant élève.

Pour Turing, la leçon est claire : ce qui vient de lui arriver est plus qu'une simple péripétie dans une vie scolaire inadaptée. Il a trouvé dans les sciences un refuge dont personne ne le délogera. Les mathématiques lui permettront de résister à cette formation de *gentleman* britannique qu'on lui inflige et dans laquelle il ne se reconnaît pas, cette formation qui le détourne de ce qui l'intéresse vraiment : percer le secret de la génération, que ce soit celle des nombres, des plantes… ou même la sienne ?

1927, Sherborne Grammar School. Alan Turing se lie d'amitié avec un élève d'une autre classe âgé d'un an de plus que lui, Christopher Morcom, qui voue le même intérêt que Turing à la science et excelle en chimie. Ils échangent leurs méthodes de calcul en chimie, s'arrangent pour se retrouver ensemble au télescope, discutent longuement… Turing parle de la croissance des plantes, s'interrogeant sur le développement mathématiquement ordonné des pétales sur les tiges. Morcom l'initie à l'astronomie et l'invite à passer l'été dans sa famille.

Turing vénère son nouvel ami. Il admire sa force, son aisance, son aura, lui dont on continue à reprocher les cahiers mal tenus, la graphie illisible, la nullité dans les sports collectifs, la tenue débraillée. Il apprend avec joie que Morcom a, comme lui, décidé de présenter l'examen d'entrée à Cambridge à la fin de son cursus scolaire. Leur plaisir est cependant entaché d'une triste nouvelle : Christopher a contracté la tuberculose en buvant du lait infecté. Chimie et poison, les deux faces d'une même médaille ?

1930, Sherborne Grammar School. Christopher Morcom, admis sur examen à Cambridge un an avant Turing – celui-ci, n'ayant pas eu les notes requises, doit rester encore un an à *Sherborne* –, meurt de tuberculose bovine le 13 février, âgé de 19 ans. Turing écrit à sa mère :

Chère mère, j'ai écrit à Madame Morcom comme tu me l'avais suggéré et cela m'a fait un peu de bien […]. *Je suis sûr que je retrouverai Morcom et qu'il y aura du travail pour nous deux, comme il aurait dû y en avoir pour nous en ce monde. Maintenant que*

Alan Turing (ci-dessus à cinq ans) et son frère John furent élevés par une nourrice en Angleterre, leurs parents vivant à Madras, en Inde, pour des raisons professionnelles. Ils firent tous deux leur scolarité à la Sherborne Grammar School (ci-dessus à droite). Page ci-contre, une pompe à eau de mer pour l'irrigation, devant l'hôtel Spencer à Madras, sur une photographie prise en 1921 par le médecin américain Wilbur Sawyer.

Christopher Morcom, camarade de classe et ami d'Alan Turing qui mourut à 19 ans, marqua profondément ce dernier.

je reste seul, c'est à moi de m'en charger, je ne peux pas le décevoir : je dois y mettre autant d'énergie, sinon autant d'intérêt, que s'il était encore là. Si j'y parviens, je serai plus à même de jouir de sa compagnie qu'aujourd'hui.

S'unir à Christopher en esprit, non pas pour attendre la résurrection de son corps, mais pour s'affranchir pour toujours des corps par la puissance de la pensée, voilà l'une des clés du destin scientifique de Turing. Mais qu'est-ce que la pensée ? Et comment s'affranchit-elle du corps ? Comment la représenter ? Serait-elle un code caché qui insufflerait la vie, une sorte de contre-poison ?

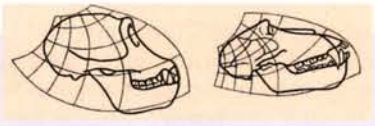
Introduction aux mathématiques

1931-1935, King's College, Cambridge. Pendant ses études au *King's College*, Turing découvre, à travers diverses lectures, le point de vue moderne en mathématiques et en physique, celui du *déterminisme formaliste* défendu par la plus grande figure mathématique des années 1930, David Hilbert *(voir* La mécanisation du monde, *page58)*. Hilbert pense que, en physique comme dans la recherche des fondements des mathématiques, le développement pour lui-même des symboles mathématiques permet de faire varier leurs interprétations sans avoir à les rapporter à un fait expérimental ou intuitif préalable : pour effectuer un calcul algébrique, on applique des règles de transformation et des formules sans s'inquiéter de la signification de chaque étape du calcul. Le calcul se développe par lui-même, et seul le résultat final a un sens par rapport au problème posé.

Hilbert interprète ainsi la mécanique quantique en termes algébriques et topologiques, ce que Turing apprend en octobre 1932 dans le livre de John von Neumann (1903-1957) intitulé *Mathematische Grundlagen der Quantenmechanik (Les fondements mathématiques de la mécanique quantique)*, puis en juillet 1933 dans l'ouvrage *Methoden der Mathematischen Physik (Méthodes mathématiques de la physique)* de David Hilbert et Richard Courant.

Ce point de vue renouvelle la façon dont Turing interprétait la théorie physique, lui qui, dès 17 ans, avait lu les livres de l'astrophysicien Arthur Eddington (1882-1944), qui se faisait le porte-parole d'Einstein dans *The Nature of the Physical World (La nature du monde physique)*, et qui suivait ses cours à Cambridge.

Dans le débat sur le fondement des mathématiques qui agite les mathématiciens du début du XX^{e} siècle, Hilbert critique le point de vue logiciste défendu par deux logiciens, Gottlob Frege et Bertrand Russell : il envisage la notion de nombre comme une donnée primitive dont on peut étudier les *représentations sous forme de signes* sans chercher à réduire le nombre à des caractéristiques ensemblistes, comme le souhaitent Frege et Russell. Ce dernier expose sa théorie dans *Introduction to Mathematical Philosophy (Introduction à la philosophie des mathéma-*



1912 — Alan Mathison Turing naît à Londres le 23 juin. Il est élevé par une famille d'accueil avec son frère John, leurs parents résidant à Madras (Inde), où le père est fonctionnaire colonial.

1917 — Le naturaliste D'Arcy Thompson (1860-1948) publie son livre *Forme et croissance.*

1926 — Alan rejoint John à la *Sherborne Grammar School.*

1927 — Turing se lie d'amitié avec un camarade, Christopher Morcom, dont il tombe amoureux.

1928 — Turing lit le livre de sir Eddington *La structure du monde physique.*

David Hilbert expose les détails de son programme formaliste pour le fondement des mathématiques au Congrès de Bologne.

1930 — Décès de Christopher Morcom, à l'âge de 19 ans.

1931 — Parution de l'article de Kurt Gödel *Sur les propositions indécidables des* Principia Mathematica *et systèmes apparentés.*

Turing entre au *King's College* de Cambridge pour suivre des études de mathématiques.

Le mathématicien allemand David Hilbert (1862-1943,* en haut) *et l'astronome britannique Arthur Eddington (1882-1944,* en bas). *En suivant les cours de physique d'Eddington, Turing s'ouvrit aux sciences ; en lisant Hilbert, il découvrit les questions de la science moderne, notamment la grande quête d'un fondement des mathématiques initiée par ce dernier.

tiques), que Turing lit en mars 1933. Dans la même direction, le jeune homme suit à Cambridge, au printemps 1935, les cours du mathématicien Max Newman, qui décrit les derniers résultats obtenus par la méthode hilbertienne dans les fondements des mathématiques. Parmi eux, ceux du logicien Kurt Gödel, obtenus l'année où Turing est entré au *King's College*, en 1931 *(voir page 63)*.

En physique comme dans les fondements des mathématiques, Turing se rallie au point de vue hilbertien : en mécanique quantique, Turing souscrit à la façon *déterministe* dont von Neumann interprète l'indétermination quantique (la nature est régie par un déterminisme complet auquel nous n'avons pas toujours accès) ; au sujet des fondements des mathématiques, Turing adopte la notion de nombre proposée par Hilbert. Néanmoins, ce sont les cours d'Eddington qui lui donnent l'idée de sa « dissertation » de licence, où il démontre mathématiquement une régularité constatée statistiquement, le théorème de la limite centrale *(voir page 49)*. Turing prouve le résultat en février 1934 sans savoir qu'il a déjà été démontré en 1922.

Juillet 1935, Grantchester, à quelques miles de Cambridge. Depuis plusieurs mois, Turing court tous les jours. Ce nouveau sport lui plaît. Dans la course, son corps fonctionne de façon mécanique et libère sa pensée. Une remarque lancée par Newman à ses élèves, à la fin d'un cours, lui revient souvent en

1932 — Turing lit *Fondements mathématiques de la mécanique quantique* de von Neumann.

1933 — Turing lit *Méthodes de physique mathématique* de Hilbert et Courant, et *Introduction à la philosophie mathématique* de Russell.

Hitler arrive au pouvoir. Les intellectuels juifs et non juifs d'Allemagne, en particulier des mathématiciens de l'école de Hilbert, s'exilent vers l'Angleterre ou les États-Unis.

1934 — Turing obtient sa licence de mathématiques.

1935 — Turing devient *Fellow* de *King's College*.

1936 — Turing prouve négativement le problème de la décidabilité *(Entscheidungsproblem)* proposé par Hilbert dans son programme formaliste de fondement des mathématiques. Il part pour Princeton travailler avec Alonzo Church et John von Neumann.

Mort du roi George V. Son frère George VI lui succède.

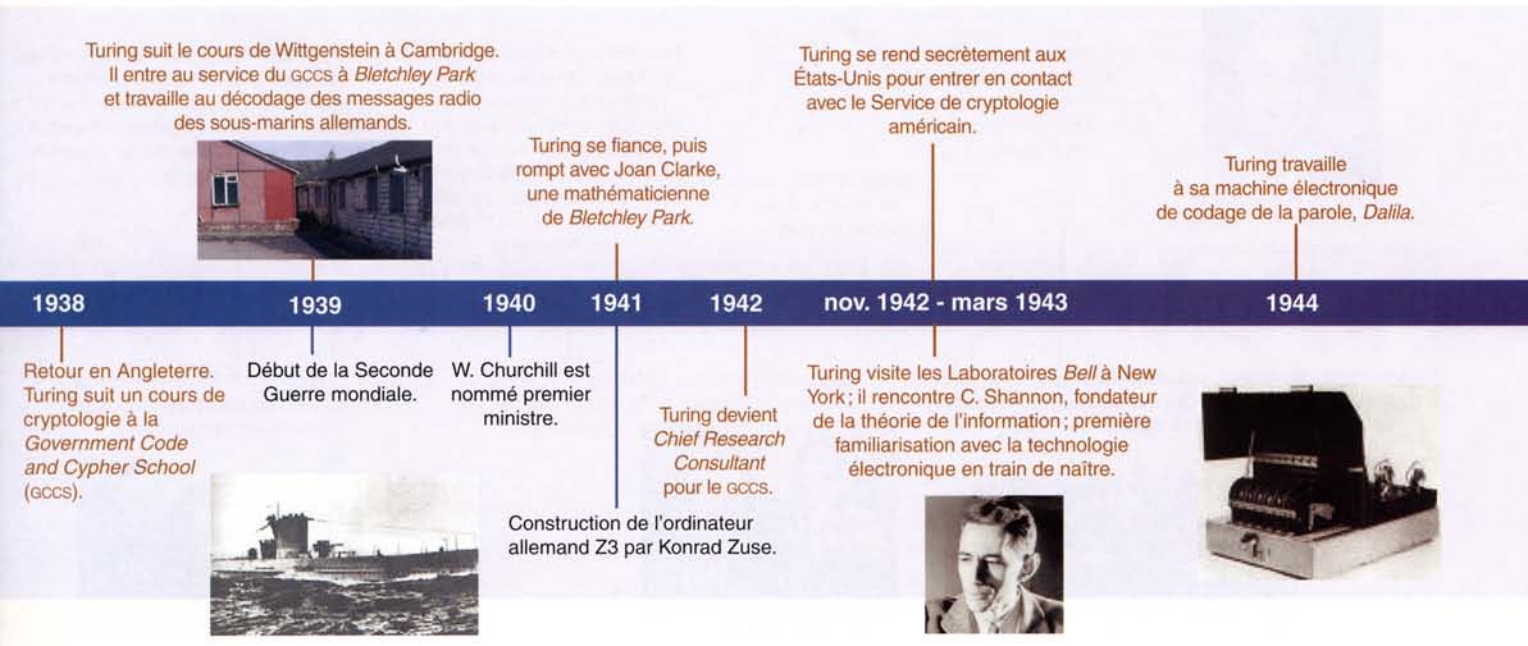
1937 — Turing publie son article *Sur les nombres calculables avec une application à l'*Entscheidungsproblem, dans lequel il expose sa preuve négative, obtenue à l'aide de ce qui deviendra la « machine de Turing ».

Turing obtient la bourse *Procter* à Princeton ; Von Neumann lui propose de devenir son assistant.

Library and Archive Center, King's College Cambridge, AMT/K/7/1 © Crown Copyright

Turing participant à une course du National Physical Laboratory, *en 1946. Il arriva second au 3 miles.*

mémoire : *Messieurs, maintenant que l'on sait grâce à Gödel qu'il n'y a pas d'engendrement complet et systématique de tous les théorèmes d'un système d'axiomes contenant l'arithmétique, on doit se poser la question de savoir si, quel que soit le système d'axiomes, il n'y aurait pas au moins une procédure passe-partout indéfiniment applicable pour trouver quelles sont les propositions que l'on peut* effectivement *déduire d'un système d'axiomes, c'est-à-dire qui déciderait de leur vérité. C'est, en substance, le problème de la* décision *ou* Entscheidungsproblem *posé par Hilbert qui est, à l'heure où je vous parle, encore ouvert.* Turing réfléchit au problème posé : il faut, se dit-il, préciser le rapport entre l'existence de la procédure qui permet de savoir si un théorème dérive bien d'un système d'axiomes, et cette fameuse proposition vraie et pourtant non déductible des axiomes, mise au jour par Gödel il y a cinq ans. Si cette procédure systématique n'existe pas, certaines propositions ne dérivent pas des axiomes puisqu'il n'existe pas de règle systématique pour les engendrer toutes. Ainsi, conclut Turing, l'absence de procédure donne immédiatement le résultat de Gödel… Reste

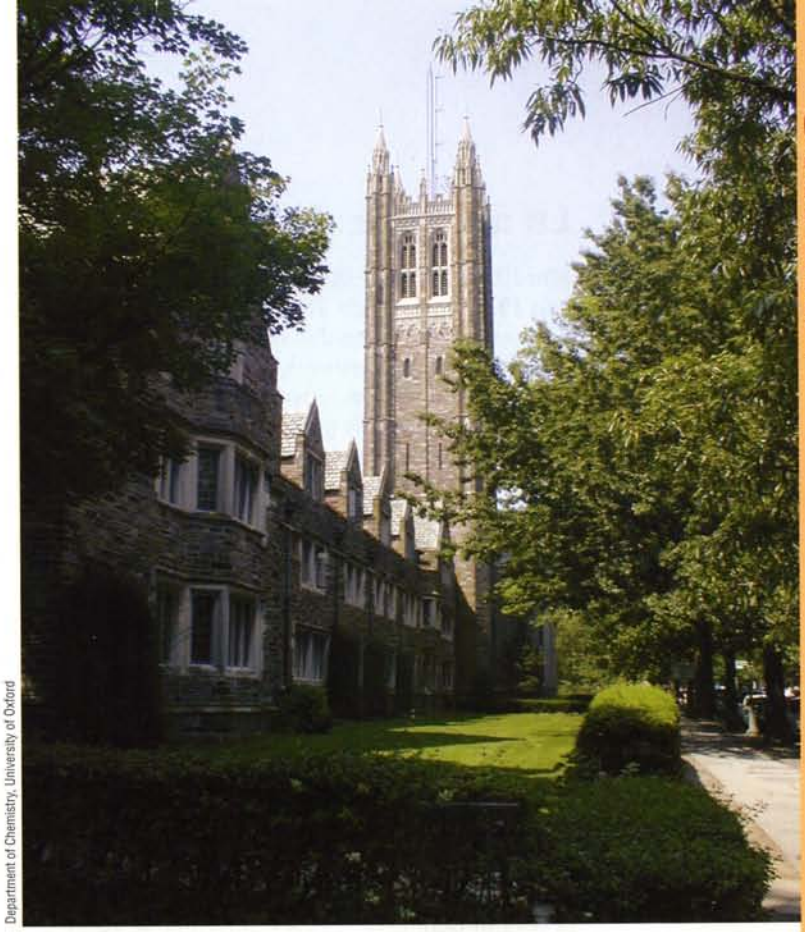


à caractériser *l'absence* d'une procédure systématique et donc à déterminer *ce qu'est* une procédure systématique, ou effective, ou mécanique, comme la qualifient les mathématiciens de l'époque, tels que Hilbert, von Neumann ou Gödel. Ce terme précisé, on saura pourquoi certaines propositions échappent toujours à la déduction.

Turing prend alors le problème au pied de la lettre : il part du principe que ce qu'un *humain* peut déduire est équivalent à ce qu'une *machine* peut déduire. Ainsi, il suffit de décrire convenablement une machine idéale pour préciser ce qu'est une procédure systématique. Et si ensuite on montre que cette machine *ne peut pas* résoudre un problème bien posé, tels que ceux qui sont à la base du déterminisme formaliste comme *prévoir à l'avance le résultat d'un calcul*, alors on saura pourquoi il existe des propositions hors système…

En d'autres termes, Turing renverse la façon dont Gödel pose le problème : il ne s'interroge pas sur les facultés mentales supérieures censées justifier l'existence de propositions inaccessibles, mais s'en tient au mécanique. Son programme est simple : il est une machine et fait les actes que peut faire la machine. Ensuite, il pose le problème censé être soluble par une machine et montre que la machine qui résoudrait ce problème est impossible parce qu'elle donnerait deux résultats contradictoires.

Se servir exclusivement de machines pour montrer qu'il y a comme un au-delà des machines, songe Turing… Le mental reste parfaitement lisible, comme une machine, sauf qu'il n'*agit* pas de la même façon. D'où vient cette différence entre une machine et un être complètement descriptible comme tel ? Voilà le vrai mystère du déterminisme formaliste : toujours descriptible mécaniquement à un instant *t*, la pensée d'un être humain et celle de n'importe quelle machine ne devraient pas se distinguer. Et pourtant



Department of Chemistry, University of Oxford

Un bâtiment de l'Université de Princeton, aux États-Unis, où Turing travailla de 1936 à 1938 aux côtés d'Alonzo Church et de John von Neumann.

si, puisque l'homme sait que la machine ne peut pas *anticiper* le résultat qu'il sait *d'avance* être contradictoire !

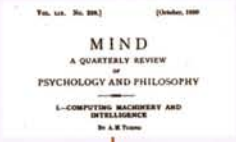
Mais alors, la nature ne serait pas une gigantesque machine, conforme au déterminisme formaliste de Hilbert ? Descartes, Leibniz, Newton, Laplace se seraient-ils trompés ? Et si la croissance dans le monde physique – cette croissance interne aux êtres vivants – est non mécanique, qu'est-ce qui l'engendre ?

1945 — Fin de la guerre en Europe. Schrödinger publie son livre *Qu'est-ce que la vie* ? Von Neumann publie un premier rapport sur les ordinateurs programmables *(Draft Report on the Edvac)* qui s'appuie sur l'article de Turing de 1936.

1947 — Turing se rend à une conférence sur les ordinateurs programmables à Harvard. Il prend une année sabbatique à Cambridge, où il suit des cours de physiologie et de neurologie.

1948 — Turing travaille sur le premier ordinateur dans l'équipe d'informatique de l'Université de Manchester.

1950 — Turing publie son article *Machine à calculer et Intelligence* dans la revue *Mind*.



1951 — Turing est élu *Fellow* de la *Royal Society*.

1952 — Turing publie un article sur les bases chimiques de la morphogenèse. Il est condamné pour homosexualité à la prison ou à la castration chimique. Il choisit cette dernière. Début de sa cure psychanalytique (jungienne).

1953 — Crick et Watson découvrent la structure de l'ADN. Elizabeth II succède à George VI.

1954 — Turing se suicide le 7 juin en mangeant une pomme empoisonnée.

1945 — Turing conçoit le projet de « construire un cerveau ». Il entre au *National Physical Laboratory* pour construire un prototype d'ordinateur, l'*Automatic Computing Engine (ACE)*, et rédige un rapport sur la construction d'un calculateur électronique *(Proposed Electronic Calculator)* citant von Neumann.

La machine de Turing

Mai 1936 - juillet 1938, Université de Princeton. En mai 1936, l'article de Turing sur le problème de la décision, *On Computable Numbers, with an Application to the Enstcheidungsproblem (Sur les nombres calculables, avec une application au problème de la décision)*, est prêt pour la publication. Sa mère l'aide à rédiger en français un descriptif de son article pour les Compte-Rendus de l'Académie des sciences à Paris. L'Académie ne réagit pas.

Par un concours de circonstances malheureux pour Turing, une autre preuve portant sur le même problème, mais obtenue par des moyens différents, est publiée en 1936 par le logicien américain Alonzo Church dans le *Journal of Symbolic Logic*, juste avant l'article de Turing : cela n'empêche cependant ni la publication de son article en janvier 1937 dans les *Proceedings of the London Mathematical Society* ni son départ pour les États-Unis où il fait, précisément sous la conduite d'Alonzo Church, un doctorat de logique mathématique à l'Université de Princeton.

Turing part de Southampton le 23 septembre 1936 et revient définitivement à Cambridge le 18 juillet 1938, après avoir soutenu sa thèse de doctorat en mai sous la direction de Church. C'est d'ailleurs ce dernier qui, dans le compte-rendu qu'il rédige de l'article de Turing pour le *Journal of Symbolic Logic*, emploie, pour la première fois, l'expression de « machine de Turing », promise à un grand avenir.

Octobre 1938, Cambridge. En 1937, Walt Disney a sorti au cinéma *Blanche-Neige et les sept nains*. Turing s'émerveille de la technique déployée par les studios Disney : 24 dessins par seconde, voilà le principe de la vie ! Finalement, songe Turing, Disney, comme lui, a compris qu'il suffit d'avoir la *bonne machine* pour reproduire la vie.

Blanche-Neige s'apprêtant à croquer la pomme empoisonnée. Lors de sa sortie en 1938, le dessin animé de Walt Disney produisit une forte impression sur Turing.

La chanson de la belle-mère de Blanche-Neige préparant sa potion empoisonnée persiste dans la tête du jeune homme : « Plonge la pomme dans le bouillon, que la mort qui endort s'y infiltre. » Le personnage de Blanche-Neige lui semble familier. D'un côté, l'horrible belle-mère et sa pomme empoisonnée, de l'autre, la frêle jeune fille, mais, coup de théâtre, la pomme se coince, le poison ne pénètre pas tout le corps. Il y a moyen de résister au poison…

1939, Cambridge. Pendant le trimestre de Pâques 1939, deux cours s'intitulent « Fondements des mathématiques » à Cambridge : le cours de Turing, dédié à la logique mathématique, et celui du logicien et philosophe Ludwig Wittgenstein, sur la philosophie du langage ordinaire… et la dissolution de la logique mathématique comme discipline fondatrice ! Turing assiste au cours de Wittgenstein. Certains dialogues entre les deux hommes montrent combien leurs points de vue sur la notion de « fondement des mathématiques » diffèrent. Par exemple, ils ne s'accordent pas sur la signification d'une *contradiction* dans un système mathématique : pour Turing, une contradiction manifeste le manque de fiabilité des principes du système formel, car elle remet en question les opérations arithmétiques de base. En revanche, pour Wittgenstein, relever une telle contradiction reviendrait seulement à *sortir de l'usage habituel* que l'on attribue à l'opération arithmétique employée. Pour Turing, la contradiction est le signe d'un manque *d'objectivité* ; pour Wittgenstein, elle est le signe que l'on ne s'est pas *entendu* sur l'usage de la règle. Voici un de leurs échanges :

Turing : « Si quelqu'un prend le symbolisme de Frege et qu'on lui fournit la technique permettant d'exécuter une multiplication au moyen de celui-ci, alors en utilisant le paradoxe de Russell, il pourrait aboutir à une multiplication dont le résultat est faux. »

Wittgenstein : « Cela reviendrait à faire quelque chose que l'on n'appelle pas une multiplication. Vous lui donnez une règle pour la multiplication et, arrivé à un certain point, il peut aller dans deux directions différentes, dont l'une le conduit à faire complètement fausse route. »

Turing abandonne assez vite le cours de Wittgenstein, car l'idée que les mathématiques ne soient qu'un accord sur des règles d'usage sans aucun lien avec leur applicabilité à la physique – Wittgenstein n'envisage pas d'*application* des mathématiques – ne lui convient guère.

Le casse-tête Enigma

1940, Bletchley Park, *Centre secret du Service britannique du chiffre.* « Il n'y a plus que nous » dit Churchill lors d'une visite au centre secret, au tout début du *Blitz*. Et pour le décryptage des codes de la marine allemande, plus que moi, pense Turing. Le jeune

homme a été recruté par le service britannique du chiffre pour décoder les messages cryptés des différents services de l'État allemand. La machine à crypter allemande, *Enigma*, résiste encore à tous les efforts, même si les Polonais ont remporté quelques succès *(voir* De la machine de Turing à l'ordinateur, *page 82)*. Denniston, aux commandes de *Bletchley Park*, ne lui a pas caché les enjeux : *Turing, les Polonais ont commencé à mécaniser le renseignement pour reconstituer le codage envoyé aux sous-marins allemands qui font le blocus, mais ça n'est plus suffisant depuis que la* Kriegsmarine *a compliqué sa méthode de cryptage : on ne peut pas continuer à capter la radio allemande pendant des heures sans rien y comprendre, quand on sait que ce sont des ordres donnés à leurs sous-marins pour couler nos bateaux ! Maintenant qu'on a réussi à capturer une de leurs machines à crypter, l'*Enigma, *il* faut *déchiffrer leurs messages, sinon bientôt, en Angleterre, on ne mangera plus que de la soupe de pissenlits… Il doit y avoir un moyen.*

Turing réfléchit. Un moyen… Le charabia du signal comporte sûrement un ordre, puisque le texte est codé. Or les ordres sont répétitifs. On peut supposer qu'ils ont été rédigés en allemand, donnés à un sous-chef par un gradé en chef, et signés par le chef. Des positions en longitude et latitude dans l'Atlantique Nord, l'expression *Heil Hitler !* et d'autres formules de protocole doivent aussi apparaître régulièrement, le tout encodé lettre par lettre.

Les Allemands ont compris qu'il était imprudent de répéter au début du message la clé du code de la journée. Néanmoins, ils continuent à coder mutuellement deux lettres : si *D* est crypté en *P*, *P* doit être crypté en *D*. Pour le logicien qu'est Turing, c'est tout simplement ridicule : la réciprocité laisse comme une trace, une signature, qu'il s'empresse d'exploiter. Il délègue à une machine le soin d'éliminer toutes les possibilités de combinaisons de lettres où la réciprocité de l'encodage de deux lettres n'est pas assurée dans le message. Couplée à deux autres procédures de son invention, cette méthode permet à Turing de déchiffrer les messages allemands *(voir page 89)*. La difficulté consiste maintenant à déjouer les plans des Allemands sans qu'ils se doutent que leur code a été décrypté. Ils risqueraient de changer les méthodes de cryptage.

Le logicien et philosophe britannique d'origine autrichienne Ludwig Wittgenstein (1889-1951) donnait des cours à Cambridge en même temps que Turing. Ce dernier n'y assista pas longtemps, n'appréciant pas le point de vue du philosophe sur les mathématiques.

***Le manoir de* Bletchley Park, *où siège le Service britannique du chiffre. Turing y travailla deux ans à déchiffrer le mystérieux code de la machine à crypter allemande* Enigma.**

Jack Harper

White house visitors center

Churchill et Roosevelt en conversation devant la Maison-Blanche. Quelques années plus tôt, une telle rencontre n'étant pas envisageable, Turing conceva une machine à crypter la voix humaine pour permettre aux deux grands hommes de converser en toute sécurité.

À *Bletchley Park*, Turing se lie d'amitié avec une jeune femme, Joan Clarke. Celle-ci a tout pour lui plaire : bonne en mathématiques, elle aime discuter avec lui et accepte ses tendances. L'épousera-t-il ?

7 novembre 1942 - 31 mars 1943, États-Unis. Turing est envoyé en secret aux États-Unis par le Service britannique du chiffre : le blocus de l'Angleterre prend des proportions jamais atteintes et il faut y remédier au plus vite. La situation se rétablit pendant son séjour outre-atlantique grâce à un certain nombre de défaillances logiques dues à des « améliorations » du système de codage allemand. Le déchiffrement s'effectuera de façon régulière à partir de 1943. Turing a pour mission d'intervenir au mieux dans la crise des sous-marins allemands en partageant son expertise avec les équipes américaines chargées du décryptage des codes japonais. Il doit aussi participer à l'élaboration d'un projet de codage de la voix humaine qui permettra une communication téléphonique cryptée entre Churchill et Roosevelt.

Il travaille aux Laboratoires *Bell* de New York où la technologie électronique commence à être maîtrisée ; à partir de janvier 1943, il y rencontre régulièrement Claude Shannon, le fondateur de la théorie de l'information. Turing déclare à Shannon qu'il lui semble possible de construire des « machines qui pensent ». Shannon, de son côté, fait part à Turing de son idée d'imiter, avec des machines électroniques, le cerveau humain dans toutes ses fonctions, y compris esthétiques. Turing a sans doute aussi été consultant sur le projet de construction de la bombe atomique. Il rentre en Angleterre en mars 1943, seul civil au milieu d'hommes de troupe, à bord d'un bateau se déroutant sans cesse au rythme des ordres de route reçus de *Bletchley Park*…

1944, Hanslope Park, laboratoire dépendant des services secrets. À l'aide de composants électroniques, Alan Turing construit une machine à crypter la voix humaine qu'il nomme Dalila, comme cette femme de l'Ancien Testament qui sut « mentir aux hommes ». Avec cette machine, il crypte le discours de la victoire de Churchill : il découpe la voix en échantillons de fréquences, neutralise toute différence de fréquence, puis retraduit le nouveau signal en une autre fréquence aléatoire. Adieu la voix humaine : ce n'est plus qu'un jeu de fréquences physiques contrôlées électroniquement.

> « Nous devrions pouvoir construire des machines qui pensent. »

Les progrès de l'électronique donnent à Turing une autre idée : puisque la technologie électronique permet de maîtriser la vitesse et la mémoire, pourquoi ne pas construire un cerveau inspiré de la « machine de Turing » ? Ce cerveau, adulte d'emblée, court-circuiterait le problème de la croissance interne…

Naissance de l'ordinateur

1945-1947, National Physical Laboratory. Les énormes calculs que l'industrie de guerre exige, que ce soit en balistique, dans la gestion des stocks ou dans le *Manhattan Project* visant la construction de la bombe atomique, ont favorisé le lancement de deux projets de construction de calculateurs électroniques, l'un aux États-Unis, l'autre en Angleterre. Deux figures capitales de la recherche mathématique se retrouvent ainsi associées autour du problème du calcul mécanisé : John von Neumann côté américain et Alan Turing côté anglais. Le 30 juin 1945, un premier rapport signé von Neumann voit le jour, le *Draft Report on the EDVAC (Rapport préliminaire sur l'Electronic Discrete Variable Automatic Computer)*, dans lequel le plan d'un calculateur électronique est décrit d'un point de vue théorique, sur la base de l'article de Turing de 1936, *On Computable Numbers*. Côté anglais, Turing est engagé le 1er octobre 1945 par le *National Physical Laboratory*, situé à Teddington : il rédige lui aussi un rapport, intitulé *Proposed Electronic Calculator (Proposition de calculateur électronique)*, en citant celui de von Neumann. En janvier 1947, une conférence est organisée à Harvard aux États-Unis. Turing, seul chercheur anglais parmi les participants, y retrouve von Neumann. Leurs deux projets, bien que très proches, ne sont pas équivalents : celui de von Neumann, orienté vers le calcul, associe plusieurs types de machines à calculer dédiées à des tâches particulières, tandis que Turing remplace les machines particulières par de la programmation, exécutable sur une machine unique, un ordinateur.

1947, National Physical Laboratory. Turing a la tête ailleurs. Il a présenté son projet d'ordinateur en 1946, et celui-ci se construit lentement. Le laboratoire a besoin non plus d'un concepteur, mais de programmeurs et d'ingénieurs. Turing réfléchit donc à une question qui l'intrigue depuis longtemps : il veut comprendre ce que l'on entend par « processus intelligent ». Pourquoi ne pas essayer d'appréhender expérimentalement

***Les premières pages des deux rapports sur la construction d'un calculateur électronique qui marquèrent la naissance de l'ordinateur : le* Proposed Electronic Calculator *de Turing* (à gauche), *et le* Draft Report on the EDVAC *de Von Neumann* (à droite).**

2 **Proposal for Development in the Mathematics Division of an Automatic Computing Engine (ACE)**
A. M. Turing

Proposed Electronic Calculator

Part I. Descriptive Account

1. Introductory 2. Composition of the Calculator 3. Storages 4. Arithmetical Considerations 5. Fundamental Circuit Elements 6. Outline of Logical Control 7. External Organs 8. Scope of the Machine 9. Checking 10. Time-Table, Cost, Nature of Work, Etc.

1. Introductory

Calculating machinery in the past has been designed to carry out accurately and moderately quickly small parts of calculations which frequently recur. The four processes addition, subtraction, multiplication and division, together perhaps with sorting and interpolation, cover all that could be done until quite recently, if we except machines of the nature of the differential analyser and wind tunnels, etc. which operate by measurement rather than by calculation.

It is intended that the electronic calculator now proposed should be different in that it will tackle whole problems. Instead of repeatedly using human labour for taking material out of the machine and putting it back at the appropriate moment all this will be looked after by the machine itself. This arrangement has very many advantages.

(1) The speed of the machine is no longer limited by the speed of the human operator.
(2) The human element of fallibility is eliminated, although it may to an extent be replaced by mechanical fallibility.
(3) Very much more complicated processes can be carried out than could easily be dealt with by human labour.

Once the human brake is removed the increase in speed is enormous. For example, it is intended that multiplication of two ten figure numbers shall be carried out in 500 μs. This is probably about 20,000 times faster than the normal speed with calculating machines.

It is evident that if the machine is to do all that is done by the normal human operator it must be provided with the analogues of three things, viz. firstly, the computing paper on which the computer writes down his results and his rough workings; secondly, the instructions as to what processes are

http://www.wps.com/projects/EDVAC/index.html

First Draft of a Report on the EDVAC

John von Neumann

1.0 Definitions

1.1 The considerations which follow deal with the structure of a *very high speed automatic digital computing system*, and in particular with its *logical control*. Before going into specific details, some general explanatory remarks regarding these concepts may be appropriate.

1.2 An *automatic computing system* is a (usually highly composite) device, which can carry out instructions to perform calculations of a considerable order of complexity—e.g. to solve a non-linear partial differential equation in 2 or 3 independent variables numerically.

The instructions which govern this operation must be given to the device in absolutely exhaustive detail. They include all numerical information which is required to solve the problem under consideration: Initial and boundary values of the dependent variables, values of fixed parameters (constants), tables of fixed functions which occur in the statement of the problem. These instructions must be given in some form which the device can sense: Punched into a series of punchcards or on teletype tape, magnetically impressed on steel tape or wire, photographically impressed on motion picture film, wired into one or more, fixed or exchangeable plugboards—this list being by no means necessarily complete. All these procedures require the use of some code, to express the logical and the algebraical definition of the problem under consideration, as well as the necessary numerical material (cf. above).

Once these instructions are given to the device, it must be able to carry them out completely and without any need for further intelligent human intervention. At the end of the required operations the device must record the results again in one of the forms referred to above. The results are numerical data; they are a specified part of the numerical material produced by the device in the process of carrying out the instructions referred to above.

University of Manchester

L'équipe de l'Université de Manchester qui construisit l'ordinateur Manchester Mark I. De gauche à droite : D. Edwards, F. C. Williams et Tom Kilburn.

cette notion, par exemple en regardant si la future machine sera capable de jouer aux échecs ?

À *Bletchley Park*, Turing et ses collègues comparaient déjà les positions des bateaux et des sous-marins à des pièces de jeu d'échec ou de go ; ils essayaient de deviner, à l'aide des *Bombes*, les intentions de ceux d'en face, cachés derrière leur *Enigma*. Les machines n'étaient pour eux que des intermédiaires. Pourtant, à bien y réfléchir, il s'agissait plutôt d'un combat entre machines : l'esprit de l'adversaire n'était que ce qui, dans les signaux, cessait d'apparaître aléatoire grâce aux machines de décryptage. Pourquoi, dans ce cas, ne pas essayer de jouer *directement* contre une machine convenablement programmée, voire faire jouer des machines les unes contre les autres pour tester la force des programmes ?

La question de Turing est simple : peut-on comparer le cerveau à un ordinateur programmable ? Sa solution est cependant complexe, car la nature n'est pas une simple machine programmée à l'avance. Le vivant ne semble-t-il pas s'auto-organiser *sans code* ? Turing se décide : il prend une année sabbatique à Cambridge pour réfléchir au problème.

1947-1948, Université de Cambridge. Le 30 septembre 1947, Turing retourne au *King's College*, où il reprend la position de *Fellow* (chargé de cours) qu'il avait abandonnée du fait de la guerre. Il suit des cours de physiologie. Son intérêt déjà ancien pour les phénomènes de croissance dans les ordres végétal et animal se développe et deviendra son champ de recherche à partir de 1951.

1948 –1950, Université de Manchester. En octobre 1948, Turing rejoint Max Newman à l'Université de Manchester. Ce dernier y a fondé une équipe de recherche sur les calculateurs électroniques. Le premier programme au monde a fonctionné sur l'ordinateur de Manchester le 21 juin 1948, et non au *National Physical Laboratory*, dont le projet n'entrera en service qu'en août 1950. Turing commence à programmer, tant pour faire du calcul numérique que pour concevoir des modèles de croissance de plantes. Les premiers débats publics sont organisés autour de la question de la mécanisation de l'intelligence : Turing participe à plusieurs d'entre eux et rédige un article pour une revue philosophique, où il soutient la possibilité de la mécanisation de l'intelligence tout en soulignant la différence radicale de fonctionnement physique entre l'ordinateur et le cerveau.

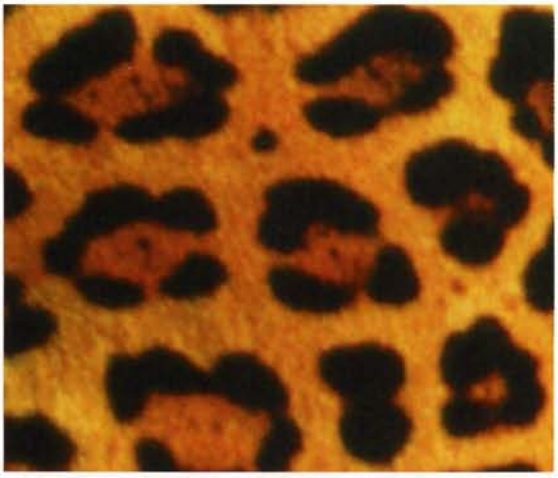
Homosexuel, donc coupable

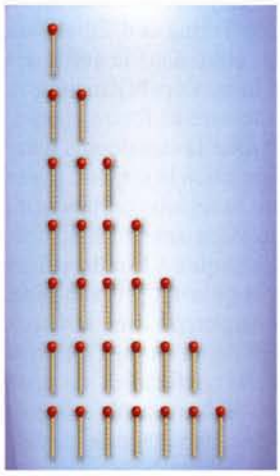
1951, Manchester. Turing est élu *Fellow* de la *Royal Society*. En décembre 1951, une aventure avec un homme, mineur au moment des faits, conduit à son arrestation le 7 février 1952.

1952, Tribunal de Manchester. Un juge doit se prononcer au sujet d'un homme ayant plaidé coupable à l'accusation de « pratiques indécentes réitérées en compagnie d'un autre homme ». Le juge est perplexe : le prévenu – un dénommé Alan Turing – n'a rien nié. Il a bien, et de longue date, des pratiques homosexuelles et ne cherche pas à s'en cacher. Toutefois, cet homme à la voix mal posée n'a pourtant rien d'un marginal : c'est un *Fellow* de la *Royal Society*, un membre de la plus célèbre société scientifique du monde, celle des Newton, des Huygens et de bien d'autres savants. C'est aussi, aux dires des témoins venus à la barre, un homme qui a participé de façon remarquable à l'effort de guerre puisqu'il a percé le code secret de la marine allemande, un homme, enfin, qui travaille dans un laboratoire de pointe au projet d'un « cerveau électronique » couvert par le secret défense et qui semble avoir par ailleurs une vie sociale des plus rangées, dévouée au travail scientifique. D'ailleurs, dans sa naïveté, l'homme croit que la loi de 1865 punissant les pratiques homosexuelles est sur le point de changer et que ce qui est considéré comme un crime aujourd'hui ne le sera plus demain…

On n'anticipe pas les évolutions de la loi. Néanmoins, on peut proposer un choix, car la science vient au secours de l'application des peines dans cette société si ouverte au progrès technologique. Pour éviter l'emprisonnement, puisque les témoins lui assurent qu'il est indispensable aux recherches en cours, le juge a alors l'idée de proposer à Turing un traitement hormonal d'un an, comme le recommandent les experts psychiatres. La solution du traitement hormonal a bien sûr des effets physiques collatéraux (absence d'érection, seins qui poussent, disent les experts), mais ceux-ci ne seront que temporaires et, surtout, ne sont pas le but visé : c'est à la normalisation de la libido que la justice travaille, pour le propre bien de l'intéressé et pour la paix sociale…

Face à ce qui lui est imposé, Turing, une fois de plus, s'en sort par la recherche scientifique : ne





Dans les années 1950, Turing conçut un modèle décrivant un des processus participant à la genèse des formes du vivant, telles que les motifs du dos d'un léopard** (à gauche). **À droite, le jeu de Marienbad, cas particulier du jeu de Nim, jeu à deux joueurs sans information cachée. Dans le jeu de Marienbad, chaque adversaire prend tour à tour le nombre qu'il veut d'allumettes, mais dans une rangée seulement. Celui qui prend la dernière allumette a gagné. En 1953, les visiteurs du festival de South Kensington peuvent se mesurer à une machine jouant à un tel jeu. L'ère du jeu informatique a commencé…

pouvant travailler en prison, il choisit le traitement. Que ces prétendus experts lui injectent ce qu'ils veulent, pense-t-il, qui sont-ils pour croire maîtriser l'esprit en s'en prenant au corps ? La pensée, pour fonctionner, n'a pas besoin d'un état matériel particulier du corps. Les programmes du cerveau électronique ne sont-ils pas exécutables quelle que soit la machine matérielle ?

Et pourtant… La mathématique sera-t-elle assez forte pour libérer sa pensée de toute forme de dépendance, comme du temps de la *grammar school* de *Sherborne* ? Les mathématiques pourront-elles le sauver encore ? Tout son travail de libération de soi – de son corps, de son milieu, de sa vie tracée d'avance – n'a-t-il pas été vain ? Peut-on vraiment concevoir une pensée indépendante du corps ? Et si au contraire la pensée d'un individu dépendait de l'histoire particulière de ce bout de matière qu'est son cerveau ? Et si la trajectoire d'une vie était avant tout une histoire singulière ? Mais alors, l'idée d'une pensée exécutable sur n'importe quelle machine n'est-elle pas une illusion ? Suffit-il vraiment de remplacer le physique par la description écrite de son fonctionnement pour le circonscrire ? Il faudrait tout reprendre, tout retravailler depuis la naissance… mais y a-t-il encore le temps ?

1952, Londres. La *Royal Society* publie dans ses annales de biologie théorique un article sur les « bases chimiques de la morphogenèse » signé par Alan Turing. Il a 40 ans. L'article décrit comment deux substances idéales, dites « morphogènes », se diffusant de façon aléatoire dans un réseau, parviennent, dans des cas particuliers mathématiquement déterminables, à des états d'équilibre laissant apparaître une structure ordonnée – une forme – constituée d'ondes stationnaires. Aucun code n'explique l'apparition de ce phénomène : c'est une forme auto-organisée. L'article fournit une explication des taches sur le pelage des animaux ou des bandes colorées sur les coquillages. Turing a rejoint ce vieux rêve né d'un livre reçu à dix ans pour Noël, celui de penser le vivant à l'aide des mathématiques, de la physique et de la chimie.

1953, au festival de Grande-Bretagne à South Kensington. Sur un stand du festival, le fabricant de l'ordinateur de Manchester a installé un prototype de la machine jouant au jeu de Nim. La machine attire du monde et suscite des expériences assez comiques. Ainsi, la *Société pour la Recherche Psychique* monte un stand à côté de la machine pour voir si cette dernière est influençable par télépathie. Après l'échec de l'expérience, les membres de la société – en majorité des vieilles dames, dit Turing – persistent dans leur idée. Ils testent si eux-mêmes ne sont pas influençables par la machine en tentant de deviner à distance comment, dans l'autre stand, la partie évolue… L'expérience est aussi un échec : les machines sont beaucoup moins coopératives que les être humains en matière d'influence télépathique, conclut Turing !

La pomme

1954, Wilmslow, près de Manchester. Depuis quelques mois, Turing est soumis à un traitement hormonal qui doit neutraliser sa libido. Sous l'effet des hormones, le voilà transformé en ce qu'il imagine être une presque-femme. Est-il encore lui-même, malgré les apparences ? Tout doit rentrer dans l'ordre bientôt, mais s'il s'était trompé ? Si la pensée et le corps entretenaient des liens plus profonds que ceux qu'il avait imaginés ? Mais alors, la pensée ne maîtriserait pas le corps… Christopher aurait-il à jamais disparu ?

Un samedi, à la foire de Blackpool où il s'est rendu en compagnie de son psychanalyste et de ses enfants, Turing entre dans la tente d'une diseuse de bonne aventure. Il ressort de la consultation tremblant et livide. Que lui ont dit les cartes ?

Le 7 juin au matin, la femme de ménage trouve Alan Turing, une légère écume aux lèvres, raide et froid dans son lit, une pomme à moitié mangée sur sa table de chevet. Le fruit avait macéré dans du cyanure. Contrairement à la pomme de Blanche-Neige, la sienne ne s'est pas coincée dans sa gorge, c'est lui-même qui l'y a enfoncée, ultime acte de maîtrise envers un corps qui n'était déjà plus le sien. ■

Regarder grandir

Dans les années 1920-1930, le jeune Alan Turing, élève réfractaire de la Sherborne School, *se découvre une passion pour les sciences en étudiant la nature qui l'entoure : la croissance des plantes, la chimie du soda…*

Alan Mathison Turing est né le 23 juin 1912 à Londres, second fils de Julius Mathison Turing et d'Ethel Sara Stoney. En 1896, Julius était entré dans le corps des fonctionnaires coloniaux en Inde, alors britannique et en 1907, en Inde, il avait rencontré sa future épouse, issue d'une famille aisée établie là-bas depuis deux générations. Le père de Turing passa la majeure partie de sa vie professionnelle dans la région de Madras où naquit le fils aîné du couple, John, en janvier 1908. Alan, lui, fut conçu en Inde, mais naquit à Londres, lors d'un congé de plusieurs mois qu'avait pris son père. La mère de Turing resta en Angleterre avec ses deux fils jusqu'en septembre 1913. Aussi étrange que cela puisse paraître aujourd'hui, elle confia alors ses deux fils, dont Alan âgé d'à peine un an, à un couple de retraités habitant St Leonards-on-Sea, un village de bord de mer jouxtant Hastings, et repartit en Inde auprès de son mari. Elle ne retourna en Angleterre que du printemps à l'automne 1915 ; Alan avait trois ans. En août 1916, le couple Turing revint en Angleterre, mais Julius rejoignit son poste en Inde peu de temps après ; ils décidèrent lors de ce séjour qu'Ethel resterait en Angleterre auprès de leurs enfants à cause de la guerre. Elle ne repartit en Inde qu'en 1919. En 1920, le père de Turing démissionna de son poste à l'occasion d'un conflit interne concernant son avancement, mais, pour des raisons fiscales, il ne devait pas résider en Angleterre plus de six semaines par an jusqu'à sa retraite en 1926 : M. et Mme Turing s'installèrent à Dinard, en Bretagne. Leurs fils venaient les voir à Noël et à Pâques, la famille ne se réunissant en Angleterre que pendant les vacances d'été. Celles-ci se déroulaient aussi en Allemagne, dans les pays scandinaves et en Suisse : pour la famille Turing, l'Europe, vue d'Inde, possédait un « air de famille » et ses déplacements y étaient nombreux. Néanmoins, les fils Turing, comme ceux de nombreux fonctionnaires coloniaux britanniques, ne virent leurs parents que très occasionnellement, grandissant tout d'abord au sein de familles d'accueil, puis d'institutions scolaires.

Il existait une tradition scientifique masculine dans la famille d'Alan Turing : du côté paternel, son grand-père avait étudié les mathématiques au *Trinity College* de Cambridge – le collège scientifique le plus réputé de l'Université, celui de Francis Bacon, d'Isaac Newton et de Charles Babbage – avant de devenir pasteur ; du côté maternel, George Johnstone Stoney (1826-1911), un cousin de son grand-père, membre de la *Royal Society*, avait inventé le terme « électron », ce dont la mère de Turing était très fière. L'intérêt de Turing pour les matières scientifiques semble s'être éveillé très tôt, à l'occasion de questions concernant le repérage dans l'espace (il en garda le goût des cartes et du repérage des étoiles). Cela ne l'empêcha pas d'apprécier aussi, à l'école élémentaire, l'apprentissage du français, sans doute à cause de l'endroit où ses parents résidaient depuis 1920.

Les années d'apprentissage

À partir de 1922, Turing est envoyé en internat à l'école primaire *Hazelhurst*, comme son frère John. Il s'intéresse déjà à l'algèbre et confie à son frère que le professeur de mathématiques a « donné une fort mauvaise impression de ce qui est signifié par x ». Pour le Noël de 1922, alors qu'il a dix ans, Turing reçoit un livre de vulgarisation qui semble avoir eu une influence considérable sur son destin scientifique, *Natural Wonders Every Child Should Know (Les merveilles natu-*

Turing regardant les marguerites grandir au lieu de jouer au hockey, sur un croquis dessiné par une de ses connaissances dans les années 1920. Page ci-contre, la foule anglaise acclamant en 1911, dans une rue de Willenhall, West Midlands, le couronnement du roi George V.

les marguerites

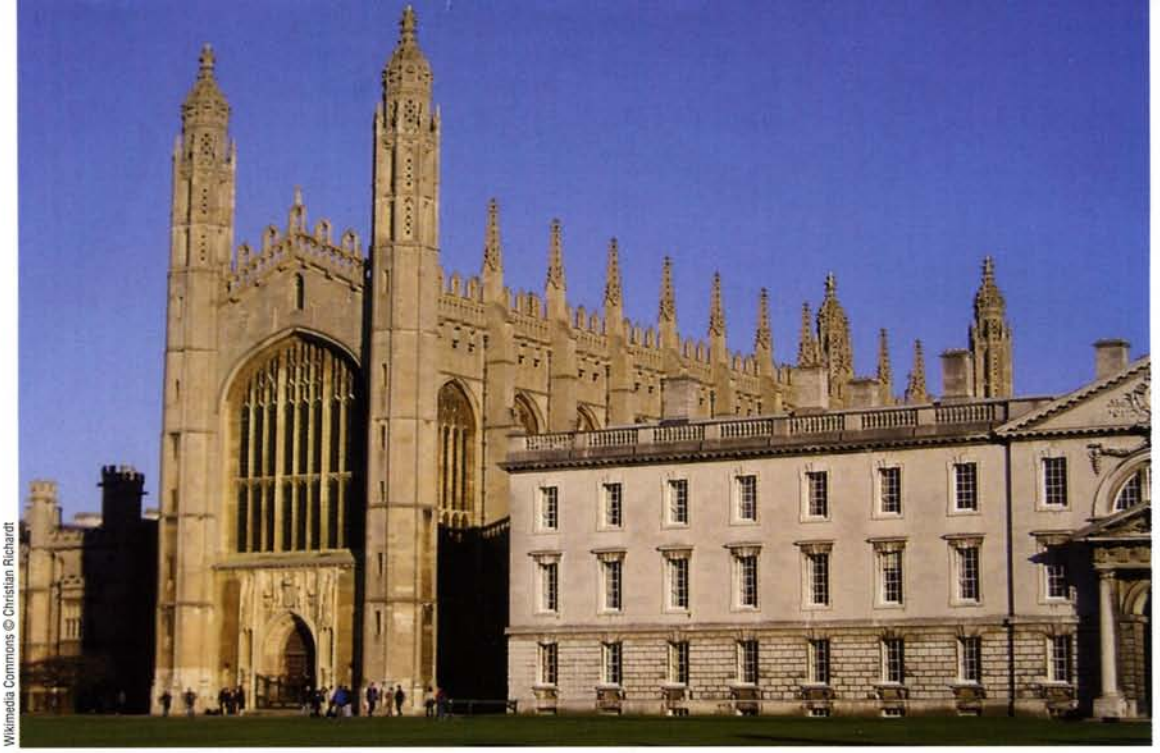
relles que tout enfant devrait connaître) dans lequel sont décrits d'un point de vue mécaniste plusieurs phénomènes physiologiques, en particulier celui de la croissance de l'embryon (sans que soit mentionné le processus de la fécondation). Ainsi, ce sont plus les mathématiques appliquées aux sciences de la nature – surtout la chimie et la biologie – que les mathématiques elles-mêmes qui ont capté l'attention de Turing.

Son intérêt pour la chimie semble avoir pris forme en 1924, quand ses parents lui eurent acheté une boîte de chimiste tout en l'autorisant à l'utiliser dans la cave à charbon de leur maison de Dinard. La chimie organique l'attire tout particulièrement : il essaye d'extraire de l'iode à partir d'algues récoltées sur la plage de Dinard, puis s'intéresse peu à peu aux formules topologiques des molécules organiques. En septembre 1924, il montre à sa mère la recette de fabrication du soda, qui permet, lui dit-il, de comprendre l'effet du sang dans les poumons.

En 1926, M. et Mme Turing envoient le jeune Alan, âgé de 14 ans, dans une école privée du Dorset d'assez bonne réputation, la *Sherborne Public School*, et non au *Marlborough College* comme son frère, ce dernier ayant averti ses parents qu'Alan, compte tenu de son caractère, ne pourrait pas s'y adapter. Sur le ferry-boat qu'il prend à Saint Malo le 3 mai 1926 pour se rendre à son école, Alan apprend qu'une grève générale s'est déclenchée en Angleterre, qui paralyse tous les transports publics. Laissant ses bagages à Southampton, il loue une bicyclette et fait, en deux jours et une nuit d'hôtel, les 60 miles qui le séparent de sa nouvelle école… au grand étonnement de l'administration de *Sherborne*, peu habituée à ce genre d'initiative de la part de ses élèves.

Dans sa *public school*, l'accent est davantage mis sur les humanités et les valeurs propres à l'éducation de *gentleman* que sur les matières scientifiques ; or Turing semble avoir déjà perdu à cette époque tout intérêt pour les disciplines littéraires, y compris l'apprentissage du français. Dispensé de grec pour inaptitude chronique, incapable de pratiquer les sports d'équipe exigés des élèves – il préfère étudier les plantes de la pelouse utilisée pour les sports collectifs ! –, il est, en 1927, en instance de redoublement, voire d'exclusion, faute d'un investissement suffisant dans les matières littéraires. Son aspect peu soigné, dans ses devoirs écrits comme sur sa personne, n'arrange rien. Alan sort de ce mauvais pas en montrant à son professeur de mathématiques un résultat (le développement de la fonction arctangente, *voir page 35*) que celui-ci considère comme « génial » vu son âge et qui l'incite à plaider la cause de son étrange élève auprès du directeur de l'établissement.

L'année 1927 est aussi celle d'un nouveau départ, tant d'un point de vue scientifique qu'affectif : au début de l'année, Alan Turing se découvre homosexuel en tombant amoureux d'un camarade d'une classe supérieure, Christopher Morcom, qui partage avec lui un goût prononcé pour les matières scientifiques, en particulier pour la chimie. De 1927 à 1930, la vie de Turing prend un tour nouveau : le jeune homme, sans avouer ses sentiments, trouve en Christopher Morcom un interlocuteur complice et féru de science, même si ce dernier est loin de soupçonner la passion qu'il engendre.



Wikimedia Commons © Christian Richardt

Un des bâtiments du King's College, *à Cambridge, où Alan Turing fit ses études de mathématiques de 1931 à 1933, et où il obtint en 1935 un poste d'enseignant-chercheur. Ci-dessous, son professeur de mathématiques en 1933, Max Newman (1897-1984), qui l'introduisit aux grands problèmes de la logique mathématique de l'époque, dont celui de la décision, qu'il résolut trois ans plus tard.*

Contrairement à Turing, Christopher Morcom n'a pas de difficultés particulières avec les matières littéraires et est à l'aise dans la vie scolaire en général. D'un tempérament joueur, il invente de nombreux jeux, tentant en particulier de faire croire à son interlocuteur des choses vraisemblables, mais fausses. Ainsi, lorsque les deux amis passent l'examen d'entrée du *Trinity College* – le prestigieux collège où le grand-père paternel de Turing a fait ses études de mathématiques –, Christopher fait croire à Alan, avec qui il s'est rendu à Cambridge, qu'ils ont changé de fuseau horaire et qu'il doit décaler sa montre de 20 minutes… Turing se rend compte de la supercherie à la grande joie de Morcom. Ce dernier est reçu à l'examen et Turing échoue.

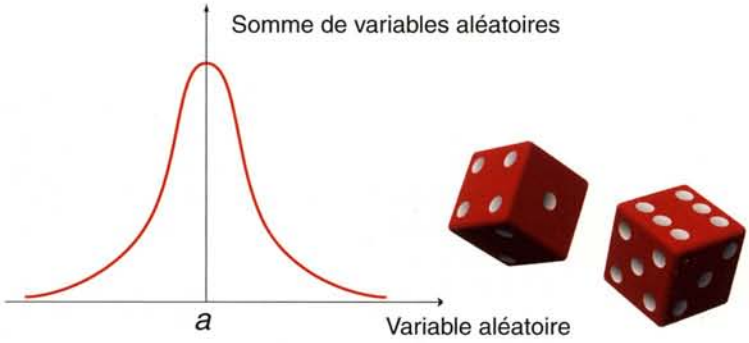
Morcom part pour *Trinity* le trimestre suivant, mais, atteint depuis l'enfance de tuberculose bovine après l'ingestion d'un lait infecté, Christopher Morcom meurt lors de sa première année à Cambridge, le 13 février 1930, à l'âge de 19 ans, alors que personne n'avait averti Alan Turing de sa maladie. Il écrit à sa mère qu'il lui faut maintenant assumer seul le destin scientifique promis à Morcom. Peu après, il élabore pour la mère de son ami disparu une théorie de la migration de l'esprit, selon laquelle l'esprit serait capable de se détacher du corps qu'il habite au moment du décès et d'en intégrer un autre ultérieurement. Turing repasse l'examen d'entrée à Cambridge un an plus tard et est reçu au *King's College*, où il entre en septembre 1931, l'année de la parution des travaux de Gödel.

http://www.rutherfordjournal.org/

Les *Tripos*

À Cambridge cette année-là, Turing fait partie des 85 étudiants qui commencent leur cursus mathématique de trois ans, appelé *Tripos* dans le jargon de l'Université. Tout en poursuivant avec enthousiasme ses études de mathématiques, Turing étend ses intérêts à la physique, en particulier à la mécanique quantique, grâce aux livres de Schrödinger, Heisenberg et von Neumann qu'il a reçus en prix en quittant *Sherborne*. À ces cours s'ajoutent ceux donnés par des scientifiques allemands qui ont pris le chemin de l'exil à l'arrivée au pouvoir des nazis en Allemagne, à partir de 1933 : Max Born en mécanique quantique, Richard Courant sur les équations différentielles, puis, un peu plus tard, John von Neumann sur les fonctions presque périodiques. L'avènement du nazisme a en particulier décapité toute l'école mathématique et physique de Göttingen, centrée autour de la figure de Hilbert : elle vit désormais éparpillée, notamment en Angleterre (Schrödinger, Born et Courant) et aux États-Unis (Einstein, Weyl, von Neumann, Gödel, Lefschetz, Noether).

Turing suit aussi en 1933 les cours de méthodologie scientifique de l'astrophysicien sir Arthur



Stephen Silver - Open Clip Art Library

Lorsqu'on lance un dé une fois, on a autant de chances d'obtenir chaque face. En revanche, quand on lance deux dés, on a plus de chances d'obtenir le nombre 7 que les autres. De même, dans certaines conditions déterminées par le théorème de la limite centrale, une somme de variables aléatoires, tel le résultat d'un lancer de dé, converge vers une variable aléatoire a quand elle tend vers l'infini. La distribution de la somme en fonction des variables aléatoires prend alors la forme d'une cloche.

Eddington, dont il a lu les livres de vulgarisation scientifique à *Sherborne*. Celui-ci insiste sur la place du calcul des probabilités dans la démarche scientifique et s'applique à en dégager la nature, surtout depuis que l'apparition de la mécanique quantique a radicalisé la présence de l'aléatoire dans la connaissance de la nature physique. Cette vision des sciences suggère à Turing un thème de « dissertation » de fin de cursus : le théorème de la limite centrale.

La mère de Turing rapporte que celui-ci lui a dit avoir démontré le théorème de la limite centrale à 15 ans, en 1927, même si la preuve qu'il présente à la fin de son premier cycle universitaire date de février 1934. Turing ignore que ce théorème a déjà été démontré par le mathématicien suédois Lindeberg dans un article de 1922, publié en allemand dans le *Mathematische Zeitschrift*. Ce théorème justifie le caractère de « courbe en cloche » d'une distribution de probabilité.

Lors d'une expérience aléatoire comme le lancer d'un dé, on peut associer l'apparition d'une face du dé à un nombre (généralement gravé sur la face du dé). Le nombre apparaît alors comme le résultat numérique d'un « mécanisme » non déterministe, le lancer physique du dé. Dans le cas général, il est possible de construire une fonction qui fait correspondre l'apparition d'un événement à un nombre réel (cette fonction est appelée variable aléatoire).

Dans l'exemple du lancer de dé, on appelle par extension « variable aléatoire » le résultat du lancer, c'est-à-dire 1, 2, 3, 4, 5 ou 6. Tous ces résultats sont équiprobables. En revanche, si on lance plusieurs dés, la somme des lancers n'est plus équiprobable. Par exemple, pour deux dés, il y a 1 chance sur 36 que la somme des lancers soit 2, mais 6 chances sur 36 qu'elle soit égale à 7 : une convergence se dessine donc vers une variable aléatoire particulière (ici le résultat 7). Que se passe-t-il dans le cas où la somme des variables aléatoires tend vers l'infini ? Le théorème de la limite centrale répond à cette question en donnant les conditions de convergence vers une variable aléatoire d'une somme de variables aléatoires quand cette somme tend vers l'infini.

Avec le recul, l'intérêt de Turing pour ce théorème nous renseigne sur le type de mathématique que Turing affectionne déjà : le théorème permet de considérer le caractère aléatoire d'un événement *physique* comme une apparence, dès lors qu'il est possible de *réitérer* suffisamment longtemps le nombre des événements. C'est donc la *réitération* qui permet, par convergence, d'approcher de façon déterministe la singularité d'un événement (indexé par un nombre réel) dont l'apparition relève de causes qui, physiquement, ne sont pas pleinement déterminables. Le schème de la réitération apparaît donc comme une *clé déterministe* pour réduire l'aléatoire tel qu'il se manifeste dans le monde physique.

En avril 1934, Turing passe ses examens de fin de troisième année avec « Distinction », l'appréciation la plus haute que l'on puisse obtenir, qui n'est accordée cette année-là qu'à huit candidats. Il rend sa « dissertation » concernant le théorème de la limite centrale quelques mois plus tard, en décembre 1934. Bien qu'elle ne soit pas la première, la démonstration présentée par Turing laisse espérer d'autres résultats originaux dans l'avenir ; l'assemblée des professeurs de *King's College* ne s'y trompe pas, puisque cette démonstration vaut à Turing, au printemps 1935, sa position de *Fellow*, c'est-à-dire d'enseignant-chercheur, pour une durée de trois ans. Turing n'a que 22 ans.

Le début de la recherche

À la même période, Turing passe la dernière partie de son cursus de *Tripos* : il choisit de suivre le cours de Max Newman sur le fondement de la théorie des ensembles. Ce cours l'introduit à la logique mathématique et à la problématique mise en place par Hilbert : formaliser les mathématiques en vue d'établir un fondement solide, au sein duquel on pourra utiliser sans restriction tous les outils patiemment mis en place par les mathématiciens, en particulier ceux qui font appel à l'infini actuel (un infini que l'on pourrait parcourir intégralement).

Le cours de Newman ne décrit pas seulement le projet de Hilbert ; il aborde aussi les limitations inhérentes au projet en question, en particulier la première d'entre elles, l'incomplétude de tout système formel telle qu'elle a été établie par Gödel en 1931 : dans tout système formel, il existe au moins une proposition vraie qui n'est pas démontrable. En avril 1936, Alan Turing présente à un Newman médusé le manuscrit tapé à la machine de l'article qui le rendra célèbre, *On computable numbers, with an application to the* Entscheidungsproblem *(Sur les nombres calculables avec une application au problème de la décision)*. Le résultat qu'il y présente s'apprécie par rapport à celui de Gödel : il en corrobore la généralité tout en donnant une définition générale de la notion de calcul, comme nous allons le voir dans le chapitre suivant. ■

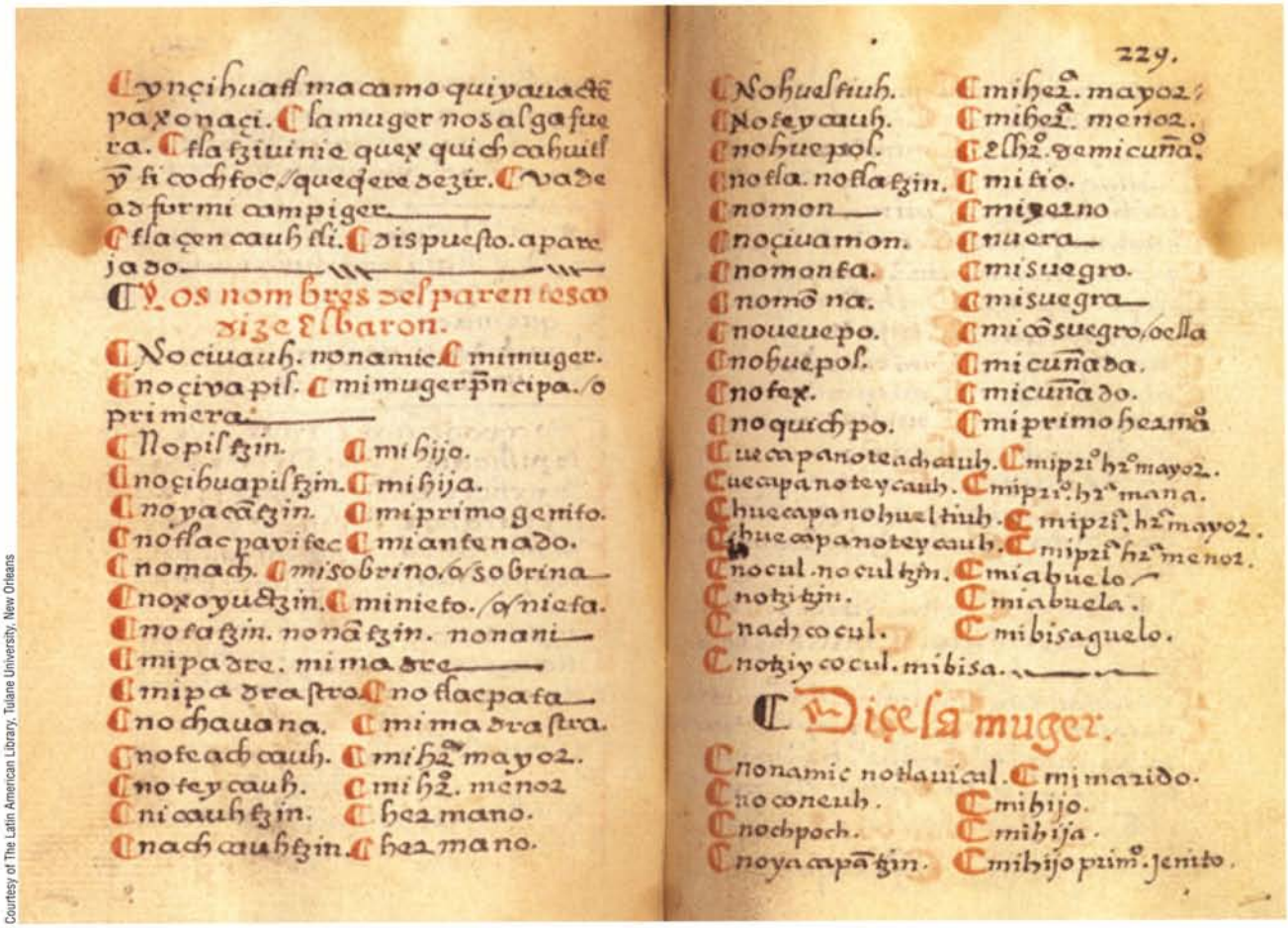
En 1936, en publiant son article sur le problème de la décision, *On Computable Numbers, with an Application to the Entscheidungsproblem*, Turing porte un nouveau coup à l'ambitieux programme lancé par le mathématicien allemand David Hilbert : il ôte tout espoir de construire des fondements intégralement formalistes pour les mathématiques. Ce faisant, néanmoins, il ouvrira un nouvel horizon très prometteur : l'informatique. Comment ce jeune mathématicien, novice dans ce domaine de recherche, est-il venu à bout d'un problème qui résistait aux plus grands scientifiques de l'époque ?

L'étymologie du mot « calcul » fournit une piste : le terme « calcul », qui vient du latin *calculus*, désigne un petit caillou qui servait soit à voter lors des assemblées et des procès, soit à compter en dénombrant les unités. Si le rapport entre « calcul » et « compter » nous apparaît immédiatement, celui entre « calcul » et « vote », c'est-à-dire décision, est moins évident. Pourtant, l'expression dérivée du latin, « marquer d'une pierre blanche », qui désigne un événement heureux, en témoigne encore : dans un vote lors d'un procès à Rome, le caillou blanc signifiait « acquitter » et le caillou noir, « condamner ». La dualité de sens du mot *calculus* se retrouve dans la notion de calcul élaborée par Turing : ce qui permet de prendre des décisions (*Entscheidungen*) est aussi ce qui sert au dénombrement (*Computable numbers*).

Quel rapport y a-t-il entre ces deux activités ? Voilà ce qu'il nous faut comprendre, car l'étymologie n'explique pas la parenté entre les deux sens de

Aux origines du calcul

L'article de Turing de 1936 sur le problème de la décision repose sur la notion de « calcul » et les différents sens que l'on peut lui donner : le dénombrement, certes, mais aussi la décision. Quel est le lien entre ces deux interprétations ?



Courtesy of The Latin American Library, Tulane University, New Orleans

Page ci-contre, un extrait de la Gramática y vocabulario de la lengua mexicana (Grammaire et vocabulaire de la langue mexicaine) *de Fray Andrés de Olmos (1547), l'une des nombreuses grammaires conçues par les Européens, au lendemain des grandes découvertes, pour consigner les règles d'usage des nouvelles langues rencontrées.*

calculus, l'un relevant des *signes* fastes ou néfastes et l'autre du domaine du *nombre*. Pour saisir ce rapport à l'époque moderne, remontons, dans l'histoire de la rationalité, au vaste mouvement qui précéda l'apparition des approches modernes de la notion de calcul telles qu'elles ont été élaborées par les mathématiciens de la première moitié du XX^e^ siècle et, parmi celles-ci, celle de Turing.

La mécanisation du langage

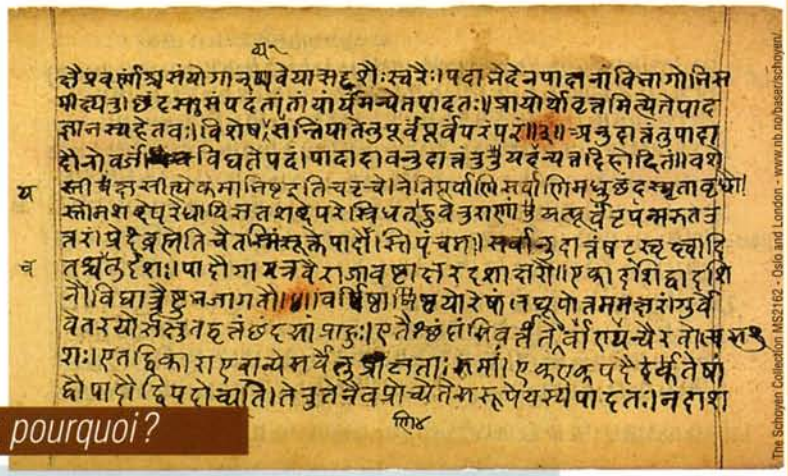
Pendant l'âge dit *classique*, qui couvrit, au sens large, la période du XVI^e^ au XVIII^e^ siècle, la façon dont on explique les phénomènes se transforma profondément, que ceux-ci soient d'ordre culturel (phénomènes linguistiques) ou naturel (phénomènes physiques) : furent considérés intelligibles les phénomènes que l'on pouvait reproduire au moyen d'instruments *mécaniques*. La raison *technique* servait dorénavant de moteur à l'explication scientifique. Retraçons les trois principales étapes de cette révolution capitale de la pensée.

La première étape est la *grammatisation* des langues du monde : dès la Renaissance se développent, à la faveur des découvertes de nouveaux continents, des instruments techniques d'intelligibilité des langues, consistant à dégager leurs grammaires et à noter leurs lexiques. Ce mouvement, qui s'accentue à l'âge classique, éclaire le premier sens étymologique du mot *calculus*, qui a trait à l'intelligibilité des signes institués, c'est-à-dire au premier chef à celle des langues parlées.

Contrairement à ce que l'on pourrait croire au premier abord, la grammaire d'une langue, c'est-à-dire l'ensemble de ses règles de construction stabilisées à un instant donné et pour un groupe social donné, n'est pas connue de ses locuteurs, bien que ceux-ci la respectent et sachent immédiatement distinguer un usage correct d'un autre jugé déviant. Décrire la grammaire d'une langue est donc un acte *théorique* qui consiste à s'extraire du simple flux de la parole et à envisager la langue comme un *objet extérieur à l'acte de parole et soumis à des principes de construction*.

Certes, l'élaboration de grammaires rendant possible la constitution des langues en objets théoriques est aussi vieille que l'apparition de l'écriture dans les civilisations qui ont développé cet outil (Mésopotamie, Grèce, Égypte, Inde, Monde arabe, Chine, Mexique), mais, à partir de la Renaissance, l'aspect *systématique* de la grammatisation est le signe d'un nouveau rapport au langage. Des centaines, puis des milliers de langues ont ainsi été grammatisées sur le modèle de la grammaire gréco-latine (les langues d'Europe à peu près en même temps que les langues amérindiennes,

Le Dasatayipratisakhya *de Saunakacarya, grammaire en sanskrit du langage utilisé dans le Rigveda, plus ancien recueil d'hymnes de la religion védique. Cette religion fut imposée en Inde par les Aryens qui envahirent la région au II^e^ millénaire avant notre ère. La grammaire présentée ici fut rédigée en 1665, en Inde. Curieusement, malgré l'impact du modèle grammatical sanskrit sur de nombreuses cultures, les Hindous n'ont pas grammatisé d'autres langues.*



Grammatiser, pourquoi ?

Pourquoi l'important mouvement de grammatisation s'est-il généralisé en Europe ? Pour des raisons contingentes à la situation linguistique européenne, avance le linguiste Sylvain Auroux : le latin avait été grammatisé sur le modèle du grec dès l'Antiquité et servait essentiellement d'outil de maîtrise de l'écriture pour des individus dont il était la langue maternelle. Lorsque le latin a cessé d'être la langue maternelle tout en restant la langue intellectuelle et religieuse de l'Europe, la grammaire latine a changé de statut et est devenue un modèle d'intelligibilité transférable à n'importe quelle langue parlée, européenne ou non. C'est donc la diversité linguistique, cependant liée à un modèle unitaire d'intelligibilité, qui a promu la grammaire latine au rang de technique transférable à d'autres langues.
Par ailleurs, on constate, sans pouvoir l'expliquer, que le sanskrit, qui a beaucoup servi de modèle grammatical, y compris à des langues éloignées (langues tibéto-birmanes, langues dravidiennes), n'a pas eu le même effet. De même, malgré une solide tradition grammaticale propre, les locuteurs de l'arabe, qui ont sans doute croisé le plus d'aires linguistiques différentes lors de l'expansion mondiale de l'islam, n'ont cherché ni à grammatiser les langues rencontrées selon leur canon grammatical ni à constituer un réseau techno-linguistique tel celui que développera l'Europe de l'âge classique.

Le grand horloger de Voltaire

« Il est vrai, j'ai raillé Saint-Médard et la bulle ;
Mais j'ai sur la nature encor quelque scrupule.
L'univers m'embarrasse, et je ne puis songer
Que cette horloge existe et n'ait point d'horloger. »

Les Cabales, *Voltaire, 1772*

même si l'adaptation du format de la grammaire latine fut, pour des raisons évidentes, plus facile dans le cas des langues européennes). « À la fin du XVI^e^ siècle, on peut estimer que le patrimoine espagnol en Amérique latine porte sur 33 langues différentes ; à la fin du XVII^e^ sur 86 langues, à la fin du XVIII^e^, sur 158 langues », explique le linguiste Sylvain Auroux dans *La révolution technologique de la grammatisation* (1994).

La grammaire devient dès lors un instrument *technique* permettant de décrire la langue comme un *mécanisme* dont la structure est régie par des règles : envisager *toutes les langues humaines* sous l'angle de leur grammaire, c'est donc faire l'hypothèse consciente, ou non, que le langage résulte d'un mécanisme de construction dont chaque langue n'est qu'une matérialisation particulière.

Ainsi, historiquement, la technique grammaticale gréco-latine a servi de modèle à un *transfert technologique massif* puisque les langues du monde ont été grammaticalement outillées à partir de ce modèle ; elle a aussi alimenté une *croyance très particulière*, celle de l'identité d'un mécanisme grammatical au-delà de la diversité des langues, projeté en amont (dans une langue originelle) ou en aval (dans une future langue universelle).

Cette attitude technicienne propre à la rationalité de l'âge classique touche aussi le lexique. Le rapport au lexique s'instrumentalise lui aussi à la même époque quand on commence à doter les langues de dictionnaires *unilingues*. L'invention nous est si familière aujourd'hui que nous n'y prêtons plus attention. Pourtant, un dictionnaire est la constitution *mécanique* d'un lexique d'une langue (y compris la langue maternelle) en tant qu'objet théorique d'étude : il favorise l'accumulation d'un savoir sur la langue qu'aucun individu n'a jamais possédé sans l'aide d'outils appropriés.

Ainsi, dorénavant, l'apprentissage et l'usage de la langue dépendent de la médiation d'un outillage technique externe, socialement construit et diffusé par l'imprimerie. Le rapport aux langues de chaque locuteur est transformé : l'idée même de langue « naturelle » a évolué, s'est « culturalisée » sous l'effet de l'outil grammatical.

Dans cette évolution majeure, la mécanisation, tout d'abord liée à l'intelligibilité des signes institués, s'est peu à peu étendue à celle de la nature. La transformation touche d'abord la physique et les mathématiques.

La nature à l'image d'une horloge

C'est dans le contexte de la mécanisation de la nature que l'on retrouve le deuxième sens étymologique du mot *calculus*, qui a trait au dénombrement. Jusqu'alors, mathématique et physique avaient été dominées par un paradigme, celui de l'axiomatique de la géométrie élaborée par Euclide au III^e^ siècle avant notre ère : dans ses *Éléments*, Euclide avait déduit toute la géométrie à partir d'un minimum de postulats et axiomes de base. La physique se devait de suivre bon an mal an l'exemple de l'axiomatique de la géométrie et, en conséquence, adopter un style *déductif* à partir de principes abstraits intuitivement reçus, même si cette déduction rendait peu compte de la diversité des phénomènes naturels et de leurs transformations.

À partir du XVII^e^ siècle, une nouvelle attitude se dégage : on considère qu'un phénomène est expliqué quand on sait le reproduire au moyen d'instruments ou d'expériences relevant de la mécanique. *L'horloge* devient un modèle explicatif fondamental des phénomènes naturels, qu'ils soient cosmologiques, physiques ou biologiques. Dieu lui-même ne se verra-t-il pas attribuer plus tard par Voltaire le nom de « Grand Horloger » ?

Le frontispice de la première édition du* Dictionnaire de l'Académie françoise *(1694). À partir du XVI^e^ siècle, avec l'engouement pour les dictionnaires bilingues apparut celui pour les dictionnaires unilingues : l'Académie française fut fondée par le cardinal de Richelieu en 1635 avec pour objectif essentiel de créer un dictionnaire du français. La mécanisation du langage se poursuit…

Et la géométrie ?

Lorsque, au XVII^e siècle, la mécanique prend le pas sur l'axiomatique de la géométrie euclidienne comme fondement des sciences, la géométrie n'échappe pas à cette mécanisation. Isaac Newton, le plus grand physicien du XVII^e siècle, décrit ainsi cette discipline dans ses Principes de philosophie naturelle *(1686) : elle doit être conçue comme science des rapports entre les figures, ce qui suppose que soient construites au préalable un certain nombre de figures pour que puissent s'exercer entre elles des rapports de mesure et, plus généralement, des rapports de détermination réciproque. Mais la construction des figures relève d'une pratique technique antérieure à la géométrie et sur laquelle cette dernière se fonde : « Les constructions de lignes droites et de cercles sont des problèmes, mais non des problèmes de géométrie. La solution de ces problèmes relève de la mécanique et, ceux-ci résolus, la géométrie en montre l'usage [...]. Ainsi la géométrie est-elle fondée sur la mécanique pratique. » Cette mécanique pratique est le terrain de la physique par excellence.*

Isaac Newton (1642-1727) sur une gravure du XIX^e siècle.

Dès lors, faire de la physique une science véritable consiste à mettre au jour la nécessité des rapports entre les phénomènes sous la forme du *déterminisme*, c'est-à-dire en anticipant sur tout le développement du processus avant même qu'il ait lieu. L'image la plus parlante de ce rapport de cause à effet est celle du *mécanisme*, régi par des lois d'où tout hasard est exclu. La notion de *fonction* apparaît pour répondre à ce nouveau besoin : une fonction opère une transformation réglée d'une situation – géométrique ou numérique – en une autre.

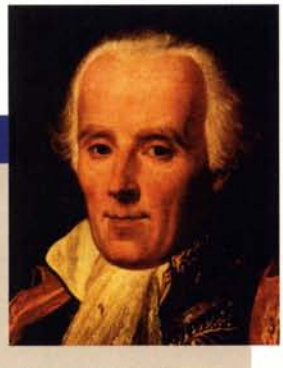
La notion abstraite de fonction fut élaborée dès la fin du Moyen Âge par le philosophe et savant français Nicolas Oresme (1320-1382), mais son usage ne se généralisa qu'avec Descartes et l'avènement de la géométrie analytique, au XVII^e siècle : une fonction opère une transformation, par exemple d'un nombre en un autre nombre, d'une figure géométrique en une autre ou d'une figure géométrique en un nombre selon une règle dont l'application donne un résultat unique, s'il est défini. On conçoit l'importance de cet instrument mathématique pour l'étude des mouvements physiques : pour la première fois, il devient possible d'établir une relation précise entre les processus *causaux* en physique et le calcul *numérique* en mathématiques. Cette nouvelle façon de procéder sera promise à un grand avenir dans les sciences de la nature, en particulier en physique.

Cette conjonction entre processus causaux et calcul constitue le cœur de la notion de déterminisme en physique : toute détermination causale devient, en droit, *mathématiquement calculable* (d'où la conception quantitative des processus physiques). Au début du XIX^e siècle, le mathématicien et physicien Pierre Simon de Laplace (1749-1827) systématisera le cadre théorique du déterminisme en utilisant l'image d'un « démon omniscient », dit « démon de Laplace » : si cette créature connaissait la position et la vitesse de toutes les particules de l'Univers à un instant donné, elle serait capable d'en déduire tous ses états futurs comme tous ses états passés. *Déterminisme, mécanisme et calculabilité* entretiennent ainsi des rapports étroits dans les sciences de la nature de l'âge classique.

La crise de la géométrie euclidienne

L'attitude axiomatique elle-même, essentiellement liée à la géométrie euclidienne, semble avoir échappé jusqu'alors au mouvement de mécanisation qui a touché les sciences humaines comme les sciences de la nature. Deux événements précipitent son renouveau, le premier lié à l'évolution du statut philosophique de la géométrie et le second dû à l'apparition d'une nouvelle arithmétique de l'infini.

Depuis l'origine de la géométrie euclidienne, nombre de savants avaient tenté de clarifier le cinquième postulat d'Euclide, dit « postulat des parallèles », selon lequel par un point hors d'une droite, il ne passe qu'une parallèle à celle-ci. Le postulat des parallèles n'avait cessé de poser problème au cours de l'histoire des mathématiques pour



Le démon de Laplace

« Nous devons envisager l'état présent de l'Univers comme l'effet de son état antérieur et comme la cause de celui qui va suivre. Une intelligence qui, pour un instant donné, connaîtrait toutes les forces dont la nature est animée et la situation respective des êtres qui la composent, si d'ailleurs elle était assez vaste pour soumettre ces données à l'analyse, embrasserait dans la même formule les mouvements des plus grands corps de l'Univers et ceux du plus léger atome ; rien ne serait incertain pour elle, et l'avenir, comme le passé, serait présent à ses yeux. »

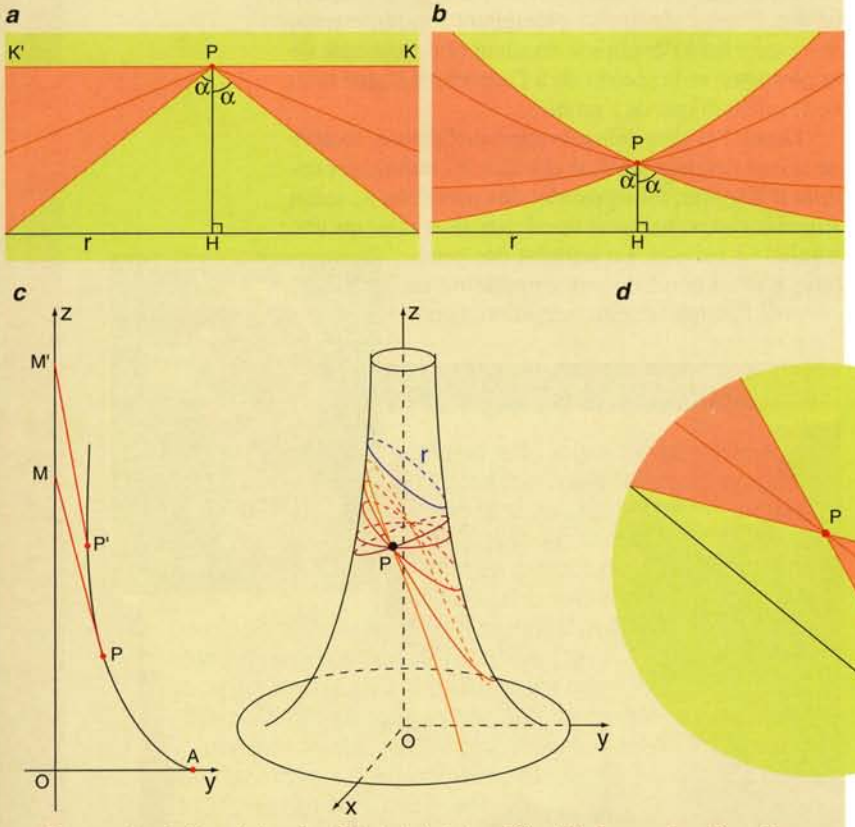
Essai philosophique sur les probabilités,
Pierre Simon de Laplace (1814)

Une géométrie non euclidienne

Au XIXe siècle, le mathématicien russe Nicolaï Lobatchevski (1793-1856), l'un des pionniers de la géométrie non euclidienne, construit une géométrie qui s'affranchit du postulat euclidien des parallèles, la géométrie hyperbolique. Dans cette géométrie, il existe une infinité de droites passant par un même point P *et parallèles à une droite* r *qui ne contient pas ce point. Lobatchevski définit ainsi un angle de parallélisme* α*, angle compris entre 0 et* $\pi/2$ *mesuré à partir de la perpendiculaire* PH *à* r *passant par* P*, tel que toute droite faisant avec* PH *un angle supérieur à* α *est parallèle à* r*, et toute droite faisant avec* PH *un angle inférieur à* α *coupe la droite* r*. La géométrie euclidienne correspond au cas limite où l'angle* α *vaut* $\pi/2$*: dans ce cas, le secteur se résume à une droite* KK'*. Une façon d'appréhender cette géométrie est de considérer un morceau de plan tel celui représenté sur le schéma* a *: l'angle de parallélisme* α *est l'angle limite pour lequel on peut dessiner une droite qui, dans ce morceau de plan, coupe la droite* r*. En d'autres termes, toutes les droites incluses dans le secteur orange sont des parallèles à* r*.*

Une autre façon est d'imaginer que la surface sur laquelle est décrite la géométrie de Lobatchevski est telle que ses droites (ou géodésiques) sont des courbes, comme celles représentées sur le schéma b*. Le mathématicien italien Beltrami montrera que la pseudosphère est un bon support pour la géométrie hyperbolique.*

La pseudosphère, construite en 1839 par Ferdinand Minding, est une surface engendrée par la rotation d'une courbe particulière, la tractrice, autour de son asymptote (voir le schéma *c*; l'asymptote est ici l'axe *z*)*. La tractrice est définie de la façon suivante: chaque tangente, tracée à partir d'un point* P *quelconque de la courbe, rencontre l'axe* z *en un point* M *tel que la distance* MP *est égale à une constante déterminée. Beltrami montra que les géodésiques de la pseudosphère sont équivalentes aux cordes d'un cercle du plan euclidien, dit cercle limite* (voir le schéma *d*)*. Par un point* P *extérieur à une droite* r*, on peut tracer une infinité de parallèles à cette droite : toutes celles qui sont dans le secteur orange.*



Deux représentations de la géométrie de Lobatchevski (a et b). La pseudosphère et la construction de sa tractrice (c). Le modèle du cercle limite de Beltrami (d).

des raisons philosophiques tenant à la conception que l'on se faisait de la vérité géométrique : jusqu'au XIXe siècle, la vérité de toute proposition géométrique provenait soit de *l'évidence* que l'on pouvait en avoir, soit de la *démonstration* à partir d'autres propositions considérées comme évidentes. Cette double source de la vérité permettait d'étendre la légalité de la géométrie non seulement à l'espace abstrait des mathématiques, mais aussi à l'espace perçu par les sens, l'espace de la physique : l'évidence des axiomes et postulats de la géométrie euclidienne tissait un lien de parenté entre les formes perçues dans l'espace concret et les formes géométriques abstraites imaginées dans l'espace abstrait. Or le postulat des parallèles – dont la vérité n'était pas remise en question puisqu'elle faisait partie des réquisits minimaux de la géométrie – ne tombait ni dans la catégorie de l'évidence ni dans celle de la démonstration.

Le problème philosophique posé par sa vérité était le suivant : dans les deux cas de vérité classiquement admis, le projet d'une description mathématique de la réalité spatiale était *garanti* par la *vérité* des propositions géométriques, c'est-à-dire par leur caractère évident ou démontrable ; le fait que le cinquième postulat résistât à la justification par l'évidence ou la démonstration *rendait donc précaire tout le projet rationnel d'une description mathématique de la réalité physique*, le lien entre mathématique et physique se trouvant compromis. Justifier la vérité du cinquième postulat en termes d'évidence ou de démonstration *devenait ainsi une tâche préalable à tout projet philosophique de justification de la nature physique en termes mathématiques*. On comprend mieux pourquoi les savants se penchèrent sur le problème avec autant d'obstination.

Ce n'est qu'au tout début du XIXe siècle qu'apparaît l'idée qu'il faut non pas faire entrer coûte que coûte le postulat des parallèles dans les deux catégories de vérité classiquement reçues, mais s'interroger sur la *sorte de vérité* que recèle le postulat lui-même, puisqu'il semble relever d'un *troisième* genre de propositions vraies. Cette idée remet en question le lien, admis jusqu'alors, entre la géométrie comme science de l'espace et la réalité spatiale telle qu'elle est perçue : la vérité des propositions géométriques peut ne pas avoir d'incidence sur la vérité des propositions décrivant l'espace physique.

Le mathématicien allemand Carl Friedrich Gauss (1777-1855) est le premier à émettre des doutes sur le caractère naturel du lien unissant l'espace perçu et la géométrie euclidienne. Il en conclut que la géométrie euclidienne n'a pas le même caractère de vérité que d'autres parties des mathématiques telles que l'arithmétique, où la justification de la vérité de postulats ne se pose pas en termes aussi problématiques. Il déclare dans une lettre à son collègue Heinrich Olbers : « Je deviens de plus en plus convaincu que la nécessité de notre géométrie ne peut pas être prouvée, du moins pas pour des

raisons humaines [...]. Peut-être que dans une autre vie, nous serons capables de mieux discerner la nature de l'espace, ce qui est présentement inaccessible. D'ici là, nous devons placer la géométrie non pas dans la même classe que l'arithmétique, laquelle est purement *a priori*, mais dans la même classe que la mécanique. » Ainsi la géométrie est-elle identifiée à la mécanique non seulement du point de vue externe, celui des physiciens utilisant les concepts mathématiques à leur usage, mais du point de vue *interne* aux mathématiques elles-mêmes.

Le désordre géométrique

À la suite de Gauss, plusieurs mathématiciens construisent des géométries qui ne s'appuient plus sur le cinquième postulat d'Euclide. Il devient possible de décrire l'espace physique à l'aide de représentations géométriques différentes de la représentation euclidienne : la géométrie reste cohérente si l'on change le contenu du cinquième postulat en faisant l'hypothèse qu'il existe une infinité de parallèles à une droite en un point donné extérieur à celle-ci, ou qu'au contraire il n'en existe aucune. Ces géométries sont tout aussi recevables que la géométrie euclidienne à partir du moment où le lien entre espace géométrique et espace perçu est rompu. Trois conséquences majeures en découlent.

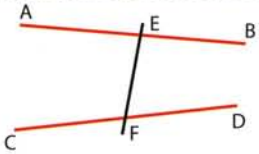
Du point de vue de la physique se pose la question de savoir quelles géométries appliquer *à bon escient* à quelles réalités naturelles : immense programme de recherche qui occupera les physiciens jusqu'à la découverte de la relativité par Einstein, et au-delà. Les modèles géométriques des réalités physiques, devenus instruments opératoires de l'intelligibilité de la nature, connaîtront un tel essor qu'il est aujourd'hui possible, avec le recul, de suivre le cours de l'histoire de la physique moderne comme celle des étapes de sa *géométrisation*.

Du point de vue mathématique, la voie ouverte par Gauss pose la question de la nature de *l'espace* et de son rapport aux différents modèles géométriques possibles. Quels rapports entretiennent ces géométries puisqu'elles sont contradictoires ? Dans quelle mesure peut-on représenter des espaces non euclidiens dans l'espace euclidien ? Y a-t-il moyen d'élaborer une axiomatique générale de la géométrie conçue comme science de l'espace, qui englobe les axiomatiques euclidiennes et non euclidiennes comme cas particuliers ? Là encore, le chantier est immense et l'étude des rapports abstraits entre axiomatiques occupera la recherche mathématique jusqu'à l'époque contemporaine.

Le mathématicien Carl Friedrich Gauss (1777-1855, ci-dessus) renouvela la vision de la géométrie en considérant une surface non pas comme la frontière d'un solide, mais comme un solide flexible et inextensible de dimension nulle. Le plan euclidien, soumis au postulat des parallèles, n'est alors plus qu'un cas parmi d'autres.

Le postulat des parallèles

« Si une droite tombant sur deux droites fait les angles intérieurs et du même côté plus petits que deux droits, les deux droites, indéfiniment prolongées, se rencontrent du côté où sont les angles plus petits que deux droits. »



Du point de vue philosophique enfin, la nature de la vérité géométrique dans le cadre axiomatique exige d'être à nouveau interrogée : du fait de la présence possible de propositions semblables au cinquième postulat, certaines propositions résistent à toute démonstration. En d'autres termes, la démonstration ne suffit plus à assurer le transfert d'évidence d'une proposition à une autre. Par conséquent, les savants risquent d'admettre, sans le vouloir, des propositions contradictoires. La confusion axiomatique qui résulte de cette constatation confine même, pour certains, à un véritable « délire », selon l'expression employée par le mathématicien et logicien allemand Gottlob Frege (1848-1925) dans *Les fondements de l'arithmétique*, car aucun principe ne permet plus de juger du caractère contradictoire de deux propositions et, plus globalement, de la vérité des propositions dans un cadre axiomatique.

Confrontés à ce chaos théorique, les mathématiciens orientent leurs réflexions dans la direction opposée à celle choisie par les physiciens : ces derniers, en prenant acte de la diversité des modèles géométriques de l'espace, jouent de cette pluralisation pour multiplier les modèles de la réalité physique, favorisant l'exactitude descriptive plutôt que le maintien de la cohérence unitaire des principes axiomatiques. Les mathématiciens, en revanche, privilégient la quête des *conditions* a priori *de la vérité axiomatique*.

Nous allons voir qu'ils rechercheront ces conditions dans *l'arithmétique* et son ordre calculatoire. Mais le prix à payer sera lourd : la crise géométrique, qui a introduit une rupture entre l'intuition naturelle de l'espace et ses modèles géométriques, ne sera surmontée qu'au prix d'un *divorce à l'intérieur même des sciences exactes* entre les physiciens, partisans de la géométrisation multiple du monde, et les mathématiciens, tentant de trouver dans l'arithmétique, c'est-à-dire *à l'opposé* de tout rapport à l'espace, une zone de sécurité pouvant servir de fondement à l'édifice mathématique. C'est dans cette *mise en ordre arithmétique* que se développera l'idée moderne de calcul. ■

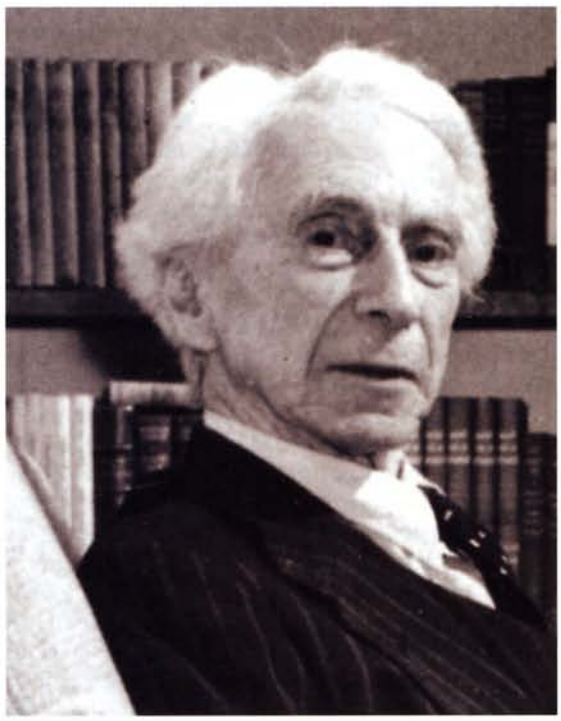
La mécanisation

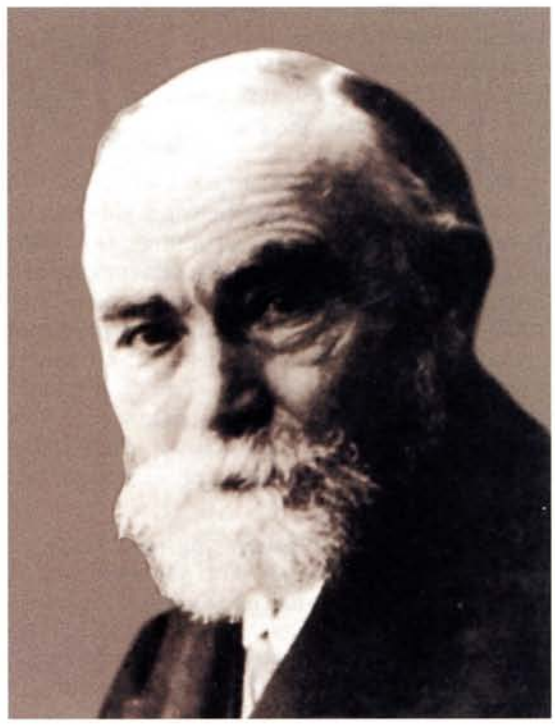
Peut-on construire un système d'axiomes à partir duquel les mathématiques seraient déductibles ? Cette question conduisit les mathématiciens du début du XX^e siècle à préciser la nature et le rôle du nombre.

Lorsque, au XIX^e siècle, une grave crise d'identité secoue la géométrie, les mathématiciens en quête d'un fondement axiomatique de leur science, tournent leurs espoirs vers l'arithmétique *(voir* Aux origines du calcul, *page 50)*. Pourquoi cette discipline ? Aurait-elle été épargnée par le débat sur la vérité axiomatique ? Non. Au XIX^e siècle, l'arithmétique a eu, à l'instar des autres sciences, son lot de bouleversements, qui ont conduit, tout comme en géométrie, à la quête d'une axiomatique dont découlerait toute l'arithmétique. Cependant, cette quête portera, en partie, ses fruits. Une nouvelle théorie révolutionnant l'usage de l'infini en mathématique est à l'origine de cette remise en question : la théorie des ensembles, fondée par le mathématicien Georg Cantor (1845-1918). Cette théorie attaque l'usage naïf de l'intuition en arithmétique, qui a tendance à tenir dans l'infini des raisonnements qui ne valent que pour le fini.

Depuis l'Antiquité grecque, la distinction entre le fini et l'infini en mathématiques avait gravité autour de trois notions : le fini, propriété des collections dont on peut dénombrer les éléments, l'infini potentiel, qui autorise les successions indéfinies telles que la suite des entiers naturels, et l'infini actuel, qui envisage d'un seul bloc les collections infinies, par exemple la collection des nombres entiers ou celle des nombres pairs. Jusqu'au XIX^e siècle, l'infini actuel n'avait pas droit de cité en mathématiques parce qu'on limitait à l'infini potentiel toutes les démonstrations faisant usage de l'infini. On avait démontré par exemple dès l'Antiquité qu'il y avait une infinité de nombres premiers (nombres divisibles seulement par eux-mêmes et par 1), d'une part, en montrant qu'il serait logiquement contradictoire qu'il en fût autrement (*voir l'encadré page 58*) et, d'autre part, en fournissant les moyens de localiser par le calcul les nombres premiers dans la liste ouverte des nombres entiers.

L'usage de l'infini en arithmétique était ainsi encadré par les règles de la logique établies par Aristote au IV^e siècle avant notre ère, dont trois formaient l'armature de tout raisonnement : le principe de noncontradiction, le principe selon lequel le tout est





Die logischen Grundlagen der Mathematik[1]**.**

Von

David Hilbert in Göttingen.

Meine Untersuchungen zur Neubegründung der Mathematik[2]) bezwecken nichts Geringeres, als die allgemeinen Zweifel an der Sicherheit des mathematischen Schließens definitiv aus der Welt zu schaffen. Wie nötig eine solche Untersuchung ist, gewahren wir, wenn wir bedenken, wie wechselnd und unpräzise die diesbezüglichen Anschauungen oft selbst der hervorragendsten Mathematiker waren, oder wenn wir uns erinnern, daß von einigen der namhaftesten Mathematiker der neuesten Zeit die bisher für die sichersten gehaltenen Schlüsse in der Mathematik verworfen werden.

Zur vollständigen Lösung der in Rede stehenden prinzipiellen Schwierigkeiten ist, wie ich glaube, eine Theorie des mathematischen Beweises selbst nötig. Diese Beweistheorie habe ich nunmehr unter der wirksamsten Hilfe und Mitarbeit von Paul Bernays soweit fortgeführt, daß in der Tat durch sie die einwandfreie Begründung der Analysis und Mengenlehre gelingt; ja ich glaube nunmehr so weit zu sein, daß man auch an die großen klassischen Probleme der Mengenlehre von der Art des Kontinuumsproblems und an die nicht minder wichtigen noch offenen Probleme der mathematischen Logik erfolgreich wird herantreten können.

Diese ganze Theorie mit ihren langen und schwierigen Entwickelungen hier darzulegen, ist unmöglich. Es haben sich aber im Laufe der Untersuchung eine Reihe von neuen Einsichten und Zusammenhängen herausgestellt, die auch einzeln für sich und von den übrigen losgelöst Interesse verdienen. Ich möchte eine solche, wie ich glaube, neue Einsicht hier zur Sprache bringen, die außerdem gerade von der Art ist, daß sie den Kern meiner Beweistheorie sehr tief berührt.

[1]) Vortrag, gehalten in der Deutschen Naturforscher-Gesellschaft. September 1922.

[2]) Vgl. meine in Kopenhagen und Hamburg gehaltenen Vorträge, Abhandlungen aus dem mathematischen Seminar der Hamburgischen Universität 1922.

du monde

plus grand que la partie et le principe dit *du tiers-exclu* qui permet, confronté à une alternative, d'opter machinalement pour une branche si l'autre est impraticable (si une proposition *P* n'est pas vraie, alors la négation *non-P* de cette proposition est nécessairement vraie). Ce dernier principe intervient par exemple dans la résolution du problème suivant. Sachant qu'un nombre rationnel est un nombre qui s'exprime sous la forme d'un rapport entre nombres entiers et qu'un nombre irrationnel ne peut pas s'exprimer sous cette forme, on peut se poser la question : existe-t-il deux nombres irrationnels a et b tels que soit rationnel ? Examinons si le nombre $\sqrt{2}^{\sqrt{2}}$ a la propriété d'être ou non rationnel. Par le principe du tiers-exclu, la disjonction suivante est vraie : ($\sqrt{2}^{\sqrt{2}}$ est rationnel) ou ($\sqrt{2}^{\sqrt{2}}$ n'est pas rationnel).

Si $\sqrt{2}^{\sqrt{2}}$ est rationnel, il suffit de choisir $a = b = \sqrt{2}$ (dont on peut prouver qu'il est un nombre irrationnel) pour répondre positivement à la question. Si $\sqrt{2}^{\sqrt{2}}$ est irrationnel, il suffit de choisir $a = \sqrt{2}^{\sqrt{2}}$ (supposé irrationnel) et $b = \sqrt{2}$ (nombre irrationnel) pour obtenir un nombre rationnel : $(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = (\sqrt{2})^2 = 2$, qui est un nombre rationnel.

L'intervention du principe du tiers-exclu dans cet exemple permet de démontrer l'existence de a et b, mais ne permet pas de préciser les nombres ayant la propriété recherchée. Confrontés à cette difficulté, les mathématiciens remirent en question l'usage du principe du tiers-exclu, en particulier le mathématicien néerlandais Luitzen Brouwer (1881-1966) et l'école intuitionniste.

La nouvelle arithmétique de l'infini

Avec la théorie des ensembles, tout change : l'infini actuel apparaît en mathématiques. Depuis longtemps déjà, des mathématiciens audacieux comme Thabit Ibn-Qura, les jésuites de l'école de Coimbre ou Galilée avaient spéculé sur des faits troublants : il était par exemple possible de mettre en correspondance chaque nombre entier avec chaque nombre pair et de conclure que la liste des nombres entiers et celle des nombres pairs avaient un « poids » *identique* (il y avait une correspondance bijective entre elles), alors que l'une des listes avait *deux fois moins* d'éléments que l'autre !

Cette conclusion remettait en question les principes logiques hérités des Anciens : dans ce cas en particulier, la partie (les nombres pairs) semblait équi-

Le mathématicien allemand David Hilbert* (ci-dessus) *et une page du fameux article publié en 1923 où il annonce son vaste programme de construction des fondements des mathématiques,* Die logischen Grundlagen der Mathematik (Les fondements logiques des mathématiques). *Turing contribuera à montrer que ce projet ne peut être mené à bien. Page ci-contre, le philosophe britannique Bertrand Russell (1872-1970,* à gauche) *et le logicien allemand Gottlob Frege (1848-1925,* à droite), *qui tentèrent de construire une logique axiomatique dont ils auraient déduit toutes les mathématiques.

Georg Cantor (1845-1918, ici en compagnie de sa femme), fondateur de la théorie des ensembles, introduisit le trouble dans l'arithmétique en osant utiliser la notion d'infini actuel dans ses calculs.

valoir au tout (les nombres entiers). L'évidence était violée et l'esprit se trouvait confronté à une contradiction, puisqu'une même réalité était à la fois identique *et* non-identique à une autre. De façon générale, dans tous les raisonnements faisant usage de l'infini actuel, les mathématiciens n'étaient plus certains qu'ils pouvaient utiliser en l'état le principe du « tiers exclu », car rien ne départageait plus deux solutions mutuellement exclusives. La logique classique semblait adaptée aux raisonnements portant sur des collections finies ou potentiellement infinies, mais, dès que l'infini actuel apparaissait, les principes de logique, pourtant réputés universels, perdaient toute validité.

L'infinité des nombres premiers

Dès l'Antiquité, les savants ont montré – par l'absurde – qu'il existait une infinité de nombres premiers. Voici la trame du raisonnement d'Euclide. Supposons que les nombres premiers sont en un nombre fini n : p_1, p_2, ... p_n et considérons le nombre $p = p_1 \times p_2 \times ... \times p_n + 1$. Deux cas se présentent, selon que p est premier ou non. Si p est un nombre premier, il existe alors un nombre premier (p) supérieur à chacun des n nombres premiers p_1, p_2, ... p_n. Nous obtenons ainsi $n + 1$ nombres premiers, ce qui contredit l'hypothèse de départ. Si le nombre p n'est pas premier, alors il est divisible par un nombre premier q. Cependant, aucun des nombres p_1, p_2, ... p_n ne peut être son diviseur, car, par définition, la division de p par chacun d'eux donne un reste égal à 1. Ainsi, le nombre premier q qui divise p est différent de p_1, p_2, ..., et p_n, ce qui contredit à nouveau l'hypothèse selon laquelle il n'existe que n nombres premiers. Cette hypothèse n'étant vérifiée dans aucun cas, il existe donc une infinité de nombres premiers.

Au XIX^e^ siècle, Georg Cantor et, à sa suite, plusieurs mathématiciens tels que Richard Dedekind, proposent de définir la notion même d'infini actuel par l'équivalence du tout et de la partie. Ils s'engagent alors dans une exploration de l'arithmétique des collections actuellement infinies, dites « transfinies », sans recourir aux principes classiques de la logique et en s'en tenant à la manipulation de règles formelles. Des écoles mathématiques différentes se constituent, divergeant quant à l'usage de la notion d'infini. Certaines prohibent l'usage de l'infini actuel tandis que d'autres l'encouragent : l'arithmétique devient aussi « chaotique » que la géométrie. D'où la volonté d'y mettre bon ordre en procédant à une axiomatisation de l'arithmétique, ce qu'entreprennent deux mathématiciens italiens, Alessandro Padoa (1868-1937) et Giuseppe Peano (1858-1932). Les deux hommes développent des outils formels, en particulier la description logique de l'induction de la suite des entiers, qui rendront cette axiomatisation possible *(voir l'encadré page ci-contre).*

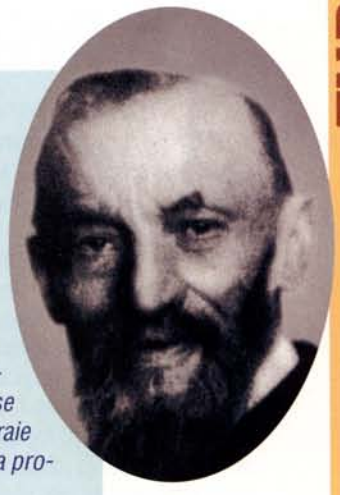
La grammatisation de l'axiomatique

Confrontés à cette situation de crise, tous les mathématiciens reconnaissent cependant dans le fini et dans l'infini potentiel de la suite des entiers une « zone de sécurité » au-delà de laquelle la pratique des mathématiques est laissée à la discrétion de chacun. Mais il faut s'entendre sur cette « zone de sécurité » où réside la certitude de l'arithmétique, dernier bastion des fondements des mathématiques depuis que la géométrie a déclaré forfait.

Un mathématicien joue un rôle particulièrement important dans l'élaboration du diagnostic et du remède à apporter à cette crise géométrique et arithmétique : David Hilbert. Le mathématicien allemand réduit les théories les moins sûres (géométriques) aux théories réputées plus fiables (arithmétiques). Il montre notamment que la non-contradiction de la géométrie cartésienne se fonde sur la non-contradiction des nombres réels : si l'arithmétique des nombres réels est non-contradictoire, alors la géométrie cartésienne l'est aussi. Ainsi, le problème de la non-contradiction d'une axiomatique est à la fois *déplacé* à une autre axiomatique et *réduit* à une axiomatique plus fondamentale, celle de l'arithmétique des nombres réels et des entiers.

Ultérieurement, Hilbert montrera que ce déplacement, qui permet d'obtenir des preuves de non-contradiction *relative* d'une axiomatique par rapport à une autre, se réduit à la preuve de non-contradiction *absolue* de l'arithmétique des entiers : si on démontre la non-contradiction de l'arithmétique des entiers – telle qu'elle a été axiomatisée par Peano –, on en déduit *la non-contradiction de toutes les autres axiomatiques*. Reste à montrer cette non-contradiction : or le problème semble d'une difficulté bien plus grande et, à vrai dire, *insoluble* dans la mesure où le nombre, concept de base de toute arithmé-

L'arithmétique de Peano



En 1889, le mathématicien italien Giuseppe Peano (1858-1932) répertoria les caractéristiques structurelles des nombres naturels dans une liste d'axiomes énoncés dans la symbolique logique. Cette dernière est un langage du premier ordre (c'est-à-dire un langage dans lequel il n'y a que des prédicats sur des objets du langage, et pas de propositions sur les propositions), incluant l'identité. L'identité (dont le symbole est «=») est définie par deux propriétés :

- $a = a ; a = b \rightarrow b = a ; (a = b \text{ et } b = c) \rightarrow a = c$ (1)
- $a_1 = a_2 \rightarrow \varphi(a_1) = \varphi(a_2)$, (2)

où → représente l'implication. Cela signifie que : 1) l'égalité est une relation réflexive, symétrique et transitive et 2) quand deux objets sont identiques, lorsque l'un des objets possède une propriété φ, le second objet la possède également.

Le concept central de l'arithmétique de Peano est celui de successeur : tout nombre naturel x a un successeur. Celui-ci ne peut être écrit x + 1, car l'addition n'est pas encore définie. Peano note donc s(x) («successeur» de x) le nombre qui suit x et précise que la fonction s est définie pour tout nombre naturel x. Il a ainsi formalisé une propriété importante des nombres naturels («on peut toujours compter un de plus») et établi «tacitement» qu'il existe un nombre infini de nombres naturels.

Les constantes du langage de l'arithmétique de Peano sont les suivantes : 0 (le nombre zéro), s (la fonction successeur), «+» et «×», les opérations d'addition et de multiplication. La signification de ces constantes est définie par les axiomes suivants :

- $\forall x (\neg s(x) = 0)$

(0 n'est pas le successeur d'un nombre naturel) où ∀ représente l'expression « quelque soit » et ¬ la négation.

- $\forall x \forall y (s(x) = s(y) \rightarrow x = y)$

(des nombres distincts ont des successeurs différents).

- $\forall \alpha (\alpha(0) \wedge \forall x (\alpha(x) \rightarrow \alpha(s(x))) \rightarrow \forall x \alpha(x))$

où ∧ représente la conjonction « et ». Ceci est le principe d'induction complète : si une propriété α est vraie pour le zéro et si la phrase «Si α est vraie pour un nombre x, α est aussi vraie pour son successeur s(x)» est exacte, alors la propriété α est vraie pour tout entier naturel.

- $\forall x \forall y (x + 0 = x) \wedge x + s(y) = s(x + y)$
- $\forall x \forall y (x \times 0 = 0) \wedge x \times s(y) = x \times y + x$

Ces deux axiomes définissent par induction l'addition et la multiplication.

L'axiomatique ainsi construite par Peano permettait donc d'induire toute la suite des entiers naturels. Restait à savoir si elle pouvait servir de fondement à l'arithmétique, c'est-à-dire si toute l'arthmétique était déductible de cette axiomatique. Peano a fondé une véritable école, dont l'un des élèves les plus fameux, Alessandro Padoa, inventa une méthode permettant de déterminer si un axiome est indépendant des autres. Il présenta son résultat au congrès de Paris d'août 1900, ce même colloque où Hilbert dressa, dans un exposé intitulé L'avenir des mathématiques, *une liste de 23 problèmes ouverts capitaux pour les mathématiques.*

tique, ne se réduit à rien d'autre. Les tentatives de réduction de la notion mathématique de nombre à la notion logique d'ensemble expérimentées par Frege et Russell paraissent en effet inadéquates à Hilbert, car cette réduction extérieure à l'arithmétique produit des paradoxes *(voir l'encadré page 60)*.

Il s'agit donc de prouver la non-contradiction de l'arithmétique *sans sortir de l'arithmétique*, comme les mathématiciens l'ont fait jusqu'alors en recherchant des preuves intuitives de non-contradiction ou par le biais de la notion d'ensemble. Pour y parvenir, Hilbert transforme la notion même d'axiomatique en la scindant en deux, l'une continuant à jouer le rôle qu'elle a toujours eu et l'autre servant de grammaire à la première.

La première sorte d'axiomatique est une axiomatique à *contenu* – celle qui s'est pratiquée chez Euclide pour la géométrie ou chez Peano pour l'arithmétique – et la deuxième une axiomatique *formelle*. L'axiomatique à contenu comporte deux sortes de propositions : des propositions *finitistes*, c'est-à-dire relevant du fini ou de l'infini potentiel et donc toujours vérifiables par des procédures qui s'effectuent en un temps fini, et des propositions *idéales* sans aucun contenu, qui ne sont pas vérifiables par ce moyen. Cette dernière catégorie comprend les propositions portant sur le transfini, dans lesquelles les symboles logiques « Il existe » et « Pour tout » portent sur les individus d'un domaine infini actuel. Hilbert espère réduire ces

Le mathématicien néerlandais Luitzen Brouwer (1881-1966, ci-contre) faisait partie du courant intuitionniste. Pour lui, en mathématiques pures, il n'existe pas plus qu'ailleurs de langage absolument sûr. Par conséquent, le principe du tiers-exclu, qui n'affirme « rien d'autre que la décidabilité de tout problème », n'est pas toujours vrai.

Construire les mathématiques sur une base logique sûre

À l'aube de xx^e siècle, Frege et Russell tentèrent de faire reposer les mathématiques sur une base logique sûre. En particulier, ils s'efforcèrent de montrer que l'arithmétique se déduit entièrement d'axiomes logiques. Pour cela, ils construisirent, avec beaucoup de génie, la succession des nombres naturels à l'aide de concepts purement logiques : ils considérèrent la classe O *des objets qui ne sont pas identiques à eux-mêmes. Les objets non identiques à eux-mêmes n'existant pas, cette classe* O *est vide. Ils définirent alors le zéro comme la classe de toutes les classes comprenant autant d'objets que la classe* O.

La définition peut paraître artificielle, mais elle leur permit de construire tous les nombres naturels. La définition du nombre un utilise le fait qu'il n'existe qu'une seule classe vide : le nombre un est la classe des classes comportant autant d'éléments que la classe dont le seul élément est la classe vide. Le nombre deux est alors la classe des classes comportant autant d'éléments que la classe dont les éléments sont le zéro et le un ; et ainsi de suite… Hélas, ce début prometteur se heurta à un obstacle : dans une lettre de 1902, Russell attira l'attention de Frege sur un paradoxe, connu depuis sous le nom d'antinomie de Russell : appelons R *l'ensemble de tous les ensembles qui ne se contiennent pas eux-mêmes comme élément. Par exemple, l'ensemble des carottes n'est pas une carotte et cet ensemble appartient à* R. *En revanche, l'ensemble des idées est une idée et donc cet ensemble n'appartient pas à* R. *L'ensemble* R *se contient-il lui-même ? Si la réponse est non,* R *est un ensemble qui ne se contient pas lui-même comme élément. Il appartient donc, par définition, aux ensembles dont est composé* R ; *ce qui signifie que* R *se contient lui-même comme élément. Cependant, si l'ensemble* R *est un élément de lui-même, il appartient à l'ensemble des ensembles qui sont des éléments d'eux-mêmes et n'est donc pas un élément de* R. *D'une affirmation découle son contraire, et réciproquement. La contradiction est insoluble. Le rêve de Frege de faire reposer tout l'édifice des théories mathématiques sur le fondement de la logique formelle s'effondra avec la lettre de Russell. La définition des nombres naturels posait la question suivante : peut-on former à volonté des classes, des classes de classes, des classes de classes de classes, etc. ? L'antinomie de Russell montrait que quelque chose n'allait pas.*

Russell s'efforça de la résoudre en élaborant une « théorie des types » : selon cette théorie, des individus, des classes, des classes de classes, etc., doivent appartenir à différents types logiques. Il établit ainsi une hiérarchie de types logiques. Le théorème « La classe de tous les X *est un* X *» n'est ainsi ni vrai ni faux, mais tout simplement dépourvu de sens. Lorsqu'on remplace «* X *» par « humain », l'absurdité est évidente. Le problème devient intéressant quand «* X *» représente « la classe qui ne se contient pas elle-même comme élément ». L'astuce de la théorie des types de Russell empêche de formuler l'antinomie de façon sensée. Russell modifia alors la construction des nombres naturels, mais cela l'obligea à ajouter à ses axiomes un axiome d'infinité, selon lequel il existe dans le monde un nombre infini d'objets. Cette hypothèse impliquait, tout comme la théorie des types, une limite à la déduction purement logique de l'arithmétique initialement visée.*

propositions à leur forme grammaticale au sein de la deuxième axiomatique : par ce biais, il montrerait qu'aucune contradiction ne survient de leur usage dans les axiomatiques à contenu et qu'elles peuvent donc être utilisées par les mathématiciens. Comment, toutefois, caractériser cette deuxième sorte d'axiomatique ? On y trouve une seule sorte de propositions, répliques de celles de l'axiomatique à contenu. Ces propositions sont cependant dépourvues d'interprétation et forment un système grammatical réglé par la seule inférence logique conçue comme procédure effective, c'est-à-dire dans laquelle l'itération des règles se fait toujours dans un cadre fini. Dans ce type unique de propositions sont codés à la fois les signes mathématiques et les signes logiques : ainsi tous les signes sont-ils traités de la même manière et toutes les propositions sont-elles soumises à un patron commun, devenant des assemblages de signes matériels d'écriture. *L'axiomatisation formelle des axiomatiques à contenu joue donc, dans le cadre mathématique, un rôle analogue à celui de la grammatisation dans les langues naturelles.*

L'axiomatique formelle vise à engendrer une réplique du domaine de validité de l'axiomatique à contenu, constituée de propositions renvoyant à des domaines finis, potentiellement infinis ou transfinis. Si la réplique, bien que finitaire, est fidèle, on peut espérer répondre à la question de la non-contradiction de l'axiomatique formelle. Et en déduire si les axiomatiques à contenu sont contradictoires ou non, c'est-à-dire si l'on peut utiliser

Bertrand Russell, caricaturé dans une version populaire de son antinomie : le barbier du village est le villageois qui rase les villageois qui ne se rasent pas eux-mêmes. Le barbier se rase-t-il lui-même ? Oui et non…

dans des raisonnements mathématiques des propositions englobant l'infini actuel.

Pour que la réplique formelle de l'axiomatique à contenu soit fidèle, il faut éliminer la signification des énoncés et la difficile question de leur domaine de validité fini, potentiellement infini ou transfini pour s'en tenir à des conventions d'écriture gérant les rapports entre propositions formelles. Dans la transcription des formules en signes, la partie la plus délicate consiste à créer des répliques des propositions *transfinies*. Du point de vue des règles de déduction des formules, il faut aussi s'assurer que la déduction s'opère toujours de façon *effective*, c'est-à-dire en un nombre fini d'étapes.

Comment, à partir de là, prouver la non-contradiction de la réplique formelle de l'axiomatique à contenu ? En recherchant une *preuve d'impossibilité* : Hilbert suppose l'existence d'une contradiction entre les axiomes du système formel et essaye de montrer que cette supposition est elle-même contradictoire – le raisonnement par l'absurde étant considéré comme licite dans le cas d'un système fini. Si cette non-contradiction est établie, celle-ci se répercutera sur l'axiomatique à contenu arithmétique et, de là, sur les autres axiomatiques à contenu. Las, quelques mathématiciens, au premier rang desquels le logicien autrichien Kurt Gödel, s'aperçoivent que les outils formels ne captent pas tout ce qui fait l'objet propre de l'arithmétique.

Kurt Gödel à la terrasse d'un café à Vienne, en 1938. En 1931, Gödel fit chanceler l'édifice de Hilbert en montrant que l'axiomatique formelle n'est ni complète ni consistante.

Les limites du système hilbertien

Que s'est-il passé ? Kurt Gödel a découvert des *limitations internes à la formalisation qui compromettent définitivement* le projet hilbertien. En 1928, Hilbert avait formulé trois questions capitales concernant l'axiomatique formelle : premièrement, l'axiomatique formelle est-elle *complète* ? En d'autres termes, *toute* formule peut-elle y être démontrée ou réfutée ? Deuxièmement, l'axiomatique formelle est-elle *consistante*, au sens où *aucune formule contradictoire* ne peut y être engendrée à partir des axiomes ? Troisièmement, l'axiomatique formelle est-elle *décidable*, c'est-à-dire existe-t-il une *méthode effective* pour décider si une formule quelconque est vraie ou fausse ? Hilbert espérait fournir des réponses positives dans tous les cas : l'axiomatique formelle serait complète (elle engendrerait tous les théorèmes), consistante (elle n'engendrerait que les théorèmes) et décidable (il existerait une procédure effective pour décider si toute formule est ou non un théorème).

Les trois réponses, qui s'échelonnent entre 1931 et 1937, sont *toutes négatives*. La première est formulée par Gödel : une axiomatique formelle susceptible de servir de réplique à l'arithmétique des entiers est structurellement incomplète, car on peut montrer qu'il y a un « reste » arithmétique qui échappe à l'axiomatique formelle quels que soient les aménagements axiomatiques ultérieurs susceptibles de se produire. Le même Gödel répond négativement à la deuxième question : la consistance de l'arithmétique ne peut être démontrée dans le cadre de l'axiomatique formelle, du moins si l'on s'en tient à des procédures qui ne font pas intervenir l'infini actuel. La troisième réponse négative est énoncée par Church et par Turing indépendamment : il n'existe pas de procédure effective susceptible de *décider* dans tous les cas si toute formule est ou non un théorème. Le problème de la *décision* permet à Turing de préciser ce qu'il entend par *calcul* : à la fois un moyen de *compter* et un moyen de prendre des *décisions*. On retrouve ainsi la double étymologie du mot *calculus*.

Deux points concernant ces réponses négatives méritent d'être soulignés : d'une part, c'est par une

preuve d'impossibilité qu'elles ont été formulées ; d'autre part, il y a un reste qui échappe à la formalisation et il convient de s'interroger sur ce reste. Une preuve d'impossibilité, contrairement à une preuve classique qui dévoile le cheminement vers un résultat, consiste à montrer qu'un résultat est *inaccessible*. L'intérêt d'une telle preuve réside dans le fait qu'elle permet de *ne plus* chercher à obtenir un résultat, celui-ci étant reconnu hors de portée compte tenu des outils formels dont on s'est doté au départ. Cette preuve est donc *interne*, dans la mesure où, sans faire appel à un domaine de validité élargi, elle délimite – de l'intérieur – un périmètre de validité au-delà duquel on ne peut obtenir de nouveaux résultats. En pratique, une preuve d'impossibilité consiste à montrer que le résultat escompté serait contradictoire : on suppose au départ le résultat acquis et on montre qu'il entraîne une contradiction. Nous verrons que c'est par ce moyen que furent obtenues les réponses négatives qui anni-hilèrent le projet hilbertien, en particulier celle de Turing *(voir* La machine de Turing, *page 72)*.

Le deuxième point a trait à ce *quelque chose de plus* contenu dans l'arithmétique « naturelle », que l'axiomatique formelle ne parvient pas à capter. Quel est ce « plus » de l'arithmétique « naturelle » des nombres entiers ? À cette question difficile, sur laquelle mathématiciens, logiciens et philosophes des mathématiques ont débattu jusqu'à aujourd'hui, Turing a apporté des éléments de réponse au cours de son itinéraire intellectuel : nous verrons que, pour lui, ce qui échappe au processus de formalisation relève du champ *géométrique*, c'est-à-dire *précisément de ce que la formalisation axiomatique a tenté d'éliminer*.

Par champ géométrique, il faut moins entendre l'étude des figures et des principes de leur mesure que le *principe d'intelligibilité* qui permet d'appréhender les *formes structurées* dans *tous* les domaines des sciences de la nature, y compris en mathématique, et donc en arithmétique. Par ce biais, mathématique et physique retrouvent cette parenté séculaire qu'elles entretenaient au sein des sciences de la nature avant les bouleversements géométriques et arithmétiques du XIX^{e} siècle.

De ce point de vue, Turing occupe une position charnière dans le développement et la transformation du programme formaliste fondé par Hilbert : il a poussé le projet formaliste à ses limites extrêmes en étendant au maximum son champ de validité et, du même coup, tracé ces limites en s'aventurant au-delà.

Mécaniser l'esprit

Le projet de Hilbert et son échec relatif ont donné lieu à une étape ultime dans la mécanisation du monde, étape à laquelle Turing a puissamment contribué : la mécanisation de l'esprit.

Nous avons vu que dans le cadre de l'axiomatique formelle, l'enchaînement des propositions s'effectue à l'aide de la seule inférence logique, itérée un nombre fini de fois. Sur quoi repose cette limitation à un « nombre fini de fois » ? Il s'agit d'une pure *discipline de pensée*, d'un acte *mental* qui ne peut apparaître explicitement dans *aucune* règle puisqu'il est la condition de leur applicabilité. En d'autres termes, cette limitation est un postulat philosophique sur la nature du *mental*, qu'Hilbert exprime en 1923 sous la forme suivante, dans ses *Fondements logiques des mathématiques* : « [...] notre pensée est finitiste ; quand nous pensons se déroule un processus finitiste. » Ainsi, Hilbert, pour la bonne marche de l'axiomatique formelle, a supposé un principe que seul un postulat philosophique justifie.

Les travaux de Turing occupent une ligne de crête entre une vision classique du monde – déterministe, réductionniste et calculatoire – et une vision déterministe et non prédictive, qui est celle de la science moderne.

Turing, plus hilbertien qu'Hilbert, trouve le moyen de « grammatiser » ce principe en proposant une *contrepartie formelle* à l'acte mental finitiste. Cette contrepartie deviendra la définition même du *calcul* de Turing : un système constitué d'un nombre fini d'instructions, itérables un nombre potentiellement infini de fois. Ainsi, en 1936, dans l'article même où il répond négativement à la question de Hilbert sur la décidabilité des mathématiques, Turing

La notion de bon ordre n'est pas formalisable : on ne sait pas formaliser le fait que nous nous représentons la suite des nombres entiers sur une droite dans le sens de l'écriture, c'est-à-dire dans le bon ordre. L'esprit n'engendre pas que des actes mentaux formalisables.

se lance dans une analyse de l'acte mental sous-tendant toute inférence logique au sein de l'axiomatique formelle, c'est-à-dire l'acte mental sous-tendant tout calcul.

Au lieu de décrire de l'extérieur l'acte mental finitiste, comme l'avait fait Hilbert, Turing demande au lecteur de son article de se placer en pensée dans la situation d'effectuer les actes mentaux, et de les limiter au mouvement (finitiste) des signes : il incite donc le lecteur à entrer dans un cadre de pensée finitiste sans supposer qu'il y a un « extérieur » à ce cadre, comme le laisse encore entendre le postulat philosophique hilbertien. Dans cet état d'esprit, on peut alors dresser la liste finie des « comportements » qu'adopte un être humain en train de calculer – un « calculateur » *(computer)*, selon le terme employé par Turing : le processus mental devient une contrepartie formelle au sein même des signes.

Turing montre ainsi qu'il est possible de décrire les étapes (en nombre fini) de l'acte mental formaliste sous une forme tabulaire. Cette table, que Turing nommait « machine », est ce que nous appelons aujourd'hui un *programme*. En donnant une contrepartie formelle d'un acte mental postulé, Turing a défini le *formel* comme relevant intégralement du *mécanique*. Ultime étape de la mécanisation du monde, la mécanisation de l'esprit clôt le vaste mouvement qui débuta avec la grammatisation des langues du monde et la mécanisation de la nature. Ce mouvement de mécanisation s'achève en incluant *l'instrument même de la connaissance*, l'esprit lui-même.

L'intelligibilité informelle

La mécanisation de l'esprit n'est cependant pas le mot de la fin : le projet de Hilbert, bien qu'ayant permis de préciser un certain nombre de concepts, dont celui de calcul, a débouché sur une impasse. La stratégie de réduction des axiomatiques à l'arithmétique formelle n'a pas eu tout le succès espéré. Gödel, Turing et Church ont montré – grand succès rationnel du projet hilbertien que de démontrer ses propres limites ! – que le formalisme ne capte pas plusieurs caractéristiques propres de l'arithmétique, pourtant fondamentales. En particulier, certaines façons de penser en arithmétique, voire en mathématiques ou dans les sciences de la nature en général, n'ont pas de contrepartie dans la grammaire formaliste. Comme le fait remarquer le mathématicien G. Longo dans l'ouvrage coécrit avec le physicien F. Bailly *Mathématiques et sciences de la nature ; la singularité physique du vivant* (2006), un principe aussi fondamental que celui de *bon ordre* n'a pas de contrepartie formelle. Or ce principe est à la base même de notre *compréhension géométrique* de l'arithmétique des entiers, bien avant leur reconstruction logique au moyen de l'induction : c'est grâce au bon ordre, par exemple, que nous nous représentons la suite des nombres entiers sur une droite s'étendant indéfiniment dans le sens de l'écriture et que nous comprenons instantanément la notion de *plus petit élément*.

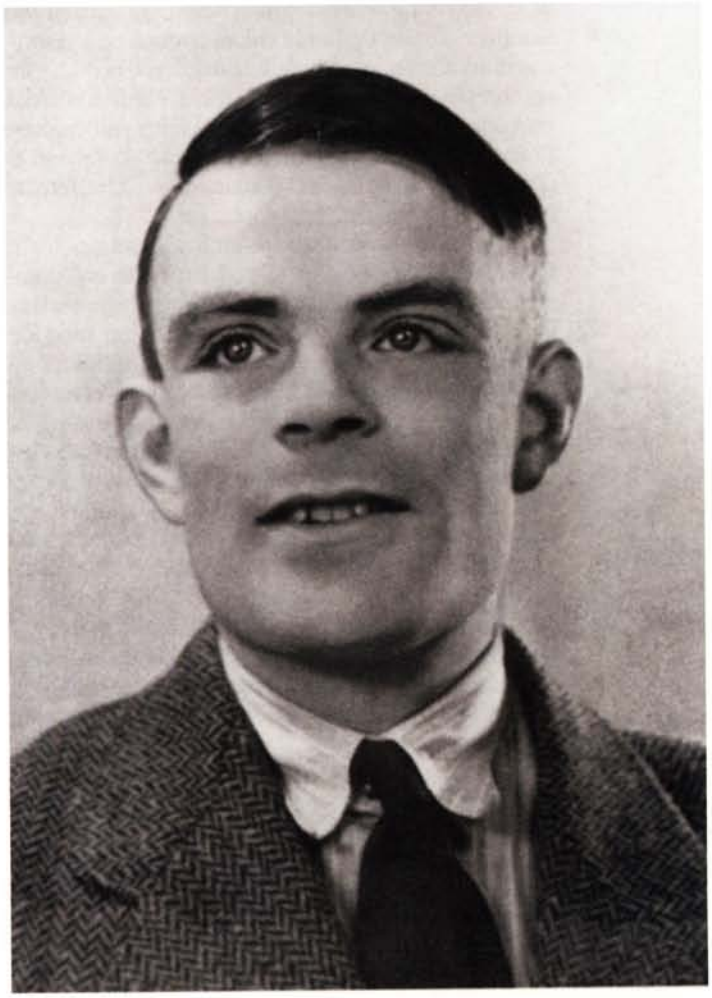
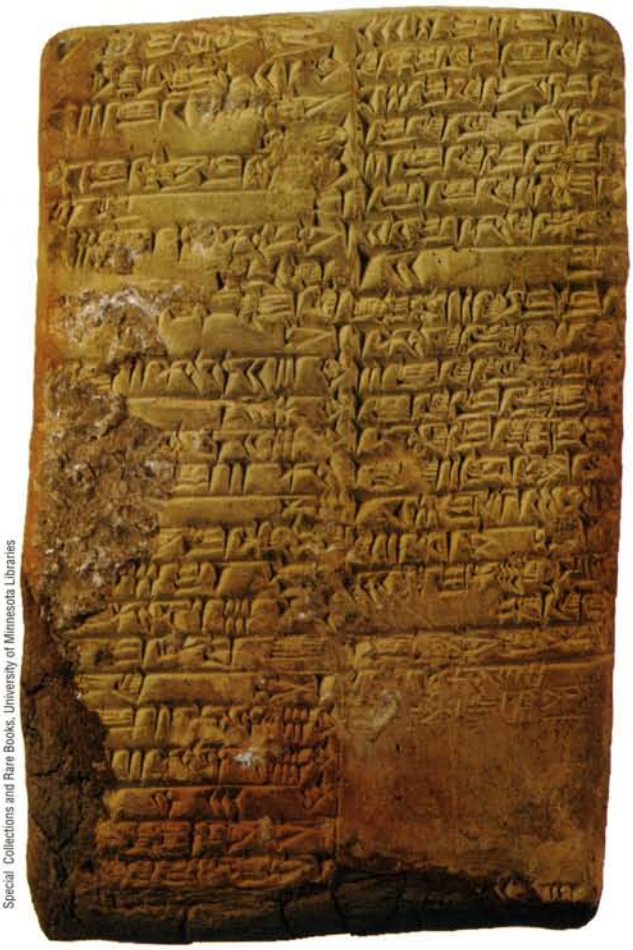


Photo Elliott & Fry

Alan Turing, dans les années 1950.

Il existe donc *autre chose* dans l'esprit que la contrepartie formaliste d'un acte mental finitiste, c'est-à-dire autre chose qu'un *calcul*. Et ce quelque chose est lié à la perception de *formes* ayant pour nous un *sens*, comme la représentation mentale de la droite numérique dotée d'un bon ordre. Une conséquence fondamentale découle de cette constatation : il est nécessaire de redéfinir la notion même d'esprit à partir de principes d'intelligibilité qui ne soient pas exclusivement mentaux – c'est-à-dire qui ne soient pas « contenus » *dans* un « intérieur », plus ou moins assimilé au « cerveau » –, mais déjà présents dans les langues naturelles ou la nature physique.

Cette énigme de *l'au-delà du formalisme*, autrement dit de l'*intelligibilité de formes porteuses de sens*, fait, pour nous aujourd'hui, toute la valeur des recherches futures de Turing. Ses travaux occupent une ligne de crête entre une vision déterministe, réductionniste et calculatoire du monde – dans la lignée de la science de l'âge classique – et une vision déterministe, mais non prédictive du monde, qui est celle de la science moderne et que Turing a magistralement contribué à fonder. ■



Special Collections and Rare Books, University of Minnesota Libraries

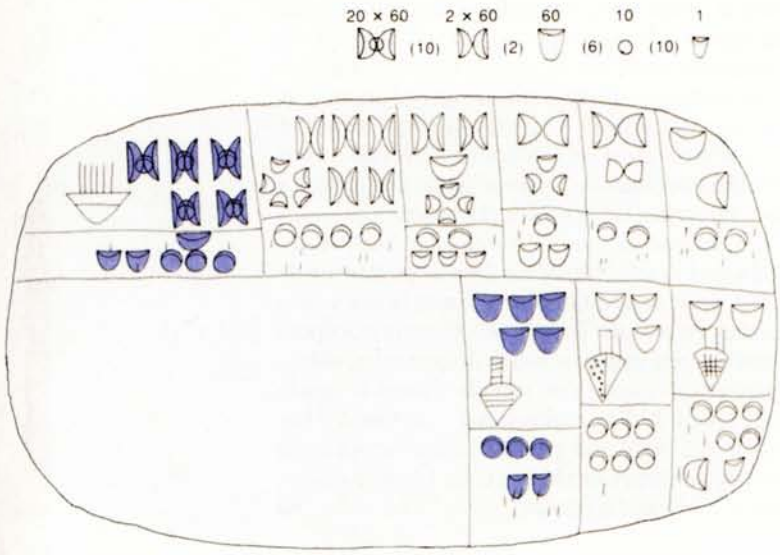
Mécaniser le calcul

Lorsque, au IXe siècle, Al-Khuwarizmi développe la notion d'algorithme, le calcul connaît une profonde révolution. Une nouvelle étape est franchie au début du XXe siècle, autour de la notion de calculabilité.

Nous avons vu que dans son article *On computable numbers*, Turing définit le calcul comme un système constitué d'un nombre fini d'instructions, itérables un nombre potentiellement infini de fois. Or cette démarche est connue et pratiquée depuis plusieurs millénaires. Pourquoi, alors, la définition de Turing bouleversa-t-elle notre vision du calcul ? Pour le comprendre, retournons aux sources de cette procédure.

La notion de calcul a toujours été associée à l'exercice même de l'activité mathématique, et on imagine mal qu'il puisse en être autrement. Cette ancienneté va même bien au-delà de ce que l'on imagine généralement : *l'invention de l'écriture* en Mésopotamie est liée à la représentation des nombres et au calcul, *parallèlement* à la transcription des sons des langues, comme en témoigne le grand nombre des tablettes les plus archaïques servant à la comptabilité. Ainsi, la découverte du caractère précis et déterminé du nombre est liée à la stabilité offerte par le support écrit et à l'aspect tabulaire des présentations qu'il rend possible, peut-être avant que ce type de support ne soit utilisé à d'autres fins telles que l'enregistrement des langues parlées.

Les découvertes archéologiques récentes en Mésopotamie nous enseignent une autre chose, capitale pour notre propos : mêlées aux tenues de compte des troupeaux ou des récoltes, les tablettes numériques mésopotamiennes contenaient des amorces de *procédés de calcul*, détachées de tout contexte et donc susceptibles d'avoir une portée générale. Ainsi, le support écrit permettait d'adopter un même angle d'attaque pour traiter des objets que l'on avait besoin de compter et des procédés de calcul abstraits ayant les nombres pour seul support. Certes, les civilisations de tradition orale possèdent des procédures de calcul, où les parties du corps et les gestes sont associés au



Un texte proto-sumérien (fin du IVe millénaire avant notre ère) exposant un exercice portant sur de grandes quantités de pain et de bière et utilisant un système numérique « bisexagésimal », précisé dans le schéma au-dessus du texte : dès la fin du IVe millénaire avant notre ère, les tablettes mésopotamiennes contenaient des ébauches de procédures de calcul.

	1358
+	947
	2305

Une addition, posée à l'occidentale. Cette présentation de l'opération est une procédure de calcul. On additionne successivement les chiffres d'une même colonne en ajoutant le nombre d'unités à la colonne suivante quand le résultat est plus grand que dix.
Page ci-contre, une tablette babylonienne du XXIe siècle avant notre ère, sur laquelle sont répertoriés des comptes d'animaux (chèvres, moutons). Les nombres et les calculs sont indissociables de l'invention de l'écriture en Mésopotamie.

dénombrement. Mais les comptables-mathématiciens de Mésopotamie s'en éloignent et introduisent une notion de calcul qui reste la nôtre : pour nous comme pour eux, un calcul est une *séquence de gestes en nombre fini à exécuter dans un certain ordre* sur un ensemble d'éléments ordonné à l'avance, séquence dont on peut indiquer les étapes par écrit, ce qui rend la procédure réutilisable à l'infini, dans différents contextes.

Prenons quelques exemples de procédures. Le plus familier est lié à l'écriture des langues : la transcription d'une phrase orale au moyen des lettres de l'alphabet exige l'application d'une procédure. Comme l'apprennent les écoliers, les sons du français sont rendus par des lettres ou des assemblages de lettres de l'alphabet. Évidemment, dans le cas des langues, la procédure de transcription n'est pas toujours rigoureuse et les écoliers ne connaissent que trop les pièges qui se cachent dans les nombreuses façons possibles de transcrire un même son. Il ne s'agit donc pas à proprement parler d'un *calcul*, parce que le résultat n'est pas uniforme. Néanmoins, l'exemple donne une idée de la nature d'une procédure : à partir d'un ensemble d'éléments ordonné au préalable (ici la concaténation des sons tels qu'ils sont ordonnés dans la parole pour former des mots), la procédure de transcription met en correspondance cet ensemble des sons et le répertoire des lettres composant l'alphabet.

Qu'est-ce qu'un calcul ?

Les procédures mathématiques sont plus rigoureuses car elles portent sur des nombres, dont la transcription en signes est univoque et uniforme. C'est le cas de celles développées pour effectuer les quatre opérations arithmétiques usuelles. Dans notre système décimal, par exemple, nous « posons » les opérations : nous superposons les nombres en colonnes correspondant aux chiffres des unités, dizaines, centaines, etc. *(voir la figure ci-dessus)*. Cette utilisation de la position des chiffres est une procédure de calcul, qui permet d'effectuer l'opération rapidement tout en minimisant les erreurs. À l'inverse, certaines activités ne peuvent se mettre sous forme de procédure : c'est le cas des jeux de hasard comme la roulette des casinos dont le cinéma a popularisé l'exemple en montrant des illuminés hantant les salles de jeux à la recherche de l'algorithme qui permettrait de gagner à la roulette à tous les coups…

Notons que la notion de procédure ne se limite pas au strict domaine numérique : elle a d'autres applications scientifiques, en géométrie ou en logique. Dans ce dernier cas, le mathématicien anglais George Boole (1815-1864) a conçu des moyens de vérification de la vérité ou fausseté des propositions élémentaires dans le cadre d'un calcul : il attribua les deux valeurs possibles *vrai* et *faux* à des éléments *P* et *Q* représentant des propositions et détermina les valeurs de vérité que ces éléments peuvent prendre quand ils sont soumis à des opérations logiques (la négation, la conjonction « et », l'exclusion « ou » et l'implication « implique », *voir la figure page 66*). La procédure qu'il a mise en place consiste à dresser un tableau où tous les cas possibles sont répertoriés.

Ainsi, toute question portant sur les éléments d'un ensemble qui peut recevoir une réponse sous la forme d'une suite réglée de gestes indéfiniment réitérables peut être résolue à l'aide d'une procédure, pourvu que l'on en circonscrive clairement les étapes. Bien entendu, la procédure ne fait pas le tout de l'activité mathématique ; elle est plutôt un regard porté *après coup* sur une activité *déjà conçue* et des objets *déjà mis en place* (par exemple la conception d'un ensemble bien ordonné d'éléments). Par conséquent, la mise au point d'une procédure de calcul recquiert une conception du

Certains problèmes ne sont pas solubles au moyen d'une procédure de calcul : aucune procédure, par exemple, ne peut déterminer sur quel nombre tombera la bille de la roulette, au grand dam des joueurs.

P	Q	non P	P et Q	P ou Q	P implique (P ou Q)
Vrai	Vrai	Faux	Vrai	Vrai	Vrai
Vrai	Vrai	Faux	Faux	Vrai	Vrai
Faux	Vrai	Vrai	Faux	Vrai	Vrai
Faux	Faux	Vrai	Faux	Faux	Vrai

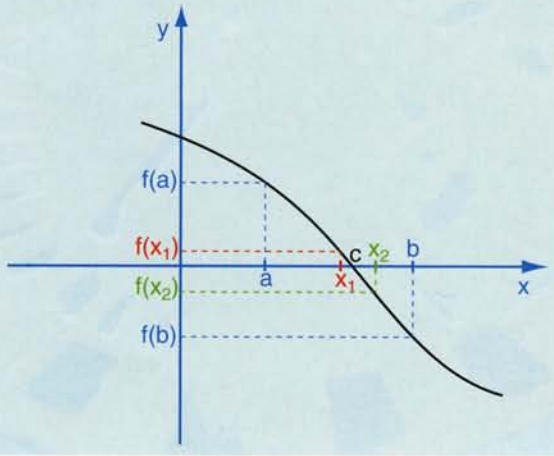
Une table de vérité telle celle ci-dessus est une procédure qui ne s'applique pas au domaine numérique, mais à des propositions vraies ou fausses, telles que « Turing aime les mathématiques » (proposition vraie) et « Pluton est une planète » (proposition fausse, semble-t-il aujourd'hui).

temps particulière : celle-ci implique d'une part que soit donné un ensemble d'éléments *hors du temps* et, d'autre part, que lors de l'exécution de la procédure, l'intervention du temps ne définisse qu'une séquence *finie*.

Il faut distinguer ici le calcul d'une valeur exacte de celui d'une valeur approchée. Dans le cas d'une valeur exacte, le calcul s'arrête au bout d'un temps fini. Dans le cas d'une valeur approchée, atteindre le degré d'approximation défini à l'avance marque la fin du calcul, qui se poursuivrait indéfiniment sinon. Ainsi, le calcul est toujours fini dans son exécution, que le résultat attendu soit une valeur exacte ou seulement approchée. De ce point de vue, un calcul approché peut durer indéfiniment si on cherche une précision toujours plus grande (comme en témoigne par exemple l'expansion décimale infinie d'un nombre comme π), mais à un degré d'approximation fixé à l'avance, une procédure de calcul se termine toujours par un résultat parce qu'une telle procédure est toujours constituée d'instructions en nombre fini.

Un algorithme de calcul

Pour approcher une valeur c pour laquelle une fonction f s'annule, on procède par dichotomie : on détermine deux nombres a et b tels que les valeurs f(a) et f(b) soient respectivement positive et négative, et on calcule la valeur de $f(x_1)$ pour le milieu x_1 de a et b. Si $f(x_1) = 0$, on a trouvé la solution. Si $f(x_1)$ est positive, on calcule la valeur $f(x_2)$ pour le milieu x_2 de x_1 et b. Si $f(x_1)$ est négative, on calcule la valeur $f(x_2)$ pour le milieu x_2 de a et x_1. On regarde de même le signe de $f(x_2)$ et on réitère le procédé jusqu'à obtenir une valeur x_n qui s'approche de c avec la précision souhaitée.



L'algorithme : une vieille histoire

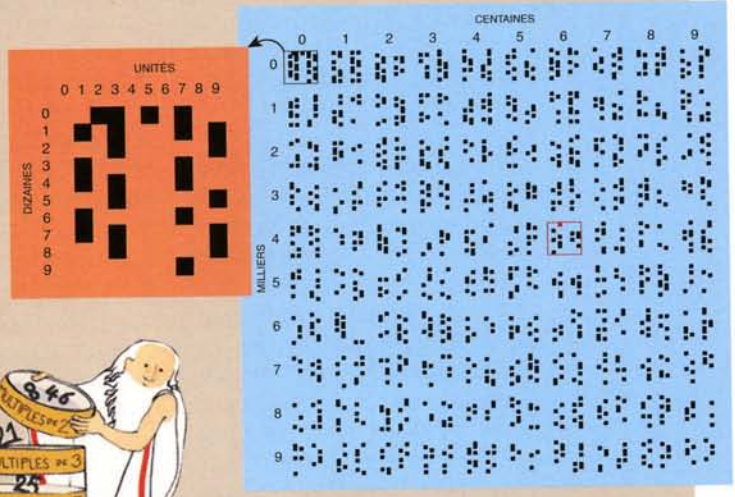
Les procédures de calcul sont donc très anciennes et on en retrouve la trace à toutes les étapes de l'histoire des mathématiques et dans toutes les civilisations. Au début du deuxième millénaire avant notre ère, par exemple, les Babyloniens utilisent une procédure qui donne l'inverse $1/n$ d'un nombre n *(voir la figure page 67)* et leur permet ainsi de diviser par le nombre n (la division « directe » n'étant pas pratiquée). Au IIIe siècle avant notre ère, les Grecs dressent la liste des nombres premiers grâce à une procédure appelée « crible d'Ératosthène » *(voir l'encadré page ci-contre)*. Entre les IIIe et Ier siècles avant notre ère est composé le Classique de la tradition mathématique savante de la Chine ancienne, *Les Neuf chapitres sur les procédures mathématiques*, qui présente les procédures de référence de la culture chinoise. Au XIIe siècle, les Indiens développent une procédure tabulaire pour effectuer les multiplications, où une place est aménagée à chaque étape pour les calculs intermédiaires *(voir la figure en haut de la page 70)*.

Cependant, la notion de procédure de calcul ne fut considérée pour elle-même, en tant qu'objet mathématique à part entière, que dans le monde musulman médiéval : jusqu'alors utilisée en mathématiques en tant qu'outil pratique, la notion de procédure de calcul ne devint un objet mathématique qu'avec les premiers traités d'algèbre de cette époque, en particulier ceux d'un mathématicien persan de langue arabe, Muhammad ibn Musa Al-Khuwarizmi (fin du VIIIe siècle-début du IXe siècle). Le nom de ce mathématicien donna dans les langues européennes, *via* l'Espagne arabe, le terme « algorithme » qui, depuis, désigne une procédure de calcul.

Pourquoi Al-Khuwarizmi a-t-il éprouvé le besoin de considérer la procédure de calcul comme un objet mathématique ? Une question générale de l'algèbre consiste à se demander si une équation possède une solution dans le domaine numérique donné au départ. Cette question implique de circonscrire le domaine numérique dans lequel les solutions possibles existent. Par exemple, dans l'ensemble des entiers naturels positifs, les opérations de soustraction et de division n'ont pas de solution lorsque leur résultat est un nombre négatif ou une fraction. Une telle circonscription équivaut à préciser le domaine où il n'existe pas de solution (en l'occurrence le

Le crible d'Ératosthène

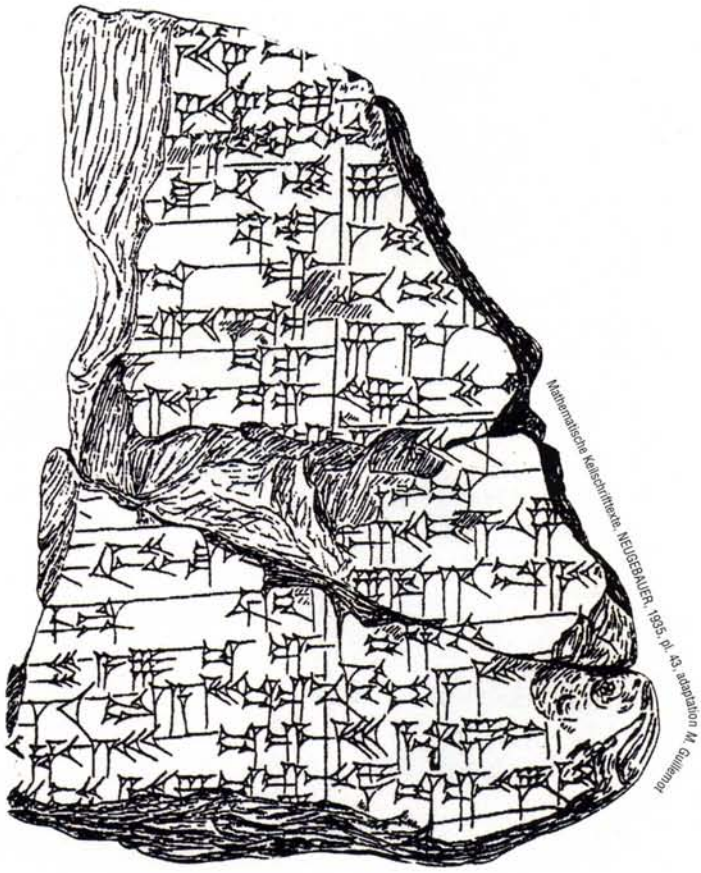
Dès le IIIe siècle, les Grecs avaient trouvé une procédure de calcul pour dresser la liste des nombres premiers : le crible d'Ératosthène. Le crible est évoqué dans le premier volume de Introductio arithmetica *de Nicomaque de Gérase (vers 100) : pour déterminer les nombres premiers inférieurs à un entier* n, *on écrit la suite des nombres compris entre 2 et* n. *On supprime ensuite tous les multiples (supérieurs à 2) de 2, c'est-à-dire un nombre sur deux à partir de 2, puis tous les multiples de 3, c'est-à-dire un nombre sur trois à partir de 3, puis tous les multiples de 5, c'est-à-dire un nombre sur cinq à partir de 5, puis de même les multiples de 7, de 11, de 13 et ainsi de suite. Les nombres restant à la fin du tri constituent la totalité des nombres premiers inférieurs à* n.



En rouge, le crible d'Ératosthène pour n = 100. Les nombres premiers sont indiqués par des carrés noirs. Par exemple, sur la première ligne apparaissent les nombres premiers inférieurs à 10 : 2, 3, 5, 7. En bleu, le crible d'Ératosthène pour n = 10 000 : chaque bloc représente une centaine. Par exemple, le bloc entouré en rouge correspond à la centaine entre 4 600 et 4 699. Pour savoir si 4 603 est premier, on repère dans ce bloc l'intersection de la ligne 0 et de la colonne 3 : elle est occupée par un point (en rouge), donc 4 603 est premier. Les colonnes paires de chaque bloc (sauf la colonne 2 du premier) sont toutes vides, car elles contiennent les multiples de 2. Les colonnes dont le numéro se termine par 0 ou 5 contiennent des multiples de 5 et sont aussi vides.

domaine des entiers négatifs). De même, l'étude algébrique des cas non intuitifs comme celui des équations du second degré – domaine de recherche engagé par Al-Khuwarizmi – ou d'un degré supérieur nécessite la définition du domaine numérique dans lequel les solutions possibles existent. Or ce domaine, s'il existe, *doit être accessible au moyen d'une procédure de calcul*. On comprend dès lors le rôle que joua la notion d'algorithme pour elle-même : elle permit non seulement de répondre à une question générale (par exemple comment trouver les nombres premiers ?), mais de s'interroger sur la *solvabilité en général* d'une question pour un domaine donné. Étudier la nature et la portée des algorithmes selon les domaines numériques où ils interviennent devint alors une tâche mathématique à part entière.

Quelle fut la nature de l'intervention de Turing dans cette histoire multi-millénaire ? En un mot, il contribua à définir la nature de la notion d'algorithme dans le cadre du formalisme hilbertien en précisant ce que signifie « circonscrire un domaine de solutions », c'est-à-dire en précisant les rapports entre les deux aspects de la notion d'algorithme : d'une part la délimitation



Reproduction d'une tablette datée de l'ancien âge babylonien (2000-1650 avant notre ère) où est mis en œuvre un algorithme pour déterminer l'inverse d'un nombre à travers cinq exemples. La démarche consiste à calculer l'inverse d'un nombre à partir des inverses de deux nombres plus petits, déjà connus et répertoriés dans une table « standard ».

Entretien avec Clarisse Herrenschmidt, chercheur au CNRS, Laboratoire d'anthropologie sociale du Collège de France.

Pour la Science : ***Vous êtes anthropologue de l'écriture, spécialiste de l'Orient ancien et de l'apparition de la monnaie frappée en Grèce ancienne. À première vue, votre terrain de recherche ne vous prédisposait pas à vous pencher sur la naissance de l'informatique, ni à rencontrer un personnage comme Turing. Pourtant, dans votre dernier ouvrage à paraître prochainement* Les trois écritures. Langue, nombre, code *(Collection Sciences humaines, Gallimard, Paris, 2007), vous consacrez toute une partie à l'informatique et au rôle qu'a joué Turing dans la naissance de celle-ci. Qu'est-ce qui vous a amenée à vous intéresser à Turing et à l'informatique ?***

Clarisse Herrenschmidt : *L'informatique est la troisième étape, contemporaine, d'une vieille histoire, celle de l'écriture, considérée dans la zone qui va du Moyen-Orient au Proche-Orient, du Proche-Orient à la Méditerranée (Grèce antique, Rome), et en Europe médiévale, moderne et contemporaine (l'ensemble Chine-Corée-Japon ayant une autre histoire graphique). Dans cette zone, la volonté de rendre* visibles *les nombres favorise le développement des trois grands types d'écritures : écriture de la langue, du nombre et du code. La première étape a commencé à Uruk et Suse, en 3 300 avant notre ère, par la représentation de nombres pour une chose, suivie de l'écriture des langues. La deuxième étape est l'apparition en Grèce, vers 620 avant notre ère, de la monnaie frappée : la pièce de monnaie est vecteur et support d'une écriture arithmétique, où le nombre est une forme abstraite. Le rapport entre écriture, nombre et monnaie a évolué grâce à l'emprunt, en Europe du Moyen Âge, du 0 et des chiffres indo-arabes que nous utilisons toujours. La troisième étape, qui commence à peine, est celle de l'informatique, nouvelle écriture de nombres. Turing ouvre cette étape de l'histoire des signes en révélant un nouvel aspect des nombres. Voilà pourquoi il occupe une place éminente dans ma recherche.*

Les premiers signes d'écriture, vers 3 400 avant notre ère, provenant de Suse et d'Uruk : il y a autant de marques numérales que de quantités à noter.

Pour la Science : ***Vous montrez que l'écriture est d'abord liée, à l'origine, à l'écriture des nombres. Pouvez-vous préciser le rapport entre calcul, écriture des nombres et écriture des langues ?***

Clarisse Herrenschmidt : *Au IV^e millénaire avant notre ère, au Moyen-Orient, l'écriture est reliée à des pratiques comptables. On passait un contrat entre deux personnes, l'une livrant à l'autre un certain nombre d'objets, par exemple les vaches d'un troupeau. Pour matérialiser le contrat, on fabriquait des boules creuses d'argile, les bulles-enveloppes, dans lesquelles on plaçait des* calculi, *sorte de billes représentant les quantités des objets à dénombrer. Ces boules creuses étaient scellées pour garantir l'authenticité de l'accord des contractants. Puis on les confiait au messager qui devait acheminer les biens. À l'arrivée, on cassait la boule d'argile pour vérifier si les quantités fournies étaient bien les quantités commandées.*
Plus tard, on inscrivit à la surface des boules des signes représentant les calculi *qu'elles contenaient : ce sont les premiers « chiffres ». Enfin, on remplaça les boules par des tablettes sur lesquelles étaient écrites les quantités des biens dont on gardait la trace et des signes*

Bulle-enveloppe et* calculi *de Suse, en argile crue, vers 3 300 avant notre ère. Sa surface porte des signes en forme d'encoches, identiques aux calculi enfermés à l'intérieur : cela évitait de briser l'enveloppe pour en connaître le contenu.

représentant la nature de ces biens. L'écriture est apparue en Mésopotamie et en Iran à la faveur de ce processus. Elle permet de manipuler les nombres – il y eut de grands mathématiciens en Mésopotamie et en Égypte – mais pas seulement. L'écriture des langues consiste en une analyse de segments de plus en plus petits du discours : division de l'énoncé en mots (systèmes à logogrammes) ; division du mot en syllabes (syllabaires) ; division de la syllabe en consonnes et espaces (alphabets consonantiques) ; division de la syllabe en consonnes et voyelles (alphabets consonantiques et vocaliques).

Pour la Science : ***Pour vous, quels rapports entretiennent monnaie et calcul, à travers l'histoire du développement des écritures ?***

Clarisse Herrenschmidt : *La notion savante de nombre ne diffuse pas en dehors de cercles étroits. En revanche, la circulation de la monnaie, laquelle résulte de l'écriture des nombres, a joué un rôle capital dans les pratiques calculatoires de cette partie du monde. Apparue en Italie à la fin du Moyen Âge, la comptabilité en partie double, mettant en rapport le débit et le crédit, ou autrement dit la dépense et les moyens dont on dispose pour l'honorer, en est un exemple. La monnaie transforme en pratique sociale le nombre et, de façon plus générale, le calcul arithmétique.*

Pour la Science : ***Comment réagissez-vous à la phrase de Turing, écrite en 1950 : « Pour moi, mécanisme et écriture sont presque synonymes » ?***

Clarisse Herrenschmidt : *C'est une description minimale de l'ordinateur (qui n'écrit que des nombres) dans le contexte de l'histoire générale de l'écriture. Potentiellement, tout traitement relève d'un mécanisme. La généralisation de l'écriture des nombres passe par une généralisation de la notion de traitement et, par conséquent, par une généralisation du cadre mécanique. Les machines ont bien un lien intime avec l'écriture des nombres et, indirectement, avec l'écriture des langues. Vous voyez qu'on n'est pas loin de la phrase de Turing !*

Pour la Science : ***Nous parlions à l'instant de traitement. On présente souvent l'informatique comme une expression du mental, lequel serait identifiable à un traitement, justement. Pour vous au contraire, l'apparition de l'informatique relève davantage de la pratique sociale collective que représentent l'écriture et la monnaie, que de la psychologie individuelle. Comment, selon vous, s'articulent la notion de traitement et la sphère du mental ?***

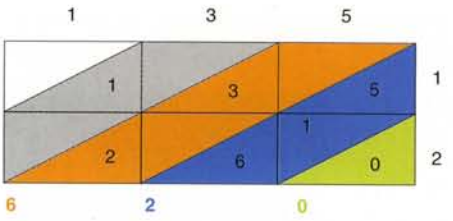
Clarisse Herrenschmidt : *On a eu trop souvent tendance à « mentaliser » le traitement informatique et même, encore récemment, à identifier fonctionnement mental et traitement informatique. Mais cela revient à occulter toute l'histoire de l'écriture des nombres et des langues : quand on envisage l'esprit comme une machine, on ne se rend plus compte que la machine elle-même est le produit d'une société et du développement progressif des techniques de l'écriture. L'écriture rend* visibles *d'abord les nombres et leurs rapports, ensuite les langues parlées. Cette visibilité nous permet de déléguer à des machines qui nous sont extérieures le soin d'effectuer une partie du traitement sur les signes : voilà ce qui constitue la profonde originalité de la culture écrite dans la région du monde dont nous parlons, et non pas le dévoilement anhistorique de je ne sais quelle nature du « mental ».*

Pour la Science : ***Turing est l'auteur d'un célèbre théorème d'impossibilité touchant le calculable, théorème qui est une des bases de l'informatique. Comment l'interprétez-vous du point de vue anthropologique ?***

Dagli Orti

Luca Pacioli (1445-1517), moine mathématicien italien, présente dans son ouvrage* Summa de arithmetica, geometria, de proportioni et de proportionalita *(Venise, 1494) la* méthode vénitienne *de tenue des comptes, aujourd'hui dénommée comptabilité en partie double.

Clarisse Herrenschmidt : *Quand on se rend compte que le système des signes que l'on possède n'est pas adapté à la réalité mathématique que l'on vise, on est amené à transformer le système des signes en question. C'est ainsi depuis que les mathématiques existent en tant que savoir autonome. Turing ne fait rien d'autre à partir de la découverte de son théorème d'impossibilité de 1936 : il trace des limites au calculable dans le système d'écriture qu'est la machine de Turing, se demande si ces limites sont stables, se pose la question de leur dépassement possible, en mathématiques mais aussi en physique et en biologie. De ce point de vue, il est l'héritier de plein droit d'un trésor intellectuel que nous cultivons depuis plusieurs millénaires, et dont je retrace, dans mon livre, les grandes étapes.*



Dès le XII^e siècle, les Indiens effectuaient les multiplications à l'aide d'une procédure de calcul. Ci-dessus, un exemple de multiplication donné dans un commentaire du livre indien le Lîlâvati de Bhaskara (XII^e siècle), écrit par Ganesa, astronome indien du XVI^e siècle. La multiplication de 135 par 12 est présentée sous forme de table. Les résultats des produits élémentaires sont disposés dans chaque case de la table, les unités étant placées dans le triangle inférieur et les dizaines dans le triangle supérieur. Puis on somme les unités, les dizaines, etc.

d'un périmètre de calculabilité et, d'autre part, la détermination des solutions. Turing montra qu'en délimitant un périmètre de calculabilité pour un problème traduit formellement, on s'assure que des solutions existent et que cela vaut la peine de les chercher. Avec Turing, l'algorithme devint non seulement un objet mathématique, mais un objet mathématique que l'on peut manipuler formellement, au même titre que le nombre en arithmétique. À partir de ses travaux, un nouveau champ de logique mathématique s'est constitué dans les années 1930 –la théorie de la *calculabilité* – qui a rendu possible l'avènement d'une nouvelle discipline, l'informatique.

Les nombres de Gödel

Rappelons la situation : nous sommes en 1931. Hilbert a lancé un vaste programme dont le but est de représenter toutes les propositions mathématiques par le biais d'une axiomatique formelle *(voir* La mécanisation du monde, *page 56)*. Dans ce contexte, Gödel a répondu par la négative aux deux premières questions posées par Hilbert : l'axiomatique formelle est-elle complète (toutes les propositions vraies sont-elles démontrables à partir des axiomes du système formel) ? Est-elle consistante (peut-on démontrer, à partir des axiomes du système formel, que l'axiomatique formelle est non-contradictoire) ? En outre, sa démonstration suggère que la réponse à la troisième question de Hilbert, le problème de la décision (peut-on toujours décider si une proposition est vraie ou fausse à partir des axiomes du système formel ?), est aussi négative. Voyons pourquoi.

Gödel mit au point une technique originale pour parvenir à son résultat. Il partit d'un système formel qui empruntait les axiomes logiques à un système formel que Russell et Whitehead avaient conçu en 1910-1913 *(Principia Mathematica)* et les axiomes arithmétiques à l'arithmétique formalisée de Peano. Il y coda les propositions en leur attribuant des nombres entiers : étudier les propositions dans le système formel revenait alors à étudier les nombres entiers qui les représentent, ces nombres entiers étant soumis aux lois classiques de l'arithmétique *(voir* Gödel. Logique à la folie, Les Génies de la Science *n°20, août 2004)*. Appliquant le même raisonnement que le paradoxe du menteur qui énonce : « cette phrase est fausse », Gödel montra que, dans son système formel, la phrase « La proposition de nombre x n'est pas démontrable » peut être représentée par une formule arithmétique $F(x)$, et que cette formule arithmétique s'applique au nombre f qui la représente dans le système formel, c'est-à-dire à elle-même. En d'autres termes, il construisit la formule $F(f)$, qui dit d'elle-même qu'elle n'est pas démontrable, tout comme le paradoxe du menteur rend impossible la distinction du vrai et du faux. Cela signifie que dans un système d'axiomes que l'on suppose non-contradictoire tel que le système formel de Gödel, la formule $F(f)$ dit d'elle-même qu'elle n'est pas démontrable formellement ; elle est donc indécidable, au sens où l'on ne peut pas décider à l'aide des axiomes si elle est vraie ou fausse.

Ainsi, le théorème de Gödel semble régler le problème de la décision : puisqu'il existe au moins une propriété indécidable, la réponse au problème doit être négative. Toutefois, cinq ans après l'article de Gödel, une question se pose encore : le théorème de Gödel est-il général ou spécifique à la façon dont le logicien a associé des nombres entiers aux propositions ? S'il n'est pas général, la réponse négative qu'il apporte au problème de la décision est peut-être spécifique à la démarche de Gödel.

La généralité du théorème de Gödel dépend de la façon dont on définit la notion de *calculabilité*. En effet, pour coder sous forme arithmétique les propositions et les algorithmes de démonstration de son système formel, Gödel a défini une classe de fonctions susceptible d'opérer cette mise en rapport. Mais cette classe de fonctions arithmétiques, dites « calculables » puisqu'elles opèrent dans le cadre strict de l'arithmétique finitiste, est-elle circonscrite de façon adéquate ? Gödel a conçu *une* façon de définir ces fonctions, mais s'agit-il de *la* façon ? Bref, est-on sûr que le résultat négatif de Gödel n'est pas seulement l'effet d'un codage arithmétique *particulier* des propositions, lié à une restriction trop étroite de la classe des fonctions calculables employées ? Mais alors, qu'est-ce qu'une fonction calculable ?

Les grands esprits se rencontrent

« Grâce à certains travaux qui ont suivi cet article [sur l'incomplétude de l'axiomatique formelle], en particulier ceux de A. M. Turing, nous disposons désormais d'une définition sûre, précise et adéquate du concept de système *formel* [...] dont la propriété est qu'en son sein, et en principe, le raisonnement peut être entièrement remplacé par des règles mécaniques. »

Kurt Gödel, note du 28 août 1963 ajoutée à son article de 1931.

De l'arithmétique pour étudier l'arithmétique

En 1931, Kurt Gödel (en médaillon) *publia un article intitulé* Sur les propositions formellement indécidables des *Principia Mathematica* et des systèmes formels apparenté (voir la première page ci-contre) *qui mit à mal le projet de Hilbert de construire une axiomatique formelle dont toutes les mathématiques seraient déduites. Il montra qu'il existerait toujours, dans une telle axiomatique, au moins une proposition vraie non démontrable. Pour arriver à son résultat, il utilisa un procédé puissant et subtil : il partit de l'axiomatique formelle de l'arithmétique de Peano et associa à chaque objet de cette axiomatique (nombre, proposition, formule, etc.) un et un seul nombre entier. Ainsi, il utilisa le formalisme bien connu de l'arithmétique de Peano, mais dans un contexte où les nombres ont une autre signification, pour étudier la consistance… de l'arithmétique de Peano.*

Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I[1]).

Von Kurt Gödel in Wien.

1.

Die Entwicklung der Mathematik in der Richtung zu größerer Exaktheit hat bekanntlich dazu geführt, daß weite Gebiete von ihr formalisiert wurden, in der Art, daß das Beweisen nach einigen wenigen mechanischen Regeln vollzogen werden kann. Die umfassendsten derzeit aufgestellten formalen Systeme sind das System der Principia Mathematica (PM)[2]) einerseits, das Zermelo-Fraenkelsche (von J. v. Neumann weiter ausgebildete) Axiomensystem der Mengenlehre[3]) andererseits. Diese beiden Systeme sind so weit, daß alle heute in der Mathematik angewendeten Beweismethoden in ihnen formalisiert, d. h. auf einige wenige Axiome und Schlußregeln zurückgeführt sind. Es liegt daher die Vermutung nahe, daß diese Axiome und Schlußregeln dazu ausreichen, alle mathematischen Fragen, die sich in den betreffenden Systemen überhaupt formal ausdrücken lassen, auch zu entscheiden. Im folgenden wird gezeigt, daß dies nicht der Fall ist, sondern daß es in den beiden angeführten Systemen sogar relativ einfache Probleme aus der Theorie der gewöhnlichen ganzen Zahlen gibt[4]), die sich aus den Axiomen nicht

La thèse de Church-Turing

La notion de calculabilité relève d'une définition et non d'une démonstration. Or une définition possède toujours un aspect arbitraire ; ainsi, pour s'assurer la pleine généralité du théorème de Gödel, on doit s'entendre sur la définition de la calculabilité. C'est précisément là que se situe l'intervention de Turing, ainsi que celle de deux mathématiciens-logiciens américains, Alonzo Church (1903-1995) et son élève Stephen Kleene (1909-1994).

Par des approches très différentes, tous trois montrent que les différentes formulations avec lesquelles ils définissent les fonctions calculables dessinent *le même périmètre* pour le domaine de la calculabilité. Stephen Kleene propose alors cette conjecture, qu'il appelle « thèse de Church-Turing » : une fonction est dite calculable si elle peut s'exprimer dans une des formes proposées par Church ou par Turing. Il s'agit d'une conjecture parce qu'elle porte sur le *pouvoir expressif* des définitions de la calculabilité : elle propose une équivalence entre ces définitions et ce que l'on admet intuitivement comme relevant du calcul. Il n'y a donc pas de preuve envisageable de cette thèse, mais seulement de bonnes raisons informelles de parier qu'elle est adéquate pour englober tout le domaine du calcul.

Musée de l'Observatoire de Ulugh Begh, Ungentsch (Kharezm), Ouzbékistan.

Muhammad Ibn Musa Al-Khuwarizmi (environ 783-850), sur une gravure sur bois réalisée par un artiste Ouzbek en 1983 à partir d'un manuscrit persan. Al-Khuwarizmi mit en évidence l'algorithme en tant qu'objet mathématique à part entière. Il écrivit en outre l'un des premiers traités d'algèbre – terme arabe signifiant « réduction » que l'Europe a conservé tel quel.

Cette conjecture appliquée à la démarche de Gödel dans ses travaux de 1931 donne un résultat clair : tout formalisme finitiste de type hilbertien encode *la même classe de fonctions*. Ce résultat, par la stabilité qu'il manifeste malgré la diversité des voies qui y mènent, renforce le théorème de Gödel et lui assure la pleine généralité qui lui faisait encore défaut.

Ainsi est-ce *à l'occasion* du problème de la décision que Turing propose sa définition de la calculabilité. Il s'agit donc là d'une simple *application* du cadre général qu'il conçoit pour clarifier la notion de calculabilité, comme le souligne le titre de son article de 1936 *Théorie des nombres calculables, suivie d'une application au problème de la décision*.

Parmi les approches de la notion générale de calculabilité, celle de Turing est la plus convaincante, au dire de Gödel lui-même, parce qu'elle lui paraît être la plus proche de l'intuition que l'on se fait de la notion de calcul tout en restant indépendante de tout formalisme particulier. C'est ce qui fait d'ailleurs que la méthode de Turing n'est pas seulement une démonstration de logique mathématique au sens strict du terme, comme nous allons le voir. ■

La machine de Turing

En 1936, les logiciens savent que l'on ne pourra jamais, dans un système formel, décider si une proposition est vraie ou fausse, mais ne l'ont pas démontré. Un jeune inconnu résout le problème à l'aide d'une curieuse machine de papier.

Dans son article *On computable numbers with an application to the Entscheidungsproblem*, Turing cherche, à l'instar de nombre de mathématiciens et de logiciens, à résoudre le problème de la décision. Au lieu de se placer d'emblée dans le cadre de la logique mathématique et d'expliciter directement la notion de démonstration pour elle-même, comme l'aurait fait un Hilbert, Turing reformule la notion en termes de nombres calculables. L'expression « nombre calculable » peut surprendre : que peut être un nombre sinon une entité *calculable* ? Mais il ne faut pas oublier qu'avec les techniques mises au point par Gödel, les nombres servent à coder des propositions dans un système formel *(voir page 70)*. Or comme celui-ci a montré qu'il existe des propositions codées par des nombres qui ne sont pas démontrables, les nombres qui les codent sont *non-calculables*. C'est sur le terrain des nombres codant les propositions de l'axiomatique formelle que travaille Turing.

Il commence par proposer une définition de la calculabilité : cette définition est nécessaire pour préciser quels sont les nombres qui appartiennent ou appartiendront au domaine du calculable. L'approche de Turing est tout à fait originale : il compare un être humain en train de calculer (que Turing appelle aussi un « calculateur » – *computer*) et une machine à calculer : « Nous pouvons comparer un être humain en train de calculer un nombre réel à une machine susceptible d'avoir seulement un nombre fini d'états. »

Sa comparaison est en fait une identification pure et simple : un nombre est dit calculable si la liste explicite d'instructions nécessaire à son calcul *peut être intégralement déléguée à une machine*. Qu'est-ce que Turing entend par « machine » ? La machine qu'il décrit n'est en rien une machine matérielle. Il n'a pensé à aucune machine en particulier pour en construire le plan, si ce n'est peut-être au principe de défilement des images d'un film dans un projecteur de cinéma. Cette machine, appelée aujourd'hui « machine de Turing » ou « automate abstrait », est, comme il l'écrit lui-même, une machine « de papier » décrivant de façon rigoureuse comment on passe d'une suite de symboles écrits à une autre suite selon un ordre réglé d'avance.

Une machine de Turing possède un nombre fini d'états internes. Ici, la machine est dans l'état 13. Elle modifie cet état en fonction d'une liste finie d'instructions et de ce qu'elle lit sur les cases d'un ruban. Elle change aussi éventuellement les chiffres inscrits sur les cases du ruban. Ainsi, la machine dans l'état 12 et qui lit le chiffre 1 sur la case du ruban passe à l'état 13, change le 1 en 0 sur la case du ruban et se déplace d'une case à gauche en suivant l'instruction (12, 1 => 13, 0 gauche), avant d'attaquer l'instruction suivante.

Il s'agit d'une machine *mathématique* permettant la manipulation réglée de signes. En cela, elle se distingue des machines matérielles. Au premier abord, d'ailleurs, contrairement aux machines usuelles qui regorgent de câblage, de roues et de pièces dont la fonction nous échappe, la machine de Turing est d'une description si abstraite et si simple qu'on doute qu'elle parvienne à effectuer non seulement *un* calcul, mais plus encore *tous* les calculs et, par ce biais, toute inférence dans un système formel. C'est en tout cas la première réaction de Max Newman, le professeur de topologie de Turing à Cambridge qui reçoit le manuscrit dactylographié des mains de son étudiant en avril 1936. Max Newman se convainc cependant de la justesse de l'analyse de Turing et reconnaît que le jeune homme, alors âgé de 24 ans, a non seulement résolu l'un des problèmes de logique mathématique les plus ardus de l'époque, mais proposé une définition si générale de ce qu'il faut entendre par calcul que le problème de la décision n'en est plus qu'une application particulière.

Une machine de Turing n'est donc pas une machine au sens courant du terme : c'est plutôt une « boîte noire » dont on ne précise pas le fonctionnement matériel et qui n'a ni force motrice ni énergie électrique. Il s'agit d'une *machine algorithmique* qui opère une transformation de symboles fournis en entrée en symboles lisibles en sortie, au moyen d'une succession d'états discrets qui sont tous définissables à l'avance. Cette succession d'états définit les étapes de l'algorithme que la machine exécute, c'est-à-dire ce que nous appelons aujourd'hui, depuis l'article de 1936, un *programme*. Ainsi la machine consiste en la mise en rapport algorithmique de deux ensembles : d'une part un ensemble de symboles d'entrée et d'autre part un ensemble d'états de sortie. Il est possible, à partir de ce schéma minimal, de fournir une description du mécanisme régissant les états de la machine.

ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By A. M. TURING.

[Received 28 May, 1936.—Read 12 November, 1936.]

The "computable" numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means. Although the subject of this paper is ostensibly the computable *numbers*, it is almost equally easy to define and investigate computable functions of an integral variable or a real or computable variable, computable predicates, and so forth. The fundamental problems involved are, however, the same in each case, and I have chosen the computable numbers for explicit treatment as involving the least cumbrous technique. I hope shortly to give an account of the relations of the computable numbers, functions, and so forth to one another. This will include a development of the theory of functions of a real variable expressed in terms of computable numbers. According to my definition, a number is computable if its decimal can be written down by a machine.

In §§ 9, 10 I give some arguments with the intention of showing that the computable numbers include all numbers which could naturally be regarded as computable. In particular, I show that certain large classes of numbers are computable. They include, for instance, the real parts of all algebraic numbers, the real parts of the zeros of the Bessel functions, the numbers π, e, etc. The computable numbers do not, however, include all definable numbers, and an example is given of a definable number which is not computable.

Although the class of computable numbers is so great, and in many ways similar to the class of real numbers, it is nevertheless enumerable. In § 8 I examine certain arguments which would seem to prove the contrary. By the correct application of one of these arguments, conclusions are reached which are superficially similar to those of Gödel†. These results

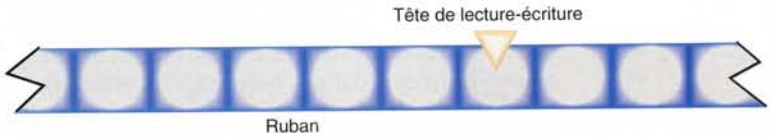
† Gödel, "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I", *Monatshefte Math. Phys.*, 38 (1931), 173–198.

La première page de l'article de 1936 de Turing
On computable numbers with an application to the Entscheidungsproblem.

De la machine au calcul

Une machine de Turing possède une *capacité de stockage externe* qui se présente sous la forme d'un *ruban* de longueur indéfinie, divisé en cases sur lesquelles sont portés des symboles. La machine est dotée d'une *tête de lecture-écriture* capable d'observer le contenu des cases du ruban, de se déplacer le long du ruban dans un sens ou dans un autre et de s'arrêter sur une case. Toutes les actions sont régies par une *table d'instructions* qui indique quelle action entreprendre – écriture ou mouvement. L'observation d'une case (sa lecture) entraîne soit l'effacement de son contenu, soit l'écriture de celui-ci. À chaque pas de temps, la tête de lecture-écriture observe une case et une seule. Le couple formé par l'état interne de la machine à un moment t et la case observée définit une *configuration* de la machine. La table d'instructions prescrit ainsi un comportement pour chaque configuration dans laquelle la machine peut se trouver. La machine effectue alors ce qui est prescrit par la table et produit un résultat. Ce mécanisme *suffit à décrire* la transformation qui affecte les symboles d'entrée pour en faire des symboles de sortie.

On peut alors représenter la machine de Turing sous la forme suivante :



Quelle que soit la tâche remplie par la machine, on peut toujours interpréter sa table d'instructions comme représentant le calcul d'une fonction d'entiers à valeurs entières. Une fonction est alors dite *Turing-calculable* quand ses valeurs peuvent être calculées par une machine de Turing. Ainsi, grâce au formalisme de la machine de Turing, la découverte d'un algorithme pour la résolution d'une classe donnée de problèmes est équivalente à celle d'une machine de Turing spécifique capable de fournir, dans un temps fini, la ou les solutions à la classe de problèmes en question. Il suffit alors d'inventer le

programme correspondant à l'algorithme pour que la machine exécute l'algorithme toutes les fois où cela est nécessaire.

Prenons pour exemple de calcul « minimal » celui donné par Turing dans le paragraphe 3 de son article *On Computable Numbers…* : la machine mise en place calcule la suite indéfinie 0101010101… Au départ, le ruban est vierge. La machine proposée par Turing pour calculer cette suite possède quatre états possibles *b*, *c*, *e*, *f*, et la table d'instructions est la suivante :

(état *b*, case vide) implique (état *c*, Inscription 0, Droite)
(état *c*, case vide) implique (état *e*, Droite)
(état *e*, case vide) implique (état *f*, Inscription 1, Droite)
(état *f*, case vide) implique (état *b*, Droite)

où « Inscription *x* » signifie « Inscription de *x* dans la case » et où « Droite » signifie « déplacement d'une case vers la droite ». La première ligne d'instructions signifie par exemple : lorsque la machine est dans l'état *b* et lorsque la tête de lecture lit une case vide sur le ruban, alors la machine passe dans l'état *c*, inscrit 0 dans la case et se déplace d'une case vers la droite. En suivant ces instructions, la machine imprime la suite 01010101… sur le ruban, en séparant chaque symbole par une case vide (pour plus de clarté) :

Le calcul de la suite continue indéfiniment puisque le ruban est vide. Ainsi, un calcul d'une longueur indéfinie peut être engendré par un programme fini d'instructions (en l'occurrence, un programme de quatre lignes). Ce « raccourci » manifeste deux traits capitaux de la notion de calcul par machine de Turing. Premièrement, il apparaît clairement – et de façon plus intuitive que dans le cas de l'algorithme – que l'aspect déterminé d'un calcul ne dépend en rien de sa longueur. Deuxièmement, il devient aussi beaucoup plus intuitif que des configurations mises au point pour effectuer des parties purement répétitives des instructions puissent resservir dans d'autres contextes. En d'autres termes, ces parties répétitives n'ont pas besoin d'être effectuées à nouveau : il suffit de reprendre les parties d'une table d'instructions où ces parties répétitives ont été rédigées sous forme d'instruction.

La machine universelle

Turing introduit un « raccourci » supplémentaire, d'importance considérable pour la théorie du calcul : la *machine universelle*. Jusqu'à présent, il a montré dans son article de 1936 comment un programme exécutable par une machine de Turing est l'image d'un algorithme. Toutefois, chaque calcul nouveau exige une nouvelle table d'instructions : le calculateur humain, selon l'algorithme qu'il veut mettre en pratique, construit telle ou telle machine de Turing. D'un point de vue psychologique, le calculateur utilise toujours le même ressort pour effectuer cette correspondance : pour tel algorithme, utiliser telle table d'instructions. La correspondance entre un algorithme et une table d'instructions fait donc l'objet d'une procédure générale. Cette procédure générale de mise en correspondance *ne pourrait-elle pas être elle-même opérée par une machine* ? En effet, on peut concevoir des machines de Turing dites « universelles » ayant la particularité d'effectuer *n'importe quel algorithme, si leur table d'instructions est capable de recevoir et d'exécuter les instructions de n'importe quelle table d'instructions* : « Il est possible d'inventer une machine unique qui peut être utilisée pour calculer n'importe quelle suite calculable. Si cette machine *U* est munie d'un ruban au début duquel est inscrite la description standard d'une machine à calculer *M*, alors *U* calculera la même suite que *M*. »

De même qu'un calculateur humain est capable de s'adapter aux différents calculs qu'il doit exécuter (une addition par ci, une multiplication par là) – ou à tout autre algorithme selon les différents problèmes qu'il rencontre –, la machine universelle calcule, selon les instructions qui lui sont confiées, ce que différentes machines de Turing peuvent calculer. L'universalité de ces machines de Turing provient donc de leur universelle capacité à rester fidèles aux instructions des machines qu'elles imitent. Les machines universelles revêtent un intérêt considérable pour qui tente de déterminer le champ du calculable, dans la mesure où elles réduisent tout calcul à la construction de la table d'instructions d'une seule machine. Grâce à une machine universelle, les tables d'instructions des autres machines sont effectuables sur une seule machine.

Le raccourci opéré par une machine universelle se situe à une autre échelle que celui effectué par une simple machine de Turing. Le concept de machine universelle implique un usage méthodique général de la réutilisation de toute instruction, quelle qu'elle soit. En d'autres termes, il est possible, par le biais d'une machine universelle, de combiner en une table d'instructions de plus en plus complexe des tables d'instructions effectuant des calculs plus simples : il suffit pour cela de réduire tout calcul à n'être qu'une partie d'un calcul plus vaste. Ainsi, non seulement chaque calcul de longueur arbitraire est fini par définition, mais *l'infinité des calculs elle-même* est virtuellement contenue dans une seule machine. Turing fait ici un formidable raccourci de ce qu'il faut entendre par calcul : dans un premier temps, il conçoit une machine qui, à partir d'un algorithme construit pour un problème, résout tous les problèmes du même type (une machine de Turing élaborée pour effectuer une addition sait résoudre toutes les additions) ; dans un second temps, il construit une machine qui, non seulement résout ce type de problèmes, mais tous les calculs de toutes les machines de Turing.

Et le problème de la décision ?

Muni de son puissant concept de machine, Turing montre, à titre d'application, que le problème de la décision tel qu'il a été posé par Hilbert est insoluble. Pour comprendre sa démonstration, précisons d'abord ce problème : Hilbert recherchait un moyen de décider, dans un système formel contenant l'arithmétique, si toute proposition est vraie ou fausse. Ce faisant, il espérait que toute proposition vraie était démontrable. Or Gödel a montré l'incomplétude de l'axiomatique formelle, c'est-à-dire qu'au moins une proposition vraie restera toujours non démontrable. La démontrabilité ne peut donc plus être le critère pour décider si une proposition est vraie ou fausse. Néanmoins, ne serait-il pas possible de distinguer les propositions démontrables des propositions non démontrables au sein du système formel, les propositions démontrables devant être le résultat d'un algorithme ? Les propositions déductibles des axiomes étant en nombre potentiellement infini, on ne peut les examiner toutes les unes après les autres, car certaines sont encore à venir, et c'est ce qui rend le problème difficile. Car s'il n'y avait qu'à examiner les propositions démontrées et non pas aussi les démontrables, il suffirait de posséder une bonne définition de la notion de démonstration pour opérer cette distinction.

On sait qu'une proposition démontrable est reliée aux axiomes par un chemin de longueur finie (sinon la proposition n'est pas démontrable). Y aurait-il moyen de tester la longueur du chemin allant des propositions aux axiomes avec la seule chose à disposition, le nombre entier servant de code à la proposition ?

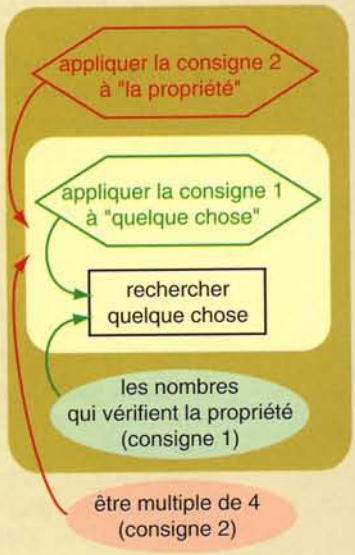
Dans ce nombre serait inscrite, sous forme arithmétique, la propriété d'être démontrable. Par exemple, si toute proposition démontrable avait pour code un nombre multiple de deux, alors il suffirait d'un algorithme très simple examinant si le nombre inspecté est divisible par deux pour savoir *à l'avance* si la proposition est démontrable.

Il faudrait donc disposer d'une procédure de calcul qui, pour toute proposition du système formel, *déciderait* si une proposition est ou non démontrable en vérifiant la présence ou l'absence de la propriété arithmétique de démontrabilité dans le nombre qui la code. Une telle procédure de calcul serait évidemment un *algorithme* qui, pour chaque nombre entier codant une proposition, détecterait si ce nombre possède ou non la propriété arithmétique codant la propriété « être démontrable ». Ce procédé permettrait de ranger la proposition dans la catégorie des propositions démontrables par simple inspection de son code numérique.

L'exécution d'un tel algorithme de décision reviendrait ainsi au calcul d'une fonction sur des nombres entiers (formant la liste des codes des propositions du système formel) dont toutes les

Le lambda-calcul d'Alonzo Church

En 1936, quelques jours avant que Turing ne remette son manuscrit à Newman, Alonzo Church et son élève Stephen Kleene publient leur définition de la calculabilité : le λ-calcul. Selon les deux logiciens américains, toutes les fonctions intuitivement calculables peuvent être exprimées en λ-calcul. Cette proposition, nommée thèse de Church, *n'est pas démontrable, car la notion de « intuitivement calculable » n'est pas formalisée. Néanmoins, la méthode de Church rend possible la constitution de la théorie mathématique la plus efficace pour rendre compte de ce que l'on entend par « calculable ». Le λ-calcul est un formalisme fondé sur la seule notion de fonction : tout élément de ce langage est une fonction à un argument (l'argument étant lui-même une fonction à un argument, etc.). Pour cela, une fonction est écrite, dans ce formalisme, à l'aide d'un λ-opérateur, qui la transforme en son action sur son argument, ce dernier étant complètement indépendant de cette définition. Considérons par exemple la fonction f qui à x associe 2x, alors λf est le concept abstrait de « doubler », que l'on peut appliquer à n'importe quel argument, et sur lequel on peut faire des déclarations formelles. Le λ-calcul permet de construire pas à pas toutes les fonctions calculables, des nombres naturels aux suites récurrentes et séries entières, en passant par les valeurs booléennes VRAI et FAUX. De façon plus générale, le λ-calcul de Church permet de formaliser toute entité mathématique, linguistique et,* a fortiori, *informatique, en une succession d'instructions qui combinent variables, applications de consignes et résultats de λ-calculs précédents. Prenons par exemple la phrase mathématique « Rechercher l'ensemble $\{x ; \exists n \in \mathbb{N}\ (x = 4n)\}$ ». Cette phrase s'écrit aussi « Rechercher les nombres qui vérifient la propriété "être un multiple de 4" ». En λ-calcul, elle devient le schéma d'instructions ci-dessous, qui signifie : 1) utiliser la λ-fonction « rechercher quelque chose », et 2) remplacer l'argument « quelque chose » par la consigne « les nombres qui vérifient la propriété ». Nous avons ainsi obtenu une nouvelle λ-fonction : « rechercher les nombres qui vérifient la propriété » ; 3) remplacer l'argument « propriété » de cette nouvelle λ-fonction par la consigne « être multiple de 4 » (qui est elle-même la fonction « être multiple d'un nombre », appliquée à l'argument « 4 », lui-même défini par une fonction du λ-calcul).*



*Le λ-calcul deviendra plus tard le fondement de plusieurs langages de programmation, dont le LISP (*List Processing Language*) et le SML (*Standard Meta Language*).*

La machine de Turing peut continuer indéfiniment des calculs, comme le calcul des chiffres après la virgule quand ceux-ci sont en nombre infini (par exemple quand il y a un « reste » inéliminable dans le calcul d'une division). Que signifie « s'arrêter » pour une machine de Turing ? S'arrêter parce qu'elle est parvenue au bout de son calcul, par exemple, lorsque le reste d'une division est nul et qu'elle en produit donc le résultat ? Ou s'arrêter parce que la transition entre deux étapes du programme a été mal rédigée et que le programme, ne pouvant pas traiter la donnée à l'étape où il se trouve, s'arrête faute d'instruction adéquate ?

Le « problème de l'arrêt » consiste à se demander s'il est possible de distinguer a priori ces deux cas. Si cette distinction pouvait être faite, on pourrait alors circonscrire intégralement la classe des programmes qui ne s'arrêtent pas pour des raisons de faute d'écriture, c'est-à-dire des programmes écrits dans une langue parfaite. Ces programmes seraient déterministes : ils ne laisseraient aucune place au hasard entre leurs étapes.

Arrêt et déterminisme

Pour répondre à cette question du point de vue mécanique, il faudrait posséder un programme qui repérerait les erreurs d'écriture causant l'arrêt intempestif de tout programme au cours de son exécution (ce qu'aujourd'hui nous appelons familièrement un « bogue »). La question posée par le problème de l'arrêt revient donc à celle-ci : un programme peut-il décider à l'avance, si un programme quelconque s'arrête ou pas ? Si tel était le cas, il serait en quelque sorte possible de trouver un résultat avant même qu'il ne soit effectivement obtenu. Cependant, le type de « raccourci » que la machine de Turing rend possible ne permet pas d'obtenir un résultat semblable.

Pour se faire une première idée de cette impossibilité, supposons qu'un tel programme « décisionnel » appelé A (pour « arrêt ») existe et qu'au seul vu des programmes et des entrées qui lui sont soumis, il décide à l'avance si les programmes en question s'arrêteront ou pas. Le programme A examinerait le calcul exécuté par un programme B, puis afficherait son résultat (« B s'arrêtera » ou « B ne s'arrêtera pas ») et s'arrêterait. Toutefois, en pratique, on voit que si le calcul de B ne s'arrête pas, A ne peut décider si le programme B, pour telle entrée, est mal conçu et entre dans une « boucle » indéfinie ou s'il poursuit seulement son calcul indéfiniment ; dès lors, A ne pourra pas décider, à un moment particulier du temps, si B s'arrêtera ou pas… à moins de se projeter à la fin des temps, ce qui est impossible mécaniquement.

Démonstration par l'absurde

Il est possible de retrouver ce résultat sans passer par une projection temporelle, en employant un raisonnement logique sur le comportement supposé du programme « décisionnel » à l'égard de lui-même. Pour ce faire, montrons que l'on peut « mal concevoir » intentionnellement un programme et que ce défaut de conception, inhérent à la notion d'arrêt, est indétectable au moyen d'un programme. En effet, puisqu'on suppose qu'il existe un programme « décisionnel », on doit pouvoir décider, comme pour tout programme, s'il s'arrête ou pas ; dans le cas présent, cette décision relève de lui-même puisqu'il est le programme décisionnel. Voyons à quoi peut conduire ce cas de figure.

Supposons que l'on classe tous les programmes dans une liste en leur associant un nombre. En outre, supposons qu'un programme décisionnel A existe dans cette liste et construisons-en une variante qui, au lieu de s'arrêter comme A une fois le résultat affiché, adopte le comportement suivant : le programme A'(p, *i*) ne s'arrête pas si le nombre p décrit un programme qui s'arrête pour une entrée i et le programme A'(p, *i*) s'arrête si le nombre p décrit un programme qui ne s'arrête pas pour une entrée i.

Maintenant, penchons-nous sur le comportement du programme A' lui-même : le programme A' porte le numéro *a'* de la liste des programmes. Considérons le programme A' appliqué à lui-même, c'est-à-dire le programme A'(a', *a'*). Dans quelle catégorie le ranger ? S'arrête-t-il ou pas ? Refaisons le même raisonnement que précédemment : le programme A'(a', *a'*) s'arrête si le nombre a' décrit un programme (en l'occurrence A') qui ne s'arrête pas pour l'entrée a'. En d'autres termes, le programme A'(a', *a'*) s'arrête si le programme A' ne s'arrête pas. Par conséquent, le programme A' est contradictoire.

Le fantôme de l'arrêt

Ainsi, il existe au moins un cas pour lequel un programme A', construit sur le modèle du programme A dont on supposait la validité universelle, conduit à une contradiction. Le programme décisionnel A n'existe donc pas. Quant à la notion d'arrêt, elle mène une existence fantomatique : il est impossible de la ranger dans la catégorie « défaut d'écriture (bogue) » ou dans la catégorie « poursuite du calcul ». Il n'y a pas de déterminisme absolu dans l'écriture des programmes.

Laplace et Jupiter

Pour les partisans du déterminisme en physique classique, les événements futurs sont prédictibles : même si des perturbations sont présentes dans le système physique, celles-ci conservent le même ordre de grandeur et interviennent peu sur l'évolution du système, à quelque moment que ce soit. Le déterministe Pierre-Simon Laplace (1749-1827) pensait ainsi que le système jovien constitué de Jupiter et des quatre satellites découverts par Galilée en 1610 était prédictible, car, expliqua-t-il dans sa Théorie des satellites de Jupiter, *la théorie (lois du mouvement et lois de la gravitation universelle) «a non seulement expliqué la cause des inégalités que les observations ont fait connaître, mais elle a développé les lois de toutes les inégalités qui, en se combinant entre elles, offraient aux astronomes des résultats trop compliqués pour qu'ils aient pu démêler les inégalités simples dont ils étaient formés. Elle a banni tout empirisme des Tables des satellites de Jupiter, et celles [de] M. de Lambre étant fondées sur la théorie de la pesanteur universelle, elles ont l'avantage de s'étendre à tous les temps, en rectifiant les données que l'observation seule peut déterminer».*

Portrait de famille composite de Jupiter et de ses quatre principales Lunes, de haut en bas Io, Europa, Ganymède et Callisto. Les images de Jupiter, Io et Ganymède furent prises en 1996 par la sonde* Galiléo, *celle de Callisto en 1979 par la sonde* Voyager *1.

valeurs seraient calculables puisqu'elle devrait pouvoir déterminer, dans tous les cas, si le code de telle ou telle proposition contient ou non la propriété arithmétique exprimant sa prouvabilité. Par conséquent, le problème de la *calculabilité* de cette fonction, et donc celui de la détermination de l'algorithme correspondant, est capital dans le cadre du « programme » de Hilbert : il est la voie d'accès à la détermination du domaine général du démontrable. Or le résultat auquel Turing parvient est dévastateur : *cet algorithme n'existe pas.*

Le résultat négatif de Turing

Voyons les grandes lignes du raisonnement de Turing sur le lien déductif entre axiomes et propositions. Pour qu'une proposition soit démontrable, la « longueur » du chemin qui la relie aux axiomes doit être finie, c'est-à-dire doit être de nature algorithmique. Peut-on savoir si une proposition quelconque est le résultat d'un algorithme ? Cette question équivaut à la suivante : sachant que l'on peut faire une liste des propositions produites exactement par algorithme (les propositions déjà démontrées), est-on sûr de pouvoir classer toutes les propositions produites par algorithme – même celles qui seront produites –, dans cette liste ?

La réponse de Turing est non : il exhibe un problème qui ne peut pas être résolu algorithmiquement. La liste elle-même, si elle vise l'exhaustivité, a dû être produite par un algorithme : à quel niveau *dans* cette liste ouverte placer l'algorithme qui produit la liste ? Cette place ne peut pas être définie sans l'aide de la liste *en train d'être construite*, et un cercle vicieux se met en place. Il existe donc au moins un algorithme qui n'est pas contenu dans la liste, qui est pourtant censée les contenir tous. Par conséquent, il n'existe pas d'algorithme de vérification de l'appartenance des algorithmes à la liste de tous les algorithmes ; il n'est donc pas possible de vérifier par un algorithme unique la longueur finie de tous les liens entre propositions et axiomes. Il faut répondre par la négative au problème de la décision : il n'existe aucun moyen algorithmique de déterminer si une proposition est démontrable ou non.

Pour appuyer ce raisonnement, Turing exhibe un problème dans le cadre de sa machine, le « problème de l'arrêt ». Ce problème a intuitivement une forme calculatoire et, pourtant, ne reçoit pas de solution en termes de machine de Turing. Il s'énonce de la façon suivante : y a-t-il moyen de *prévoir à l'avance* si une machine de Turing va s'arrêter ou non, c'est-à-dire si le calcul en train d'être exécuté sera mené ou non à son terme, au seul vu de son programme ? Un tel moyen signifierait qu'il existerait une machine « décisionnelle » qui connaîtrait *globalement* le comportement de chaque machine de Turing (c'est-à-dire le résultat du calcul effectué par la machine : son arrêt ou son absence d'arrêt) à partir de son aspect *local* (c'est-à-dire à partir de la simple inspection du contenu de sa table d'instructions). Or une telle machine est contradictoire et, par conséquent, n'existe pas *(voir l'encadré page 76)*. Turing ramène alors le problème de la décision au problème de l'arrêt en montrant qu'ils ont la même forme.

Nous avons mentionné l'autre application de la notion de machine de Turing au domaine de la logique mathématique *(voir page 71)* : elle concerne le rapport qu'entretient la machine avec les différentes définitions de la calculabilité, en particulier celle donnée par Church. Turing montre dans un appendice à son article de 1936, écrit lors de son séjour à Princeton, que le formalisme de la machine de Turing et celui du lambda-calcul de Church ont la même expressivité. En montrant que ce que Church appelle la « normalisation des lambda-termes » et l'arrêt d'une machine de Turing aboutissent à la même notion de résultat d'un calcul, il prouve du même coup que le domaine de la calculabilité est identique dans les deux formalismes, généralisant ainsi la perspective ouverte par les résultats de Gödel.

Calcul et déterminisme

Dans les années 1930, Turing fait figure d'« outsider » complet dans le champ de la logique mathématique : jusqu'alors, il a travaillé en calcul des probabilités où il a trouvé une nouvelle démonstration d'un théorème déjà prouvé en 1922 – le théorème de la limite centrale *(voir page 49)* – et en théorie des groupes où, en 1935, juste avant de s'intéresser à la logique mathématique, il a complété un résultat de von Neumann, dont il avait suivi les cours lors d'un séjour de ce dernier à Cambridge. Depuis l'adolescence, il s'est tourné vers les mathématiques appliquées à la chimie et ses lectures l'ont porté au moins autant vers la physique que vers les mathématiques : il a, par exemple, lu les ouvrages de Hilbert et de von Neumann, qui ont tous deux puissamment contribué à l'avancée des mathématiques et de la physique.

C'est en mathématicien et physicien, autant qu'en logicien, que Turing a abordé le problème de la calculabilité.

Pourquoi Turing est-il entré si vite dans le champ, à l'époque peu développé, de la logique mathématique ? Parce qu'il existe un lien caché qui unit ces disciplines : le *déterminisme* sous-jacent aux démarches classiques en physique et en mathématique. Comme l'ont montré le physicien F. Bailly et le mathématicien G. Longo, il existe une *parenté forte* entre le déterminisme développé en physique dans le cadre classique (dont la figure emblématique est Laplace) et le déterminisme mathématique (dont la figure emblématique est Hilbert), pour lequel un système d'axiomes contient virtuellement la totalité de ses théorèmes.

Événement futur dépendant strictement des événements passés et théorème dérivant strictement des axiomes du système formel *ont donc en commun d'être pleinement déterminables. Ce lien déterministe s'exprime précisément sous l'aspect du calcul.* En d'autres termes, c'est *l'aspect algorithmique qui unit mathématique et physique au sein du paradigme du déterminisme classique.* On appréhende mieux, dès lors, la synthèse qu'opère Turing entre mathématique et physique quand il cherche à décrire avec toute la précision requise la nature de la notion de calcul : il souhaite *mettre au jour le fondement déterministe du paradigme classique, que ce soit en mathématique ou en physique.*

De ce point de vue, on comprend comment il a pu investir avec autant de facilité et de puissance le champ réputé très abstrait de la logique mathématique : ce n'est pas seulement en logicien qu'il a abordé la question de la calculabilité posée par Hilbert, mais autant en mathématicien et en physicien. En témoigne le « problème de l'arrêt » qu'il brandit dans son article de 1936 à l'appui de sa démonstration selon laquelle il n'y a pas d'algorithme qui puisse résoudre le problème de la décision : ce problème est emblématique du déterminisme classique dans la mesure où il porte sur ce qui peut être *déterminé à l'avance.*

La réponse négative au problème de l'arrêt entraîne donc aussi l'effondrement du paradigme déterministe classique : l'analogie si patiemment construite entre d'une part la physique déterministe, que l'on peut qualifier de « laplacienne » selon Turing lui-même et, d'autre part, l'axiomatique formaliste ou « hilbertienne », s'en trouve *fragilisée* puisque l'axiomatique formaliste hilbertienne ne tient pas toutes ses promesses. Nous verrons que le travail ultérieur de Turing consistera à s'interroger sur la validité de l'autre branche de l'analogie, la physique déterministe « laplacienne » et sur les rapports que doivent entretenir physique et mathématique. *La construction matérielle du premier ordinateur est le résultat de cette réflexion.* ■

Informatique, géométrie et déterminisme

Entretien avec Giuseppe Longo, informaticien-théoricien au CNRS. Il travaille au Département d'Informatique de l'École Normale Supérieure, où il dirige l'équipe Complexité et Information Morphologique (voir http://www.di.ens.fr/~longo/).

Pour la Science : ***Le titre de votre équipe de recherche comporte les deux termes « information » et « morphologie ». Bien qu'ils aient une racine commune, la forme, tout semble opposer le registre algébrique de l'information et le registre géométrique de la morphologie... Comment les réconciliez-vous en tant que logicien et informaticien ?***

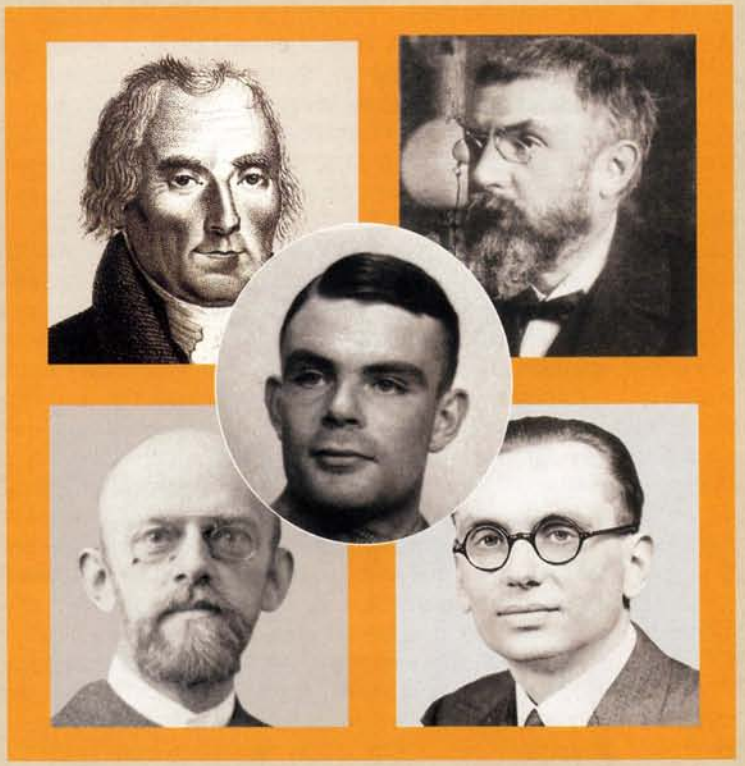
Giuseppe Longo : *On a des idées trop arrêtées et trop étroites sur la logique et l'informatique ! Cela vient de l'histoire de la logique mathématique et du questionnement sur le fondement des mathématiques (le programme d'Hilbert). Ce questionnement part d'une analyse essentiellement algébrique, centrée sur l'arithmétique, et repousse la géométrie. Après une période de gestation, la logique mathématique naît véritablement en 1931, avec le théorème d'incomplétude de Gödel. À partir de ce moment, on assiste à une explosion de résultats et des domaines entiers se constituent : calculabilité, théorie de la preuve, théorie des modèles, etc. Dans un second temps, dans les années 1970, la géométrie a fait un grand retour en théorie de la démonstration, en particulier grâce à la logique linéaire du logicien français Jean-Yves Girard, qui introduit de nouvelles représentations des preuves utilisant des graphes. Ainsi, la logique mathématique, par la diversité de ses approches, a fini par modifier la notion même de fondement, et à retrouver, tout récemment, une problématique géométrique... que la physique n'avait jamais abandonnée.*

C'est ainsi que la logique mathématique et l'informatique en sont venues à analyser, elles aussi, les structures géométriques présentes dans les formes de la nature. Incontestablement nouveau, ce dialogue entre la logique et l'informatique d'une part et les sciences de la nature d'autre part dépasse la problématique classique – formaliste et logiciste – des fondements. En outre, ce processus s'est imposé par l'évolution interne de l'informatique : les ordinateurs sont distribués à la surface du globe et reliés dans des réseaux. Ils intègrent l'espace et le temps physiques, et forment ainsi des systèmes « concurrents » (qui doivent synchroniser leur calcul), lesquels demandent aussi une analyse géométrique de leurs processus.

Pour la Science : ***Dans l'ouvrage que vous venez de publier avec votre collègue physicien Francis Bailly,* Mathématiques et sciences de la nature, la singularité physique du vivant *(Hermann, Paris, 2006), vous posez un diagnostic sur l'histoire des mathématiques et de la physique au XX^e^ siècle : alors que la physique se géométrise, en particulier avec la relativité einsteinienne et la géométrie des systèmes dynamiques de Poincaré, les recherches sur les fondements des mathématiques s'algébrisent et se détournent de toute intuition géométrique. Pouvez-vous revenir sur cette situation de divorce au cœur des sciences exactes ?***

Giuseppe Longo : *Un tel divorce n'existait pas dans la physique et les mathématiques galiléo-newtoniennes. Il vient pour l'essentiel du grand bouleversement qu'ont introduit l'avènement des géométries non euclidiennes et le renouvellement riemannien du concept d'espace. D'un certain côté, le constat posé par Frege est exact : il y a comme un « délire » dans les mathématiques de la première moitié du XIX^e^ siècle, où l'on propose les nou-*

À gauche, les défenseurs du déterminisme prédictif, Pierre-Simon Laplace et David Hilbert. À droite, les représentants du déterminisme non prédictif, Henri Poincaré et Kurt Gödel. Au centre, Alan Turing qui, dans ses travaux, fit cohabiter les deux formes de déterminisme.



Déterminisme et imprédictilibité physique

Un pendule simple, c'est-à-dire une masse oscillant au bout d'une corde, est un objet physique déterministe *et* prédictible. *Il est mathématiquement déterminé par une équation (qui régit l'angle* θ *qu'il forme avec la verticale) et par deux paramètres (sa longueur* L *et sa masse* m*), dont un déduit la période d'oscillation. Si l'on néglige la friction pour de petites oscillations (de petits angles* θ*), l'équation prédit son évolution pour longtemps* (courbe rouge).

Un double pendule, qui consiste en un pendule attaché à un autre, est aussi un objet déterministe : *il est déterminé par deux équations. Cependant il est* imprédictible *: son mouvement chaotique rend impossible de prédire la trajectoire* (courbe rouge) *au-delà d'un très petit nombre d'oscillations. Une propriété apparaît quand on observe un double pendule physique : même si on le fait démarrer dans des conditions initiales très proches, celui-ci suivra bientôt une trajectoire* différente. *Comme il est physiquement impossible de reproduire exactement les mêmes conditions initiales, à cause des fluctuations (thermiques par exemple) ou des perturbations, on n'obtient jamais la même trajectoire. Le Système solaire, où toutes les planètes sont en interaction gravitationnelle avec le Soleil mais aussi entre elles, présente les mêmes propriétés d'imprédictibilité physique. Le chaos existe même dans la « grande horlogerie céleste »…*

Au contraire, une simulation informatique, c'est-à-dire un modèle calculatoire fonctionnant sur un ordinateur, machine à états discrets, permet de relancer la simulation exactement *sur les mêmes valeurs numériques. Le double pendule simulé numériquement, une fois relancé,* suivra exactement la même trajectoire, *ce qui est un phénomène* physiquement impossible. *L'imitation informatique est certes remarquable, mais elle présente un aspect qui est* étranger *à la physique des systèmes chaotiques : la possibilité de l'itération parfaite. Le fait est que les ordinateurs digitaux sont avant tout des machines à itérer : l'itération est présente dès les principes de base de la calculabilité (la récursion primitive) et rend possible l'installation et le fonctionnement d'un logiciel sur n'importe quelle machine. Sans itération, il n'y aurait pas d'informatique.*

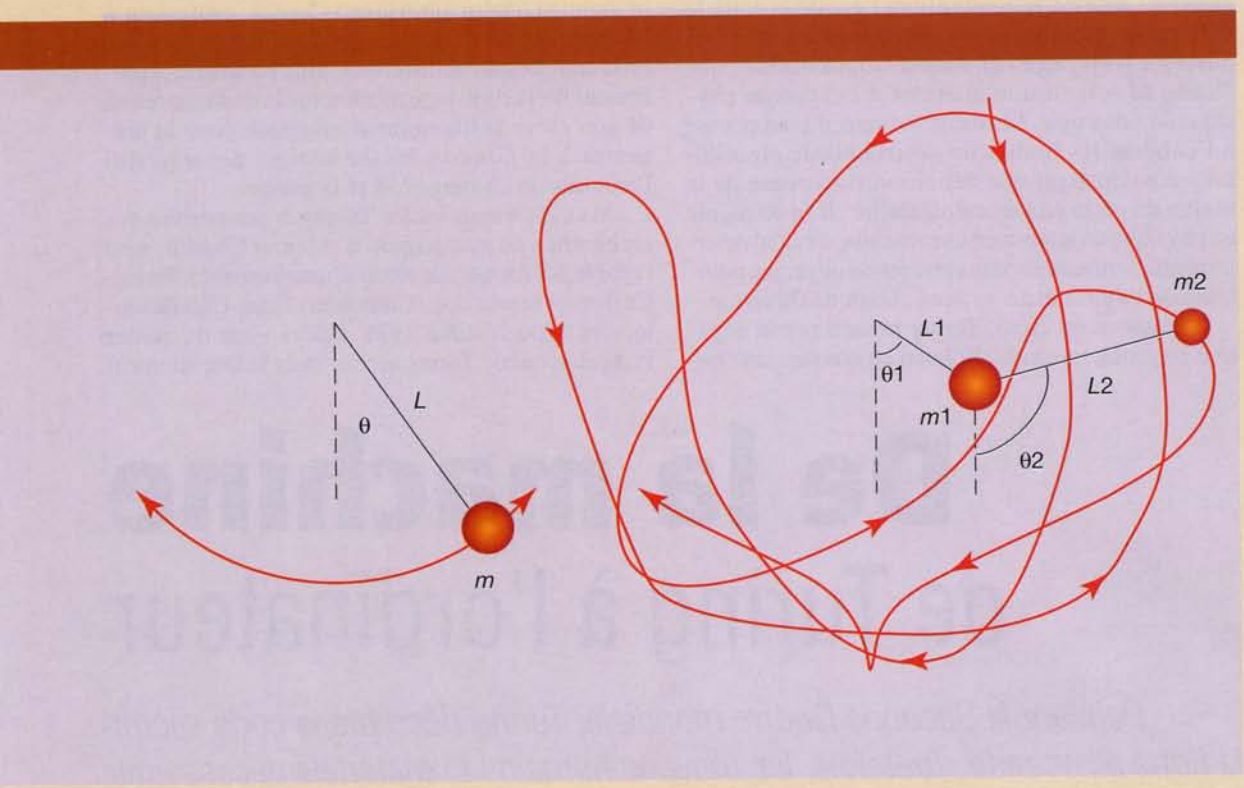
velles géométries. Il a fallu corriger ce « délire » par un retour à l'arithmétique, théorie logique par excellence, selon Frege, au cœur de la formalisation, pour Hilbert. Mais la formalisation est un moyen, pas une fin !

On assiste donc, au cours du XX^e siècle, à ce fait étrange du point de vue épistémologique : les cadres de pensée des mathématiques et de la physique, qui se fécondent mutuellement depuis la Renaissance, continuent à s'emprunter mutuellement des outils et surtout des principes d'intelligibilité, mais la logique mathématique, c'est-à-dire la perspective sur les fondements des mathématiques, devient étrangère à cette fécondation mutuelle. Cela a conduit à une perte d'intelligibilité, qu'il faut corriger aujourd'hui par un dialogue avec la physique et avec la biologie, sur les fondements même de la connaissance.

Pour la Science *:* ***Laplace et Hilbert poussent le déterminisme aussi loin que possible, le premier en physique et le deuxième sur les fondements des mathématiques. Poincaré et Gödel montrent les limitations internes de ce paradigme, l'un en physique et l'autre dans le cadre du fondement des mathématiques. Comment situez-vous le parcours de Turing par rapport aux démarches de ces quatre savants ?***

Giuseppe Longo : *Reprenons un instant l'analogie. Pour Laplace, la détermination mathématique implique la prédictibilité de l'évolution des systèmes physiques. Pour Hilbert, la formalisation en tant que détermination d'une théorie mathématique implique la décidabilité des énoncés formalisables. Poincaré démontre l'imprédictibilité de certains systèmes déterministes, dont le pendule double physique* (voir la figure ci-contre) *et même le Système solaire ! Gödel et Turing prouvent l'indécidabilité de certains systèmes formels, eux aussi très importants. Dans ce cadre, Turing, tout en étant logicien, possède une intuition de* physicien *qui le fait passer d'une branche de l'analogie à l'autre. Il est dans la lignée d'Hilbert quand il expose ce qu'il entend par déterminisme calculatoire, mais il est également dans celle de Gödel quand il conçoit l'indécidabilité du problème de l'arrêt dans le cadre du « déterminisme formel » hilbertien. Il se situe dans la lignée d'un Laplace quand il matérialise une machine déterministe, l'ordinateur séquentiel (dont l'évolution finie est prédictible, selon ses propres mots), mais il est tout autant dans celle de Poincaré quand il observe que le cerveau n'est pas une machine laplacienne et que le vivant, et plus précisément l'évolution physique de ses formes, pose des problèmes d'auto-organisation, qui doivent être abordés géométriquement.*

En particulier, Turing, dans les travaux qu'il mène à partir de 1948, se tourne vers la physique sans faire référence à la notion d'algorithme, qu'il a pourtant contribué à formu-



ler. Il travaille à ce qu'il appelle des « systèmes continus », par opposition à sa « machine à états discrets ». Il s'agit de systèmes dynamiques non linéaires, dont le matériel se déforme et qui est sujet à une « dérive exponentielle » (nous dirions aujourd'hui « sensible aux conditions initiales »). Ainsi, en à peine 20 ans, il réussit à pousser à la limite le déterminisme classique, dans le cadre du discret, de l'arithmétique formelle, et à porter son regard scientifique au-delà, dans les sciences de la nature, par son analyse des dynamiques continues. C'est d'une originalité dont on mesure encore mal les conséquences…

Pour la Science : ***Dans le titre même de votre livre, vous insistez sur la « singularité physique du vivant ». L'itinéraire de Turing, qui part d'une réflexion sur le calcul et en vient à des problèmes relevant de la biologie, vous paraît-il exemplaire de ce point de vue ?***

Giuseppe Longo : *Le parcours de Turing est une source capitale de réflexion épistémologique, comme je viens de le dire. Mais Francis Bailly et moi-même prenons acte des transformations contemporaines des sciences de la nature pour essayer de voir plus loin. En particulier, l'auto-organisation du vivant nécessite de repenser la notion d'espace et de temps physiques : par conséquent, c'est non seulement en physicien qu'il faut aborder l'espace et le temps de l'évolution des espèces et de l'ontogenèse, mais aussi par un regard* propre *porté sur les phénomènes du vivant, avec toutes les difficultés que cela comporte. D'autre part, par son étude de la morphogenèse, Turing a brillamment analysé des nombreux phénomènes qui se présentent à l'interface physique entre le vivant et son écosystème, et qui sont bien décrits par les théories physiques courantes. Mais cette analyse ne suffit pas. Il faut lui ajouter une théorisation* autonome *du biologique, avec ses propres observables et paramètres. C'est, à mon sens, l'une des grandes questions scientifiques du XXIe siècle.*

Le temps du vivant, par exemple, est loin d'être un épiphénomène du mouvement, voire un paramètre des processus, comme dans la plupart des théories physiques ; le temps biologique accompagne la mise en place de l'organisation phylogénétique et ontogénétique. C'est un « opérateur », tout comme l'énergie est un opérateur en physique quantique, constitutif des phénomènes (l'énergie est même l'observable clé de la mesure quantique). Le vivant organise son propre temps, *par ses horloges internes, ses rythmes propres. Le temps est, pour ainsi dire, l'observable principal de la biologie. L'intrication entre temps et processus du vivant mérite une théorisation propre, tout comme la physique quantique a théorisé la notion de « champ » comme corrélation entre espace, temps et énergie.*

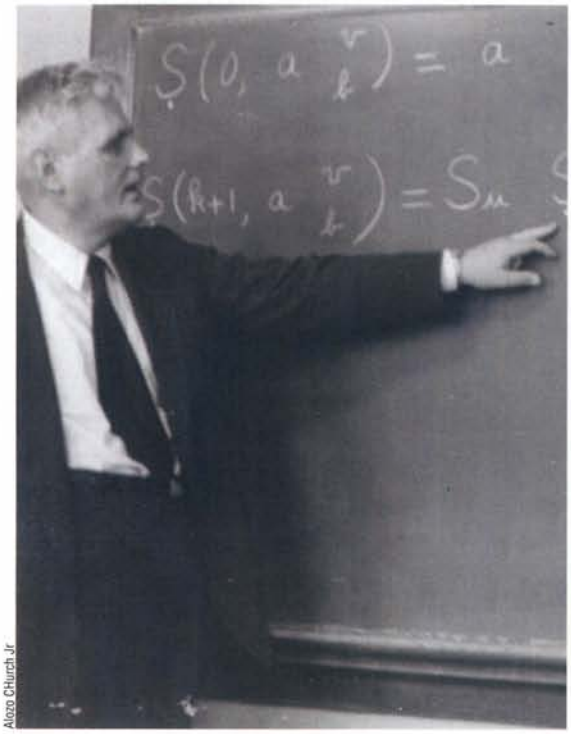
De la machine
de Turing à l'ordinateur

Pendant la Seconde Guerre mondiale, Turing décrypte le code secret de la flotte allemande. Toutefois, les moyens humains et matériels nécessaires sont considérables. Turing songe alors à matérialiser sa machine universelle.

Le problème de l'arrêt exhibé par Turing dans son article de 1936 a eu de profondes répercussions sur la cohérence du déterminisme classique : en démontrant que certains calculs ne pouvaient pas être déterminés à l'avance, il déséquilibra l'analogie entre le déterminisme calculatoire en mathématique et en physique *(voir page 78)*. Rien d'étonnant, alors, que Turing se soit ensuite intéressé à la branche physique de l'analogie. En mathématique, il avait poussé à l'extrême les limites du déterminisme algorithmique en donnant une définition rigoureuse de la notion de périmètre de calculabilité ; il fit de même en physique en imaginant une machine à calcul déterministe, *l'ordinateur*, dans un monde physique pourtant non déterministe au sens classique du terme.

Revenons en 1936. Turing répond par la négative au problème de la décision en ouvrant un nouveau champ théorique, la théorie de la calculabilité, grâce à sa notion de machine. Sa démonstration est en fait la *deuxième* réponse au problème de la décision, car quelques semaines avant sa parution, le logicien américain Alonzo Church, professeur à l'Université de Princeton, en a publié une par des biais complètement différents. Max Newman, le professeur de Turing, juge néanmoins la démonstration de son élève suffisamment originale pour la présenter à la *London Mathematical Society*, qui l'examine le 28 mai 1936 et la publie.

Max Newman incite Turing à poursuivre ses recherches en compagnie d'Alonzo Church, seul capable, à l'époque, de servir d'interlocuteur à Turing. Ce dernier rejoint donc Church aux États-Unis de septembre 1936 à juillet 1938. Gödel vient de quitter Princeton quand Turing arrive, mais le Département



Alozo CHurch Jr

de mathématiques, qui accueille de nombreux émigrés d'Allemagne et d'Europe centrale, reste un lieu riche en échanges intellectuels. Pendant son séjour américain, Turing rédige un *Philosophical Doctorate (PhD)* de logique mathématique sous la direction d'Alonzo Church tout en collaborant avec John von Neumann sur des questions de théorie des groupes. Alonzo Church mentionne pour la première fois le terme de « machine de Turing » dans son compte rendu de l'article de Turing qu'il rédige pour le *Journal of Symbolic Logic*. Au cours de son séjour, von Neumann offre à Turing de devenir son assistant à *l'Institute for Advanced Studies* de Princeton, mais Turing préfère retourner à Cambridge en Angleterre, dont l'ambiance intellectuelle et les relations sociales, plus libertaires que celles de Princeton, lui conviennent mieux.

Contourner le théorème de Gödel ?

Dans son *PhD*, Turing s'interroge sur les rapports entre sa théorie de la calculabilité et les conséquences du théorème de Gödel de 1931 en théorie de la preuve. Le théorème d'incomplétude de 1931 montre que tout système formel est incomplet, au sens où sa propre consistance ne peut être démontrée dans le système. Toutefois, il prouve aussi qu'il est possible d'enrichir n'importe quel système formel par un nouvel axiome qui inclut la consistance du système originel.

Appliquant les idées sur la calculabilité effective qu'il a développées dans son article de 1936, Turing construit alors une hiérarchie potentiellement infinie de systèmes formels, indexée sur les nombres et en étudie les propriétés. Il explique ainsi sa démarche dans un article intitulé *Systems of Logic based on Ordinals*, publié en 1939 sur le sujet :

Le but que visait l'introduction des logiques ordinales était d'éviter autant que possible les effets du théorème de Gödel. Convenablement interprétée, une de ses conséquences revient à ce qu'il soit impossible de construire [...] *un système logique complet. Nous sommes parvenus, quoi qu'il en soit, à construire, pour un système donné, un système plus complet en y ajoutant comme axiomes des formules qui, intuitivement, apparaissent comme correctes, mais dont le théorème de Gödel montre qu'elles sont indémontrables dans le système originel ; nous avons construit, à partir de là, un système plus complet par une répétition du processus, et ainsi de suite.*

On reconnaît là le style particulier de Turing, consistant à repousser à l'extrême les limites de la construction mécanisable. Grâce à cette hiérarchie, Turing cherche non seulement à contourner le théorème de Gödel, mais à montrer comment le concept d'indécidabilité peut lui-même être relativisé : au lieu d'y voir un absolu indépassable, il en fait un concept *relatif à chaque système logique de la hiérarchie*.

Dans le même article de 1939, Turing fait un parallèle entre la partie des mathématiques qui ne peut pas être formalisée (d'après le théorème de Gödel) et l'intuition. Il distingue deux facultés de la pensée mathématique, qu'il dénomme « intuition » et « ingéniosité ». La première, de nature non constructive, c'est-à-dire qui ne peut pas être cernée par un

www.arikah.com/encyclopedia/Princeton_University

Le logicien américain Alonzo Church (1903-1995, à gauche) et le mathématicien américain d'origine hongroise John von Neumann (1903-1957, à droite). Ci-contre, la tour Cleveland de l'Université de Princeton, sur une aquarelle de David Liao (2006).

raisonnement ayant un nombre fini d'étapes, n'a pas de contrepartie formelle, contrairement à la seconde, de par sa nature constructive. Gödel a démontré qu'il n'est pas possible de supprimer l'intuition (sinon tout système formel contenant l'arithmétique serait complet) ; Turing cherche donc à savoir ce qui se produit quand on réduit son rôle autant que possible, tout en augmentant *indéfiniment* celui de l'ingéniosité :

Le raisonnement mathématique peut être considéré de façon schématique comme l'exercice d'une combinaison de facultés que nous pouvons appeler l'intuition et l'ingéniosité. L'activité de l'intuition consiste à produire des jugements spontanés qui ne sont pas le résultat de chaînes conscientes de raisonnement. [...]. *L'exercice de l'ingéniosité en mathématique consiste à aider l'intuition par des arrangements adéquats de propositions et peut-être par des figures géométriques ou des dessins. Dans les temps pré-gödeliens, certains pensaient que* [...] *la nécessité d'un recours à l'intuition pourrait être entièrement éliminée.* [...] *Nous avons essayé de voir jusqu'où il était possible d'éliminer l'intuition. Nous ne nous préoccupons pas de savoir quelle quantité d'ingéniosité est requise et nous faisons donc l'hypothèse qu'elle est disponible en quantité illimitée.*

Ce point de vue était déjà celui de l'article de 1936 : le ruban de la machine de Turing *(voir page 73)* d'une longueur *indéfinie* jouait le même rôle que celui de l'ingéniosité, dont la quantité est, elle aussi, indéfinie. L'intuition, bien que de nature non mécanique, peut être assimilée à une machine, que Turing appelle « machine à oracle ». Cette machine est susceptible de prendre immédiatement des décisions pour des problèmes qu'une machine de Turing « normale », c'est-à-dire ayant à produire *effectivement* un calcul, ne peut résoudre, comme le problème de l'arrêt (la machine *A* du problème de l'arrêt, qui prédit si toute machine de Turing va s'arrêter ou non, est une machine à oracle, *voir page 78*). Il est alors possible de comparer, pour des problèmes ouverts et des programmes en construction, le nombre de pas de calcul qui seraient nécessaires à leur achèvement et le nombre de fois où la machine à oracle doit intervenir pour que le programme aboutisse. On peut alors classer les problèmes selon leur degré de complexité : plus la machine à oracle intervient, plus le programme est complexe.

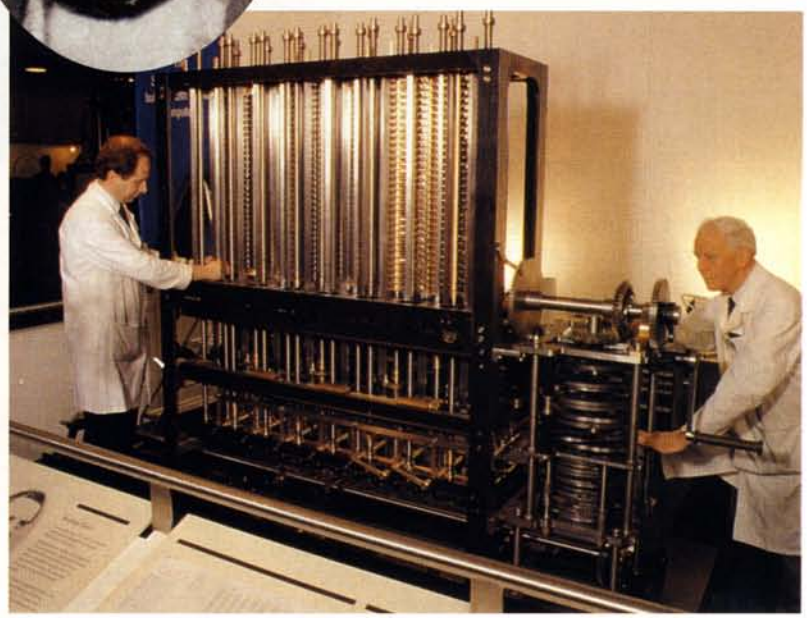
Cette recherche initiée par Turing ne prendra son essor qu'à la fin des années 1960, avec la théorie de la complexité algorithmique.

Premières machines à calculer

La théorie de la calculabilité par machine de Turing trouve un autre domaine d'application : la cryptologie. Dès 1937, Turing s'intéresse de près à la science du codage et du décodage des messages secrets : il s'est rendu compte qu'une méthode de décryptage est un *algorithme qui pourrait être exécuté par machine de Turing*. Ainsi, dès cette époque, il cherche à *mécaniser la cryptologie*. Cette recherche l'amène à s'interroger sur la construction effective de machines à crypter et leur rapport aux machines à calculer – *bien réelles* cette fois, et pas seulement sur leur structure logique, comme dans son article de 1936.

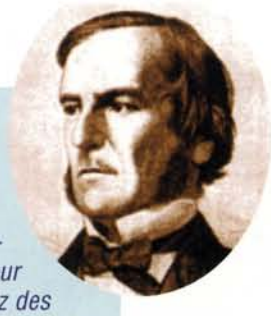
La tradition des machines à calculer est ancienne : elle remonte au vaste mouvement de grammatisation de l'âge classique. En 1641, Pascal inventa une machine à calculer effectuant des additions et des soustractions, suivi de Leibniz et bien d'autres après eux. Plus près de Turing, le mathématicien et ingénieur Charles Babbage (1791-1871) passa sa vie à construire, sans succès, sa *machine à différences*, ou *machine analytique*, qui nous apparaît aujourd'hui comme un ordinateur mécanique *(voir la figure ci-contre)*.

Turing connaît l'œuvre de Babbage et ses recherches se placent dans cette filiation anglaise. Contrairement à Babbage cependant, Turing se rend compte que les opérations arithmétiques ordinaires peuvent être traduites dans le cadre de la logique booléenne *(voir l'encadré page ci-contre)* ; or il est possible, depuis peu, d'exprimer les opérations de cette logique sous forme de circuits électriques nommés *circuits booléens*. Grâce au soutien d'un étudiant en physique qui lui ouvre l'atelier des physiciens au mépris des règles en vigueur à Princeton, Turing



La* machine à différences *construite en 1991 par deux ingénieurs anglais, Barrie Holloway* (à gauche) *et Reg Crick* (à droite), *d'après les plans conçus par Charles Babbage (1791-1871,* en médaillon) *au XIXe siècle. La construction de ce calculateur mécanique automatique fit rétrospectivement de ce mathématicien un pionnier de l'informatique.

La logique booléenne



Au XIXe siècle, le logicien britannique George Boole (1815-1864, en médaillon*) remarqua l'analogie formelle qui existait entre les conjonctions logiques «et» et «ou», et les opérations arithmétiques élémentaires + et × de l'algèbre. Développant cette analogie, il créa un système algébrique à deux valeurs numériques (0 et 1) transposable à la logique : il suffisait pour cela d'associer aux deux valeurs 0 et 1 les valeurs «vrai» et «faux» de la logique classique. Le système formel ainsi construit par Boole est connu sous le nom d'algèbre binaire, car les opérations sont limitées aux nombres 0 et 1. En utilisant des variables à la place du sujet des expressions logiques, l'algèbre de Boole libéra ces dernières des ambiguïtés du langage courant. L'algèbre logique ainsi construite par Boole lui permet de formaliser les raisonnements tenus en langage courant et d'obtenir ainsi plus rapidement les résultats. Il utilise pour cela des tables de vérité telle celle représentée en haut de la page 66, qui récapitulent, pour les relations logiques étudiées, toutes les situations de vérité possibles à partir de plusieurs propositions successivement vraies ou fausses.*
En 1937, Turing, qui cherche un moyen de construire effectivement sa machine de papier, s'inspire de l'analogie entre opérations arithmétiques élémentaires et conjonctions logiques à l'origine de la logique booléenne. Il sait en outre que, depuis peu, des physiciens tels que l'ingénieur américain George Stibitz des Laboratoires Bell *ont réussi à effectuer des opérations booléennes (addition, soustraction booléennes) à l'aide de circuits électroniques, nommés aujourd'hui circuits booléens. Son idée est, déjà, de transformer tout programme de la machine de Turing en une succession de calculs binaires effectuables par des circuits booléens, principe toujours à la base de l'informatique moderne.*

construit son propre multiplicateur électrique booléen, qui finit par fonctionner.

Sa réussite professionnelle n'estompe cependant pas le mal de vivre qui le mine depuis la mort de Christopher Morcom. Turing traverse une phase de dépression qui l'amène à écrire à un de ses amis en Angleterre qu'il a songé à se suicider avec un mécanisme incluant une pomme et du fil électrique. Le jeune homme dépasse cette crise et revient en Angleterre en juillet 1938, rapportant avec lui son multiplicateur électrique. La guerre donne alors tout son poids à ce qui n'est encore qu'un intérêt privé pour la cryptologie et les calculs effectifs, et infléchit les recherches de Turing vers l'aspect physique du déterminisme algorithmique.

Enigma

À son retour des États-Unis, Turing est recruté, ainsi qu'une soixantaine d'autres personnes, par le Service britannique du chiffre, le *GC&CS (Government Code and Cypher School)*, sans doute par le réseau des anciens de son collège de Cambridge, *King's College*. Confronté à l'expansionnisme nazi, le gouvernement britannique s'est lancé avec retard dans l'écoute aussi systématique que possible des émissions radio de l'armée allemande, des services de sécurité et des *Schutzstaffel* (les SS). Les méthodes employées par les Britanniques datent de la Première Guerre mondiale et ne donnent rien face à la machine à crypter les messages utilisée par les Allemands, *Enigma*.

Enigma est à l'origine le nom d'une machine de cryptage/décryptage inventée en 1919 par l'ingénieur hollandais Hugo Alexander. Ses plans furent immédiatement repris par l'ingénieur allemand Arthur Scherbius qui en fit un modèle commercial présenté au public en 1923. Dans les années 1930, les services secrets anglais et allemands améliorèrent chacun de leur côté ce modèle commercial, les premiers nommant leur nouvelle machine *Typex* et les seconds conservant le nom latin *Enigma* (100 000 exemplaires en furent construits pendant la Seconde Guerre mondiale).

L'*Enigma* à usage commercial ressemble à une machine à écrire, mais il s'agit d'une machine mécanique et électrique ; elle est disposée dans une caisse de bois et pèse une douzaine de kilos. Le cryptage est basé sur la permutation des lettres à travers un réseau électrique ; il est transmis par un clavier marqué des lettres de l'alphabet et servant de connecteur. La permutation est conçue de telle sorte qu'une lettre est encodée différemment chaque fois qu'on appuie sur le clavier. Cet encryptage évolutif est réalisé de la façon suivante : le réseau électrique de la machine est composé de 3 rotors (plus un rotor «réflecteur») dont chacun possède 26 positions marquées par des encoches correspondant aux lettres de l'alphabet. Lorsqu'une lettre est tapée, le premier rotor avance d'un cran ; le rotor suivant se meut à son tour d'un cran quand le premier rotor a fait un tour complet, et le troisième rotor procède de même vis-à-vis du deuxième. Les 26 lettres sont fixées dans l'ordre alphabétique sur chaque rotor, par un anneau que l'on peut faire tourner manuellement, donc aléatoirement ; cet anneau fait apparaître une lettre de l'alphabet par une fenêtre *(voir l'encadré page 86)*.

Par rapport à ce modèle commercial, la complexification du modèle de l'armée allemande porte sur deux points : l'ajout de rotors supplémentaires et celui d'un tableau de connexions qui permute deux

Astuces et faiblesses de la machine Enigma décryptée par les Polonais

Lorsqu'en 1932, les jeunes mathématiciens polonais Marian Rejewski, Jerzy Rozycki et Henryk Zygalski intègrent le Bureau du chiffre polonais pour décrypter la machine Enigma *perfectionnée par les Allemands, le Bureau ne dispose que d'une maigre documentation allemande fournie par le Service de renseignements français. Cette documentation donne des informations sur la structure générale de la machine et sur les procédures d'emploi ; elle comporte aussi des tableaux de clés permettant aux opérateurs de mettre la machine « à la clé » (arrangement des trois rotors, position de base des rotors, arrangement des six câbles du tableau de connexions). Avec cette documentation, l'équipe polonaise comprend mieux le fonctionnement de la machine :*

– les trois rotors sont arrangés dans un ordre connu de l'opérateur et du destinataire (six possibilités) pour une période de temps donnée (trois mois jusqu'à fin 1935, tous les mois jusqu'à septembre 1936 et tous les jours à partir d'octobre 1936) ;

– les trois rotors doivent être initialisés dans une position de base donnée pour le chiffrement d'une clé de trois lettres choisie par l'opérateur, cette clé servant à chiffrer effectivement le message ; en outre, cette clé de message est répétée et chiffrée deux fois en tête du cryptogramme pour détecter les éventuelles erreurs de transmission ;

– le tableau de connexions est utilisé avec six câbles, ce qui permet la substitution réciproque de 12 lettres.

La mise à la clé consistait à brancher les six câbles du tableau de connexions, à placer les trois rotors dans l'ordre secret indiqué sur le tableau de clé et à placer les trois rotors selon la position de base. Pour chiffrer, l'opérateur choisit au hasard les trois lettres d'une clé de message (par exemple QWE*) qu'il saisit deux fois* (QWEQWE). *Il note le résultat chiffré (par exemple* WTRBAZ*) et repositionne les trois rotors sur* QWE ; *il frappe ensuite le texte du message et note le résultat du chiffrement (par exemple* JOZNXPSU…*), il transmet alors la séquence* WTRBAZ JOZNXPSU… *En réception, l'opérateur ayant préalablement mis la machine à la clé, déchiffre* WTRBAZ, *ce qui lui donne* QWEQWE *(la machine est réversible). Il initialise ensuite les trois rotors avec* QWE *et saisit le chiffré* JOZNXPSU… *ce qui lui redonne le texte clair.*

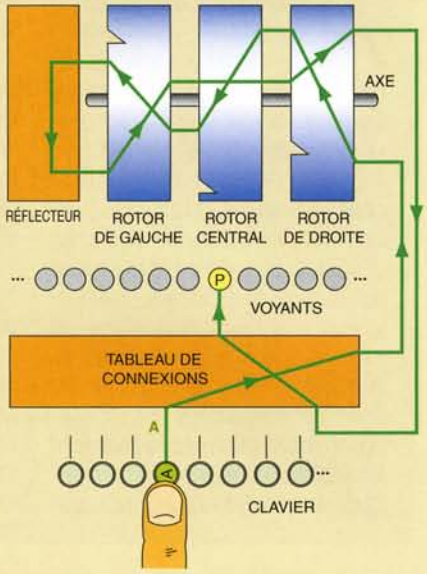
Rejewski comprend vite que le doublement de la clé de message est une faiblesse qu'il faut exploiter. Il se rend compte que la clé de message chiffrée deux fois possède la propriété suivante : si on note k_1, k_2, k_3 *les trois lettres de la clé de message en clair,* $c_1,\ldots, c_6$ *les six lettres résultat du chiffrement de la clé de message répétée, et* D_i *le déchiffrement de la substitution de la i-ème lettre réalisée par la machine, alors on a le système (S) :*

$$(S)\quad \begin{aligned} D_1(c_1) &= D_4(c_4) = k_1 \\ D_2(c_2) &= D_5(c_5) = k_2 \\ D_3(c_3) &= D_6(c_6) = k_3 \end{aligned}$$

Le système (S) prend tout son intérêt lorsque les D_i *résultent de l'action du seul rotor de droite (les deux autres rotors restant fixes). Cela est vrai dans 21 cas sur 26, c'est-à-dire quand la position de départ du rotor de droite est à plus de cinq lettres de l'ergot d'activation du rotor central. Dans ce cas, les deux autres rotors restent fixes. D'autres équations ont également pu être écrites qui, jointes à l'observation des clés de messages chiffrées, ont permis à Rejewski de calculer le câblage du rotor de droite.*

Grâce aux permutations régulières des rotors et aux nombreux messages interceptés, Rejewski reconstitue le câblage des trois rotors : en décembre 1932, moins de quatre mois après ses débuts sur la machine, Enigma *est entièrement reconstituée. Dès février 1933, la fabrication des répliques d'*Enigma *commence, 15 en juin 1933 et jusqu'à 70 en août 1939. Les répliques* Enigma *disponibles, il reste à retrouver régulièrement les clés. Rejewski remarque que de nombreux messages du même jour (même configuration de la machine) font apparaître des chaînes cycliques dans le chiffrement des clés de message. Par exemple, avec les clés de message chiffrées ci-après :*

SCHÉMA DE PRINCIPE DE LA MACHINE *ENIGMA*



Fonctionnement de l'*Enigma*.** ***Après avoir mis la machine à la clé, l'opérateur frappe le texte clair lettre par lettre sur le clavier. Chaque lettre, traduite par un courant sur un circuit électrique, subit une première substitution par le tableau de connexions. Le signal est ensuite transmis au rotor de droite (qui avance d'une position, soit 1/26e de tour), puis au rotor central (qui avance d'une position après les 26 déplacements du rotor de droite), enfin au rotor de gauche (qui tourne d'une position après les 26 déplacements du rotor central) ; l'ensemble des trois rotors réalise ainsi 3 autres substitutions, variables à chaque lettre selon leur progression. Le signal est ensuite transmis au réflecteur qui le réinjecte sur une autre broche du rotor de gauche, puis central et enfin de droite réalisant encore 3 substitutions, inverses des 3 substitutions précédentes. Pour finir, le signal est réinjecté dans le tableau de connexions et allume un voyant désignant une des 26 lettres de l'alphabet, résultat du chiffrement qui doit être noté par l'opérateur avant de chiffrer les lettres suivantes du message.

message 1 : *WTRBAZ*, message 4 : *BOSWAO [remarquez le cycle* WB, BW *noté* (WB)*]*

message 2 : *FOCPIT*, message 5 : *PYXFDE [remarquez le cycle* FP, PF *noté* (FP)*]*

message 3 : *AMWXUP*, message 6 : *XNDTUQ*, message 7 : *TWMARO [remarquez le cycle* AX, XT, TA *noté* (AXT)*]*

La chaîne <(WB),(FP),(AXT)*> correspond à la première lettre de la clé de message. Rejewski démontre que la structure de ces chaînes, observables sur les trois lettres de la clé de message, est uni-*

quement fonction de l'arrangement des trois rotors et de la position de base, soit $6 \times 26^3 = 105\,546$ possibilités. L'ensemble de l'équipe, grâce à leurs répliques Enigma, *essaye les 105 546 arrangements et positions de départ possibles et note dans un catalogue toutes les chaînes cycliques associées. Ce travail dure un an et le catalogue permet, jusqu'en 1939, environ 100 000 décryptages. L'histoire se complique lorsque, en décembre 1938, les Allemands mettent en service deux nouveaux rotors...*

D'après A. Cattieuw et P. Hébrard, De la mécanique à l'ordinateur, *Dossier Pour la Science* La cryptographie, *n°36, pp. 18-25, 2002.*

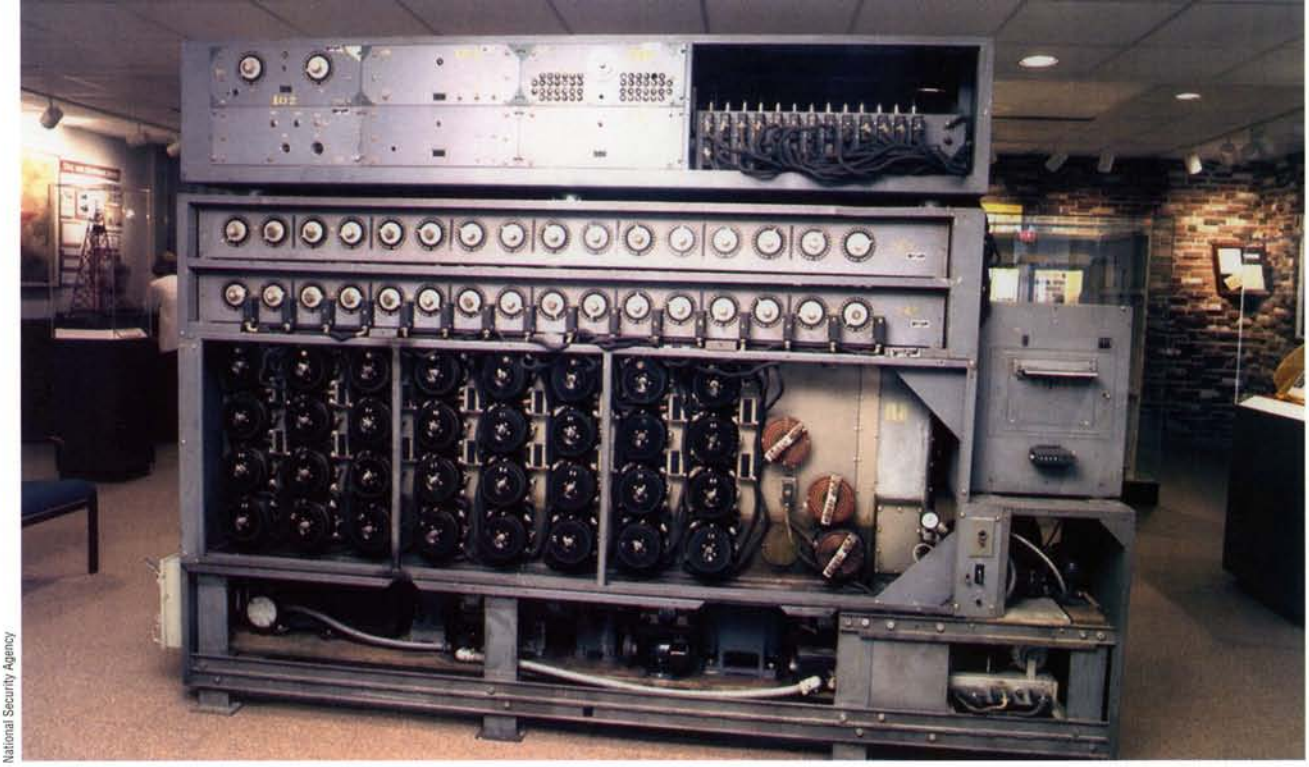
La machine* Enigma *utilisée par l'armée allemande pendant la Seconde Guerre mondiale pour crypter ses messages.

lettres avant et après les rotors. Pour crypter ou décrypter un message, il fallait donc connaître trois choses : les permutations à réaliser manuellement sur le tableau des permutations, l'encoche de départ propre à chaque rotor, et la lettre à faire apparaître sur l'anneau de chaque rotor. Ces trois informations étaient distribuées à toutes les unités sous la forme d'un carnet mensuel de codes qui devait être consulté quotidiennement pour adapter *Enigma* au code du jour.

En outre, afin d'éviter tout décryptage par un tiers, vu la grande quantité de messages envoyés avec le même code, l'armée allemande utilisait une clé particulière pour chaque message, qui consistait à définir le réglage de l'anneau sur chaque rotor : l'armée réglait sa machine selon le code du jour, y compris l'anneau de chaque rotor, puis choisissait au hasard une nouvelle position de ces anneaux, qu'elle cryptait selon le code du jour. Pour éviter les risques d'erreur dus aux interférences radio, elle envoyait cette position deux fois de façon consécutive en début de message ; la machine encryptant toujours une lettre différemment de la fois précédente, la clé, composée d'autant de lettres que de rotors, était transmise sous la forme de deux codes différents *(voir l'encadré ci-contre)*. Le nombre de permutations possibles se comptant en milliards, le haut commandement allemand était persuadé que sa machine était inviolable et l'utilisa massivement jusqu'en automne 1941.

L'exploit de Rejewski

Enigma fut adoptée par l'armée allemande dès 1929. À partir de ce moment, les services de renseignements des grandes puissances ne décryptèrent plus aucun message allemand et la machine acquit une réputation d'inviolabilité. Pourtant, une équipe des services de renseignements en Europe ne désarma pas : l'équipe polonaise. Dès 1932, les Polonais prirent conscience du danger potentiel que représentait pour eux la montée du nazisme en Allemagne, puis son arrivée au pouvoir en 1933. Forts d'une brillante tradition logico-mathématique – celle des logiciens et philosophes Jan Łukasiewicz (1878-1956), Stanisław Leśniewski (1886-1939) et Alfred Tarski (1902-1983) –, les Polonais constituèrent une équipe de cryptologie composée d'une vingtaine de mathématiciens germanophones spécialistes des permutations en théorie des groupes, et d'ingénieurs capables de construire des machines à décrypter. Le contre-espionnage français ayant obtenu les manuels d'utilisation d'*Enigma* et les ayant fait passer aux Polonais faute de pouvoir en tirer profit, ceux-ci construisirent une *Enigma* militaire à partir d'une *Enigma* commerciale, en reconstituant en particulier le câblage des fils électriques sur les rotors et le réflecteur. À l'aide de cette réplique de l'*Enigma* militaire, un jeune mathématicien de 27 ans, Marian Rejewski divisa le problème du décryptage en deux : celui du brouillage effectué par le tableau de



National Security Agency

Une des 121 **Bombes** *construites entre 1943 et 1945 par l'US Navy pour décrypter massivement les messages codés avec la nouvelle version d'*Enigma *utilisée par les Allemands à partir de 1940.*

Polish Philosophy Page

Quatre logiciens qui marquèrent les mathématiques polonaises : de haut en bas et de gauche à droite, Jan Łukasiewicz (1878-1956), Stanisław Leśniewski (1886-1939), Alfred Tarski (1902-1983) et Marian Rejewski (1905-1980). Ce dernier décrypta le code de la machine **Enigma** *utilisée par les Allemands jusqu'en 1938.*

permutations et celui du brouillage effectué par les rotors. Dans les deux cas, après un travail acharné, il cassa le code d'*Enigma*.

Dans le cas du brouillage effectué par le tableau de permutations – le plus simple –, la méthode consiste à découvrir, dans le désordre que présentent les lettres une fois « décodées » au niveau des rotors, un bout de « mot » possédant seulement quelques lettres permutées (en effet, certaines lettres sont inchangées par le tableau de connexion car il ne compte que six câbles, donc six permutations possibles de douze lettres). Ces lettres donnent un aperçu de la configuration partielle de la connexion opérée manuellement dans le tableau de connexion. Rejewski recomposa ainsi les connexions journalières du tableau.

Dans le cas beaucoup plus complexe du brouillage effectué par les trois rotors, une même lettre du message en clair se trouve cryptée par deux lettres différentes séparées par trois crans de rotor : en effet, la clé de chaque message, constituée de trois lettres, est répétée deux fois, formant les six premières lettres du message. Il devient alors possible, avec un nombre suffisant de messages, d'établir un lien entre les lettres en clair et les lettres cryptées. Sachant que les lettres en clair ne sont jamais cryptées par elles-mêmes (un *A* n'est jamais crypté par un *A*), Rejewski eut l'idée de suivre le cycle de réapparition d'une

lettre en se basant sur le cycle de permutations qu'elle subit dans un message, cycle inhérent au fonctionnement de la machine *Enigma* : par exemple, le cycle de *A* qui se transforme en *D*, puis en *B*, puis en *L*, puis de nouveau en *A*. Ce cycle des lettres est plus ou moins long, mais on peut toujours le suivre, car il correspond à la disposition des crans des rotors, indépendamment du brouillage ultérieur effectué par le tableau de permutations. En dressant pendant un an le tableau général de ces cycles, Rejewski entre en possession de tous les cycles utilisés par *Enigma* et donc de toutes les positions possibles des crans des rotors.

Toutefois, à partir du 15 septembre 1938, le cran de départ assigné à chaque rotor fut laissé à la discrétion de l'encodeur, qui devait donc communiquer son choix en début de message : le répertoire mis au point par Rejewski devint inutilisable. Rejewski trouva néanmoins une nouvelle méthode consistant à découvrir si, dans la clé du message répétée deux fois, une *même* lettre ne se retrouvait pas à trois crans de rotors successifs ; si c'était le cas, la lettre était appelée, sans raison apparente, « femelle » (en polonais « samiczka »), et indiquait une position relative des crans des rotors les uns par rapport aux autres. Par exemple, si l'on possède suffisamment de messages cryptés, il peut se produire qu'un jour donné, trois clés auront l'apparence suivante : *WAH WIK* ; *RAW KTW* ; *DWJ MWR*, où le *W* se répète, indiquant par là l'ordre relatif des rotors par rapport aux autres.

En comparant de nombreux messages, Rejewski mit au jour un stock de lettres femelles et, à partir de novembre 1938, à l'aide de machines appelées *Bombes*, il retrouva la position exacte des crans des rotors : afin d'accélérer la recherche, il monta en série six *Bombes*, chacune étant une réplique d'*Enigma* configurée avec un arrangement possible des trois rotors, et détermina alors les positions des crans des rotors qui produisaient les lettres dites « femelles ». Ainsi, ce fut à l'occasion d'une recherche centrée sur ces doublons « femelles » que la mécanisation du renseignement entra dans son ère moderne.

Jusqu'en décembre 1938, le système mécanisé mis au point par Rejewski fonctionna, mais l'addition par les Allemands de deux rotors supplémentaires rendit alors son travail partiellement obsolète, vu le nombre de *Bombes* à sa disposition. Le 24 juillet 1939, Les Polonais convoquèrent Français et Britanniques à une réunion secrète à Varsovie au sujet d'*Enigma*. Ils dévoilèrent à leurs alliés stupéfaits leur succès et leur donnèrent deux répliques de *l'Enigma* militaire ainsi que les plans des *Bombes* permettant le décodage mécanisé, tout en leur annonçant qu'ils n'avaient pas les moyens matériels de décrypter la nouvelle version d'*Enigma*. Les Alliés transférèrent le matériel polonais par la valise diplomatique. Cinq semaines plus tard, le 1er septembre 1939, l'armée allemande envahissait la Pologne, déclenchant la Seconde Guerre mondiale.

Les probabilités du brouillage

Le 3 septembre 1939, soit le lendemain du jour où la Grande-Bretagne et la France déclarent la guerre à l'Allemagne, Turing prend son poste au Service du chiffre britannique, qui a déménagé dans de nouveaux locaux à l'extérieur de Londres, au manoir de *Bletchley Park*, de peur d'essuyer les bombardements de l'aviation allemande. Au début de la guerre, 200 personnes y cohabitent (ils seront 7 000 à la fin). Turing est le responsable de la « hutte n° 8 », un préfabriqué de bois dans lequel quatre personnes (dont Turing) sont chargées du déchiffrement de l'*Enigma* navale, celle qui code les messages envoyés à la marine allemande. Turing reste assigné à cette tâche jusqu'au voyage secret qu'il effectue aux États-Unis pour le compte du service du chiffre, du 7 novembre 1942 au 31 mars 1943.

La « Hutte n° 8 » du Service du chiffre britannique à Bletchley Park, *dont Turing était responsable pendant la guerre* (ci-dessous). *À gauche, la maison de* Bletchley Park *où habitait Turing.*

Jack Harper

Jack Harper

*Des femmes travaillant au décryptement d'*Enigma *dans une hutte de* Bletchley Park.

Il s'agit de mettre au plus vite en pratique les méthodes et les machines conçues par les Polonais, mais sur une grande échelle, le *GC&CS* disposant de plus de moyens. Pendant la durée de la « drôle de guerre » (3 septembre 1939-10 mai 1940), cette stratégie fonctionne et les messages allemands sont souvent lus au jour le jour, ce qui permet de connaître les positions des unités de l'armée allemande et leurs plans d'attaque, jusqu'à l'opération de Narvik en Norvège (avril 1940). L'opération est un échec allié (en partie dû à la très mauvaise qualité du renseignement naval britannique, jaloux de son indépendance par rapport au Service du chiffre), mais elle permet au moins de récupérer sur un patrouilleur allemand une *Enigma* militaire, malheureusement sans son carnet de code, détruit par négligence pendant l'abordage.

Ainsi, la première stratégie du Service du chiffre britannique consiste à multiplier les *Bombes* et à utiliser une *Bombe* par lettre à décoder dans un mot. Par exemple, pour un mot de 7 lettres, on utilise 7 *Bombes* (chacune représentée par 3 nombres de 1 à 26 se référant aux crans des trois rotors). Cela accélère d'autant le repérage de la position des crans des rotors, mais ne règle pas la question du brouillage effectué par le tableau de connexions, avant et après le passage à travers le câblage des rotors. Turing conçoit alors de nouvelles stratégies qui, contrairement à celles de Rejewski, font appel au calcul des probabilités, sa spécialité mathématique d'origine. Ce choix est guidé par deux raisons.

L'analyse séquentielle

Pendant la Seconde Guerre mondiale, les statisticiens A. Wald et G.A Barnard ont indépendamment mis au point une méthode pour analyser la qualité de biens manufacturés, dont voici le principe. Pour décider si un lot de pièces est de bonne qualité ou pas sans avoir à passer en revue toutes les pièces, ce qui reviendrait trop cher, on opère un test où l'on distingue deux hypothèses : l'une *H* (lot de pièces défectueuses) et l'autre non-*H* (lot de pièces non défectueuses) et l'on cherche à choisir entre les deux hypothèses. On teste toutes les pièces une par une et on recalcule à chaque fois la qualité du lot, sans fixer à l'avance la taille de l'échantillon soumis au test de qualité. Après chaque observation, trois actions sont possibles : agir comme si *H* était vraie ; agir comme si non-*H* était vraie ; ordonner un autre test. Il s'agit de continuer à faire le test jusqu'à un seuil qui optimise le nombre de tests. La difficulté consiste à définir une règle d'arrêt pour le test de qualité. On voit qu'il s'agit là d'un problème analogue à celui de la décision.
Turing utilisera une méthode semblable pour coupler les problèmes de brouillage d'*Enigma* dus aux rotors et au tableau de connexions. La sélection d'un ensemble de biens dont on suppose qu'ils ont la même qualité peut être assimilée à l'acceptation d'une hypothèse sur la signification d'un mot dans un message crypté.

Tout d'abord, Turing considère trop risqué de se baser exclusivement sur la répétition du codage en début de message pour effectuer le décryptage. Si, comme il est à terme probable, les Allemands changent leur système, il n'y aura plus aucune méthode pour décrypter les messages. Il faut donc trouver une autre stratégie, indépendante de la répétition du code en début de message. Turing invente une méthode de décodage qui se fonde sur l'étude des mots probables (appelés *cribs*) et laisse de côté la clé répétée. Il remarque en effet qu'il est possible de tirer parti de la probabilité d'existence d'un mot dans un message : par exemple, le Service du chiffre a noté que toutes les unités de l'armée allemande reçoivent par radio le matin, à partir de 6 heures, un bulletin météo chiffré par *Enigma*, dans lequel le vocabulaire allemand de la météorologie est évidemment employé.

Turing développe alors un calcul reposant sur la notion de « poids d'évidence », qu'il applique à la probabilité d'un mot. L'unité de mesure de ce « poids d'évidence », dénommée *ban* (divisible en « déciban », sur le modèle des décibels) par Turing, en référence aux feuilles cartonnées de la ville de Banbury sur lesquelles il fait ses calculs, lui permet d'effectuer un calcul de probabilités sur l'ordre des rotors. À la même époque, pour résoudre des problèmes physiques de transmission du signal, le mathématicien américain Claude Shannon définit la notion d'information comme un événement possible parmi un ensemble d'événements, en distinguant, lui aussi, des événements plus ou moins probables. C'est le début de la *théorie de l'information*, dont Turing apparaît, avec le recul, comme l'un des pionniers.

La seconde raison de son choix est que les probabilités lui permettent d'articuler le problème du brouillage effectué par les rotors et celui du brouillage effectué par le tableau de connexions en distinguant, contrairement à Rejewski, le brouillage par le tableau de connexions *avant* et *après* le brouillage effectué par les rotors. Rejewski a traité de façon assez intuitive le problème du brouillage effectué par le tableau de connexions en tentant de repérer des bouts de mots en sortie, dans l'espoir de trouver des indices sur la façon dont deux lettres ont été appariées. Turing découvre qu'il existe un moyen de coupler les deux problèmes de brouillage, sans pour autant les confondre. Le type de recherche entrepris par Turing s'appelle, en statistique, l'*analyse séquentielle*. On crédite de son invention le statisticien américain A. Wald et le statisticien britannique G. A. Barnard, qui a fait, comme Turing, un *Ph. D.* de logique sous la direction d'A. Church à Princeton. Tous deux ont, pendant la guerre, développé indépendamment une méthode analogue pour faire des tests de qualité sur des biens manufacturés *(voir l'encadré page ci-contre)* ; la sélection d'un ensemble de biens manufacturés de même qualité pose le même type de problèmes que celle d'un mot probable. Là encore, Turing fait figure de pionnier. Venons-en aux méthodes qu'il met en place.

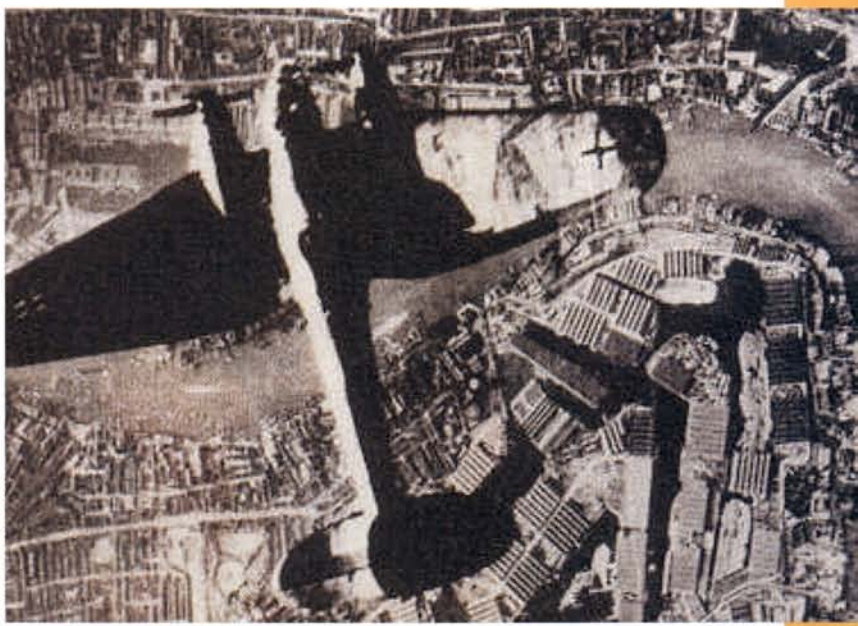
La solution de Turing

La méthode de Turing consiste en trois points. Pour augmenter la vitesse de traitement, Turing traite *indépendamment* chaque lettre par une *Bombe* dont le câblage est relié à la *Bombe* suivante, qui traite la lettre suivante, etc. Ainsi, la concaténation des *Bombes* imite le mot entier tout en permettant d'essayer les positions des crans des rotors de *l'Enigma* sur chaque lettre en même temps : en bout de traitement, toutes les *Bombes* ont donc le même ordre de cran de rotor, sauf le dernier qui exprime une lettre particulière, et elles étalent alors le mot dans l'espace, et non de façon séquentielle comme le ferait une *Enigma*.

En outre, Turing lance la recherche mécanisée tout en sachant que cette recherche ne peut pas donner le mot puisqu'il y aura un nouveau brouillage *après* celui effectué par les rotors : celui dû au tableau de connexions. Cette recherche indéfinie est donc interrompue à un moment donné, et Turing compare alors les lettres du mot probable et celles qui sont cryptées.

Enfin, Turing remarque que certaines configurations sont impossibles. En effet, selon la règle de *réciprocité dans le codage* par le tableau de connexions, une lettre et une seule peut être codée par une autre : par exemple, si une lettre *A* est codée par *P* au moyen du brouillage du tableau de connexions, la même lettre *P* ailleurs dans le texte codé ne peut être codée que par *A*. Bref, il doit y avoir *symétrie* entre la lettre codante et la lettre probable et quand ce n'est pas le cas, des milliards de configurations sont éliminées d'un seul coup. La recherche de connexions contradictoires peut dès lors être mécanisée : un premier prototype de la *Bombe de Turing* commence à fonctionner en mai 1940.

Un photomontage diffusé par la propagande allemande en 1940, représentant un bombardier allemand au-dessus des Surrey Docks, à Londres.



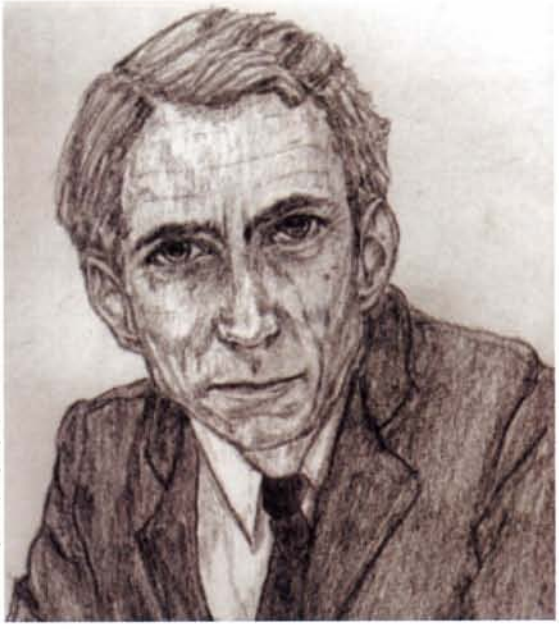
Southwark library Collection

Cependant, un collègue de Turing, le mathématicien Gordon Welchman, récemment arrivé dans le même service, se rend compte que le principe de l'analyse de Turing peut être généralisé. L'extension de la méthode par Welchman consiste à repérer de nouvelles contradictions à partir de celles déjà découvertes : puisqu'*Enigma* conserve par construction la symétrie entre le codage et le décodage, cette symétrie doit être respectée dans toutes les configurations. Par conséquent, une contradiction repérée *entraîne sa symétrique*. Dans la mesure où ces contradictions dépendent de celles déjà mises au jour, elles ne peuvent pas être prévues à l'avance. Néanmoins, elles permettent d'éliminer d'un coup autant de configurations que la contradiction originelle.

En août 1940 sont construites les premières *Bombes de Turing-Welchman*, appelées aussi plus tard *Bombes britanniques* en souvenir des *Bombes polonaises*. Elles opèrent une recherche systématique de toutes les contradictions dans le tableau de connexions pour toutes les configurations possibles des rotors. Les *Bombes* britanniques restent efficaces jusqu'au milieu de 1941.

Ainsi, trois *styles* différents en cryptologie ont rendue possible la maîtrise – jamais acquise une fois pour toutes – du décodage d'*Enigma* : entre Rejewski se limitant aux lettres de la clé du message, c'est-à-dire dépendant d'un état externe et Welchman recherchant systématiquement dans un même message les configurations contradictoires non prévisibles à l'avance, Turing occupe une *position intermédiaire* qui rassemble deux thèmes qui lui sont chers : le développement des méthodes déterministes (assimilables à ses travaux en axiomatique formelle) fondées sur des états probabilistes (ne dépendant pas d'une extériorité).

Le mathématicien Claude Shannon (1916-2001, dessiné par Bridgette Greenia), fondateur de la théorie de l'information. Lors de son passage aux États-Unis, en 1942-1943, Turing discuta longuement avec lui des notions d'information et d'intelligence.



Lexikon History of computing Encyclopedia on CD ROM

Un (trop) franc succès

Pendant toute cette période alternent succès cryptologiques et succès militaires : le 23 février et le 7 mai 1941, les Britanniques s'emparent des codes de l'ennemi sur un navire et, le 9 mai, à bord d'un sous-marin. Ces différents succès permettent de lire pendant un temps l'intégralité du trafic et de couler plusieurs navires et sous-marins (en particulier le *Bismarck*, le plus grand cuirassé de la flotte allemande, coulé avec l'aide du Service du chiffre le 27 mai 1941). À tel point que l'on s'inquiète, à *Bletchley Park*, de la *trop grande* efficacité du décodage, qui pourrait mettre la puce à l'oreille des Allemands et les inciter à modifier leur système de codage. En témoigne le message suivant, émanant probablement de l'état-major allemand, enregistré le 22 avril 1941 et décodé le 19 mai par l'équipe dont fait partie Turing :

DE : C dans C Marine

La campagne sous-marine rend nécessaire de restreindre sévèrement la lecture des messages par le personnel non autorisé. Encore une fois, j'interdis à toutes les autorités n'ayant pas reçu d'ordre exprès émanant de la Division des opérations ou de l'amiral commandant les sous-marins d'écouter la fréquence des sous-marins en opération. Je considérerai à l'avenir tout manquement à cet ordre comme un acte criminel mettant en danger la sécurité nationale.

Toutefois, le caractère paranoïaque de la dictature qui sévit en Allemagne pousse l'état-major à croire à un complot plutôt qu'à remettre en question l'inviolabilité supposée d'*Enigma*... Les choses changent en juillet 1941 quand, à côté d'*Enigma*, apparaît, pour les opérations navales, un autre système de codage fondé sur l'usage du téléscripteur, codage que les Britanniques dénomment *Fish* et qui concerne les messages émanant du haut commandement. Turing développe une méthode de décodage pour ce système, mais n'est pas le maître d'œuvre de cette entreprise, qui est confiée à Max Newman, son professeur de Cambridge. Ce dernier est arrivé à *Bletchley Park* pendant l'été 1942. D'autres responsabilités incombent à Turing.

À partir de février 1942, date d'entrée en guerre des États-Unis, la cryptologie du côté allié est autant une affaire américaine que britannique. Les Américains proposent de prendre le contrôle du trafic dans l'Atlantique en multipliant le nombre de machines à décoder : cela a pour effet de diminuer l'intérêt pour les méthodes probabilistes inventées par Turing, les machines à décoder étant suffisamment nombreuses pour résoudre les milliards de solutions par une recherche systématique. Les deux pays signent un accord concernant l'accroissement du nombre des machines ainsi qu'un échange immédiat et bilatéral d'information. C'est à Turing que

The History Place

Le Bismark, le plus grand cuirassé de la flotte allemande, qui fut coulé en mai 1941 avec l'aide du Service du chiffre britannique. Ci-contre, un des premiers bombardements aériens massifs sur Londres, en septembre 1940.

l'on confie la mission de faire le point avec les équipes de cryptologie américaines. Il part pour les États-Unis le 7 novembre 1942. À ce moment-là, l'entrée en guerre des États-Unis, bien que considérée comme une bénédiction côté alliés, provoque néanmoins une crise dramatique : les bateaux américains qui escortaient les convois jusqu'au milieu de l'Atlantique sont réquisitionnés dans le Pacifique pour participer à la guerre contre le Japon.

En voyage secret aux États-Unis

Le voyage secret de Turing aux États-Unis, toujours en partie couvert par le secret défense, reste obscur bien que des documents aient été rendus publics en 1996. On sait cependant que Turing visita les Laboratoires *Bell*, où on lui laissa libre accès, et qu'il s'intéressa particulièrement à la section chargée du cryptage de la parole. Pour assurer la sécurité des conversations téléphoniques de nature stratégique entre les gouvernements américain et britannique, il fallait en effet crypter la voix humaine en ondes sonores ; la technologie électronique, tout juste naissante, y trouva l'un de ses premiers champs d'application. Turing fut le seul étranger admis à voir le prototype et à travailler aux nombreux problèmes d'ingénierie qu'il posait.

Le projet d'installation d'une ligne cryptée appelée « projet-X » entre Washington et Londres se dessine en février 1943 et devient opérationnel en juillet 1943. Le séjour de Turing aux Laboratoires *Bell* a aussi des conséquences plus directement scientifiques : Turing y rencontre Claude Shannon, à qui il fait lire son article de 1936 et avec qui il a de nombreuses discussions concernant la notion d'information, la possibilité d'incarner la logique booléenne dans un dispositif matériel et d'imiter, par ce moyen, un fonctionnement jusqu'alors l'apanage de la pensée. Ainsi, l'idée d'une machine « capable d'imiter la pensée » commence à faire son chemin : le sens de l'expression anglaise *mechanical intelligence* qui, au début de la guerre, signifiait encore la mécanisation du *renseignement* commence à se transformer en mécanisation de l'*intelligence*, au sens de phénomène *mental*.

Lorsque Turing rentre à *Bletchley Park* après son voyage aux États-Unis, la situation a évolué : le décryptage des messages codés allemands est entré dans une phase industrielle et des milliers de personnes travaillent, à l'aide des méthodes probabilistes mises au point par Turing et des machines qu'il a conçues avec Welchman, sur les différents canaux cryptés de l'armée allemande. Max Newman dirige l'organisation des services et, en particulier, le projet de construction d'une nouvelle machine électronique de décryptage, le *Colossus*. Plusieurs versions successives en sont construites sous l'impulsion de l'ingénieur T. H. Flowers qui, dès les années 1930, s'est intéressé à l'utilisation de la technologie électronique en transmission du signal. La machine est consacrée au décryptage des messages *Fish*. Max Newman propose à Turing de revenir à la « Hutte n° 8 », celle qui traite de l'*Enigma* navale, mais Turing décline l'offre : il a conçu, sur le bateau qui le rame-

Paradoxalement, ce ne sont pas tant les travaux de cryptanalyse d'Alan Turing qui ont influencé la cryptologie moderne que ses travaux théoriques sur la Machine de Turing. *Bien évidemment, son approche de la* théorie de l'information *et les analyses statistiques pour mener à bien les attaques contre Enigma sont importantes, mais ces méthodes n'auraient plus le même impact sur les cryptosystèmes actuels, même avec les « bombes » bien plus performantes que sont les ordinateurs.*

En revanche, la Machine de Turing *est une modélisation de l'ordinateur toujours d'actualité, qui permet de définir le temps de calcul d'un programme, ou de l'accomplissement d'une tâche : la* complexité *d'un algorithme. Elle permet également de caractériser la difficulté relative de deux problèmes, avec la notion de* réduction. *La* réduction *d'un problème* P *à un problème* Q *consiste en effet à montrer que s'il existe une* Machine de Turing *capable de résoudre le problème* Q, *alors avec un nombre raisonnable d'appels à cette machine on peut résoudre le problème* P. *Ceci permet alors de conclure que le problème* Q *est au moins aussi difficile que le problème* P.

Cette comparaison de deux problèmes est désormais l'approche de la cryptologie moderne, au sein d'une cryptanalyse « constructive ». La cryptologie est divisée en deux branches : la cryptographie, *qui consiste à définir des protocoles cryptographiques, et la* cryptanalyse, *qui analyse leur sécurité effective. Pendant longtemps, ces deux branches ont mené une lutte acharnée, les cryptanalystes cherchant à attaquer les systèmes proposés par les cryptographes. Depuis une dizaine d'années, la cryptanalyse est devenue constructive : elle « prouve » la sécurité des protocoles cryptographiques.*

Plus concrètement, avec l'arrivée de la cryptographie à clé publique, on a proposé des systèmes (chiffrement ou signature) que l'on ne savait « casser » sans, par exemple, factoriser de grands entiers (le problème de la factorisation est considéré comme très difficile). Toutefois, une telle incapacité ponctuelle à casser le système n'excluait pas l'existence d'une technique non fondée sur la factorisation qui mettrait en défaut la sécurité. Il fallait donc envisager qu'une telle technique puisse être trouvée et mise en œuvre ultérieurement. Comment savoir si cette autre technique était facilement accessible ou non ? En comparant le problème du cassage du système cryptographique (censé garantir une notion de sécurité précise, telle que la confidentialité ou l'authentification) à celui de la factorisation. C'est l'objet des preuves de sécurité *: une telle preuve consiste à montrer la difficulté relative entre les deux problèmes en réduisant le second au premier. Par conséquent, sous l'hypothèse de la difficulté du problème de la factorisation (qui sous-entend que la factorisation est insoluble), le protocole cryptographique considéré est incassable.*

Ainsi, bien qu'Alan Turing soit plutôt connu pour ses attaques contre Enigma, *il a permis à la cryptologie de développer des méthodes d'analyse de sécurité : de nombreux systèmes concrets, en cryptographie aussi bien à clé secrète qu'à clé publique, ont ainsi été prouvés sûrs en utilisant ces techniques de réductions.*

David Pointcheval, *chargé de recherche au* CNRS, *et responsable de l'équipe de cryptographie de l'École normale supérieure*

Matyo

nait des États-Unis, un système de cryptage de la voix humaine. Depuis, il se consacre à ce problème de nature à la fois cryptologique et *physique*.

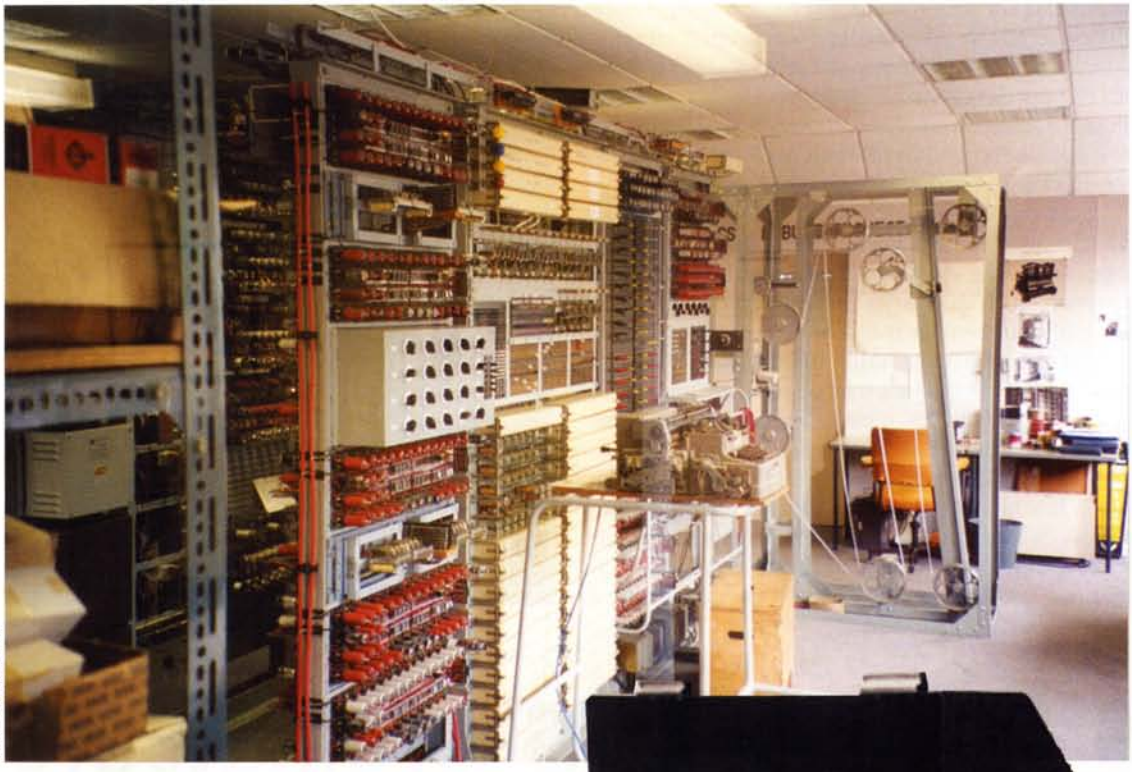
Dalila

En septembre 1943, tout en continuant à être régulièrement consulté pour des problèmes de cryptologie, Turing quitte *Bletchley Park* pour un laboratoire de recherche en ingénierie de l'armée situé à *Hanslope Park*, à une quinzaine de kilomètres de là. Il y conçoit et réalise, avec très peu de moyens, une machine électronique capable de crypter une voix humaine transmise par le téléphone. Il l'appelle *Dalila*, du nom du personnage biblique, parce qu'elle a su « se jouer des hommes ».

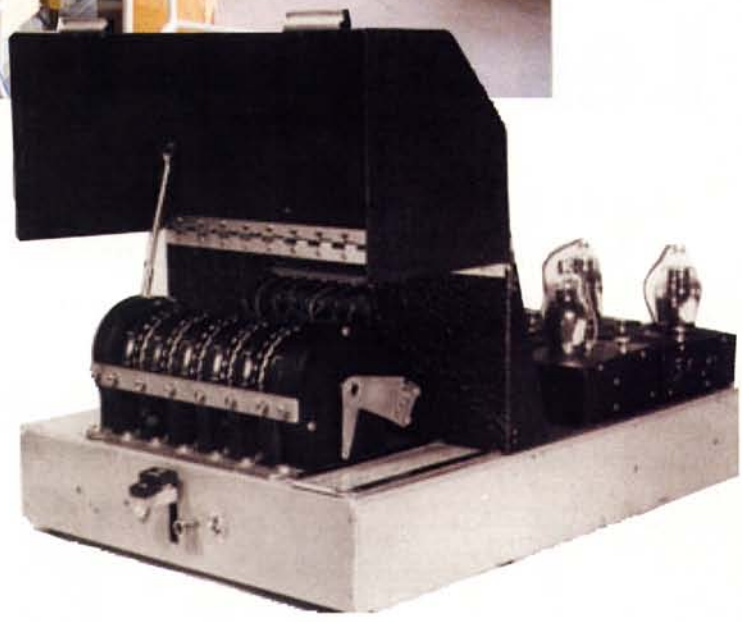
Le problème auquel répond Turing ici n'est plus celui qui se posait à *Bletchley Park* ou aux Laboratoires *Bell* : le message codé est une *onde physique*, la voix humaine. Ainsi, la clé du cryptage réside dans la façon dont l'onde elle-même est travaillée. Celle-ci doit donner, au niveau du message, l'*apparence de l'aléatoire complet*, tout en étant complètement déterminée. En outre, au niveau physique, *aucune perturbation aléatoire* ne doit introduire de désordre irréparable qui empêcherait la restitution de la voix. En d'autres termes, comment, dans un système déterministe, une onde physique peut-elle apparaître comme complètement aléatoire à un tiers tout en ne l'étant pas pour celui qui possède la clé de son décryptage ? Il faut faire en sorte qu'*une même onde* soit interprétée comme une *onde physique seulement* par un tiers et *comme message* par celui qui possède la clé. L'aléatoire est alors réduit à un sentiment subjectif propre à celui qui ne possède pas la clé et n'a plus aucune place dans le système physique déterministe envisagé.

Pour que le système de transmission soit bien un système physique déterministe permettant la reconstitution de la voix en bout de chaîne, il faut que les paramètres de transmission soient *linéaires*, de façon qu'un écart physique dans la transmission ne prenne pas des proportions trop grandes à la réception : l'aléatoire physique doit intervenir le moins possible dans la transmission. Pour cela, la synchronisation entre émetteur et récepteur doit être ajustée à la milliseconde près. C'est pourquoi la machine mise au point par Turing est limitée à la communication téléphonique ou à la liaison radio sur ondes courtes dans un périmètre local : les ondes radio plus longues ne permettent pas cette synchronisation à cause des perturbations présentes dans l'atmosphère.

Quelle est la nature du cryptage de l'onde physique mis au point par Turing ? Dans un rapport écrit le jour du débarquement de Normandie, il explique que les difficultés liées à la synchronisation de la transmission lui ont fait abandonner l'idée d'une segmentation de la clé reconstituable au moment de la réception. En conséquence,



National Security Agency



Une reconstitution de la machine* Colossus (ci-dessus), *qui fut construite en 1942 à* Bletchley Park *sous la direction de Max Newman pour décrypter un nouveau système de codage des Allemands, dénommé* Fish *par les Anglais.* Ci-contre, *la machine* Dalila *conçue par Turing en 1943 pour crypter les voix humaines.

il se propose d'utiliser une *clé périodique* sous la forme d'impulsions sur une période de huit minutes, à partir d'un répertoire de 10^{25} clés partagé par l'émetteur et le receveur. À chaque transmission, trois brouilleurs transforment partiellement la clé : la fréquence du signal qui constitue la clé est divisée en trois fréquences, qui sont ensuite réparties par les brouilleurs sur un spectre aussi uniforme que possible. Le temps de parole est limité à huit minutes (durée de réception de la clé) pendant lesquelles les trois différents types de signaux codés par les brouilleurs sont recombinés. Le prototype de *Dalila* est terminé au moment de la reddition de l'Allemagne, trop tard pour être utilisé pour les besoins de la guerre.

De juillet à août 1945, peu après la capitulation qui a eu lieu le 7 mai, Turing est envoyé en Allemagne *via* Paris, avec un groupe anglo-américain d'experts en communication, pour évaluer la situation allemande. Il visite le laboratoire de recherche sur les techniques de hautes fréquences et l'électro-acoustique qui a été fondé à Ebermannstadt, près de Bayreuth, où il rencontre des cryptologistes allemands qui font, devant un Turing jouant l'étonnement, une démonstration du cryptage effectué avec la machine que les Britanniques appelaient *Fish*. Pendant son séjour à Ebermannstadt, les deux bombes atomiques américaines sont lancées sur le Japon et Turing n'en est pas surpris, lui qui connaissait, depuis son voyage secret aux États-Unis de 1942-1943, l'existence du projet de Los Alamos – dans une proportion encore non élucidée. Il en profite pour décrire à ses collègues le principe de la réaction atomique. Le voyage en Allemagne confirme ce que les Alliés savaient déjà concernant le niveau technologique avancé de l'Allemagne.

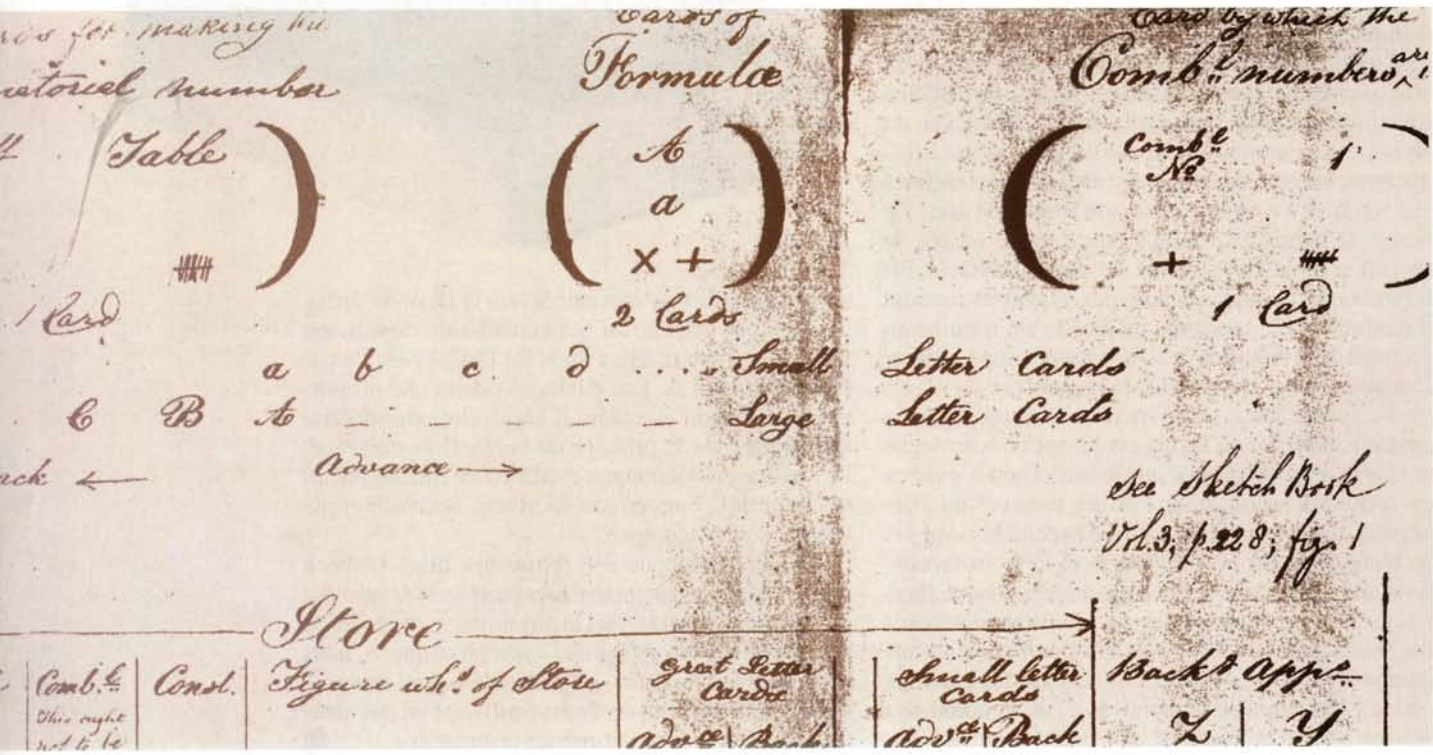
Ainsi, à partir de son retour des États-Unis en 1943, Turing *revient progressivement vers les sciences de la nature* : à la fois vers la physique – avec *Dalila* et le problème du codage de l'onde physique –, mais aussi vers la *biologie* – en particulier avec son projet de « construire un cerveau », qu'il ne faut pas identifier à celui de « construire un ordinateur ». ■

Les premiers ordinateurs

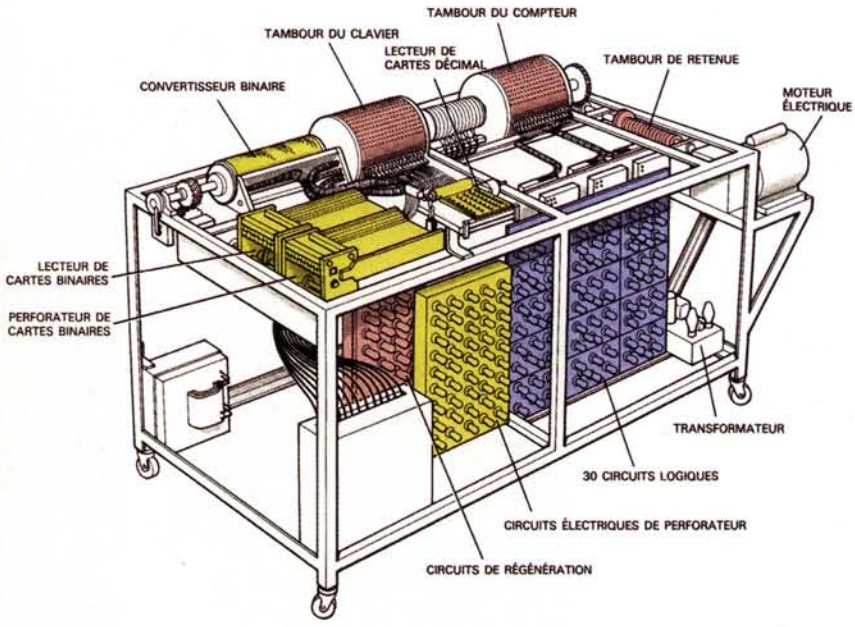
À la fin des années 1940, Turing participe à la construction de deux ordinateurs britanniques. Toutefois, ce n'est pas tant la prouesse technique qui l'intéresse que les rapports entre construction artificielle et croissance organique.

La construction des premiers ordinateurs a fait l'objet d'une querelle de priorité complexe parce qu'elle était associée à la fois à la défense nationale des pays concernés et, pour des raisons financières, à la délivrance de brevets commerciaux. Avec le recul, il est aujourd'hui possible de reconstituer l'histoire des projets et des constructions effectives des premiers ordinateurs.

Le premier ordinateur au monde fut conçu au XIX[e] siècle par le mathématicien anglais Charles Babbage, mais celui-ci ne put jamais le réaliser. De 1822 à sa mort en 1871, Charles Babbage dessina plusieurs plans d'un ordinateur mécanique, dont une réplique en état de marche fut construite à Londres en 1991 *(voir page 84)*. Cette machine est mécanique, programmable et « Turing-complète », c'est-à-dire équivalente à une machine universelle de Turing. On distingue aujourd'hui la préhistoire et l'histoire de l'ordinateur selon que la machine construite est ou non « Turing-complète ». Si l'on classe les projets et les réalisations selon ce critère, on constate alors que le premier ordinateur « Turing-complet » à avoir été construit est



Iowa State University Library/ Special Collection Department



allemand et date de *1941*. On imagine les graves conséquences sur la guerre si, au lieu d'avoir été marginalisé, l'ordinateur avait eu, côté allemand, le destin industriel auquel il aurait pu prétendre. Avant de cerner le rôle de Turing dans la naissance de l'informatique, rappelons ses grandes étapes.

Les premiers pas de l'informatique

L'informatique est née dans les années 1930, dans un foisonnement d'initiatives individuelles au sein de trois pays, l'Allemagne, la Grande-Bretagne et les États-Unis. Les machines construites visent toutes à automatiser les calculs et les opérations logiques. Pendant l'hiver 1937-1938, l'ingénieur américain John Atanasoff conçoit à l'*Iowa State College* un ordinateur programmable, électronique, binaire, digital, séparant les fonctions de mémoire et de calcul. En 1939-1942, ayant reçu une bourse, J. Atanasoff se lance dans la construction de cet ordinateur avec l'aide de l'ingénieur Clifford Berry. En 1943, l'ingénieur anglais Tommy Flowers conçoit (en février) et réalise (en décembre) à *Bletchley Park* les machines *Colossus I* et *II*, destinées à briser le code secret allemand dénommé *Fish*. Ces machines sont électroniques, binaires et digitales. Les machines d'Atanasoff et de Flowers font partie de la préhistoire de l'informatique : elles ne sont pas « Turing-complètes ». En d'autres termes, elles n'ont pas de programmes intégrés et nécessitent que l'on change leur câblage à chaque type d'opération. En outre, elles ne possèdent pas ce que l'on appelle « l'architecture von Neumann », c'est-à-dire une mémoire qui, contenant à la fois les instructions et les données, peut recevoir différents programmes. Les machines *Colossus* stockent néanmoins les programmes de façon électronique.

En 1941, l'ingénieur allemand Konrad Zuse conçoit et réalise, dans un isolement complet, le *Z3 (Zuse 3)*, premier ordinateur au monde « Turing-complet ». Cette machine est programmable (au moyen d'un langage de programmation appelé *Plankalkül*), binaire, digitale et électromagnétique. Cet ordinateur, qui effectuait des calculs dans une usine d'avions pour le profilage des ailes, fut détruit dans un bombardement allié en 1944. Konrad Zuse construisit une autre machine, la *Z4*, qui fut alors démontée par crainte des bombardements et remontée en 1945. En 1950, Konrad Zuse, ayant fondé une société commerciale, livra à l'École polytechnique de Zurich, où il avait fait ses études, un exemplaire de la *Z4*, qui continua d'y fonctionner jusqu'en 1954.

En février 1944, le physicien Howard H. Aiken et la mathématicienne Grace Hopper conçoivent, avec des crédits d'IBM, le *ASCC (Automatic Sequence Controlled Calculator)*. Le 7 août 1944, l'ordinateur est envoyé à l'Université de Harvard aux États-Unis, où il est nommé *Harvard Mark I*. Il s'agit d'une machine « Turing-complète », électronique et programmable.

Notes écrites par Augusta Ada King, comtesse de Lovelace (en haut), comportant un programme de calcul qu'elle a conçu pour le calculateur de Charles Babbage. Lady Lovelace participa activement à la conception de la machine de Charles Babbage, le premier ordinateur au monde. Ci-dessus, l'ingénieur américain John Atanasoff sur une photographie de 1938 et un schéma de la machine qu'il construisit à la même époque, l'ABC. Cet ordinateur résolvait des systèmes de 29 équations à 29 inconnues, en ajoutant ou enlevant de manière répétitive une équation à une autre, jusqu'à ce qu'une variable de la seconde équation soit éliminée. Pour cela, l'ABC lisait les coefficients des variables sur des cartes perforées, les convertissait en base 2, effectuait les opérations et stockait les nombres restants de l'équation sur des cartes perforées, pour un usage ultérieur.

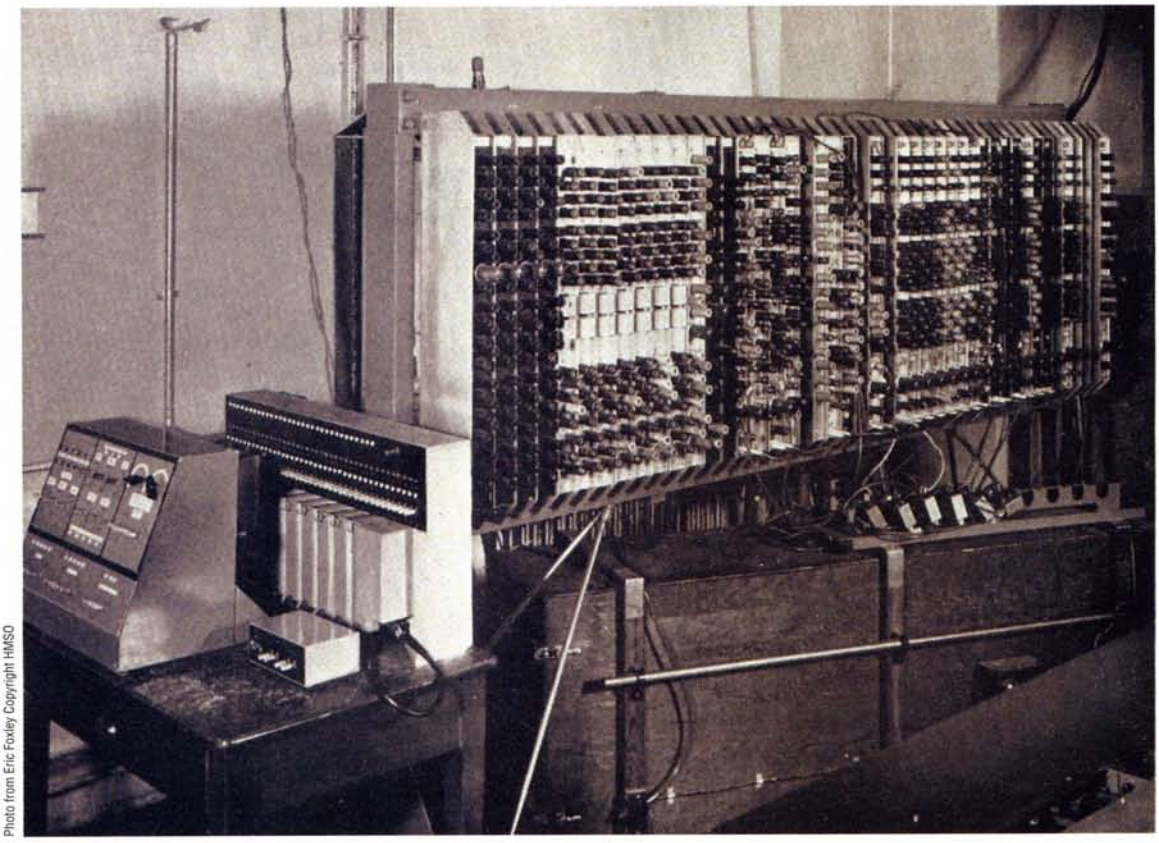


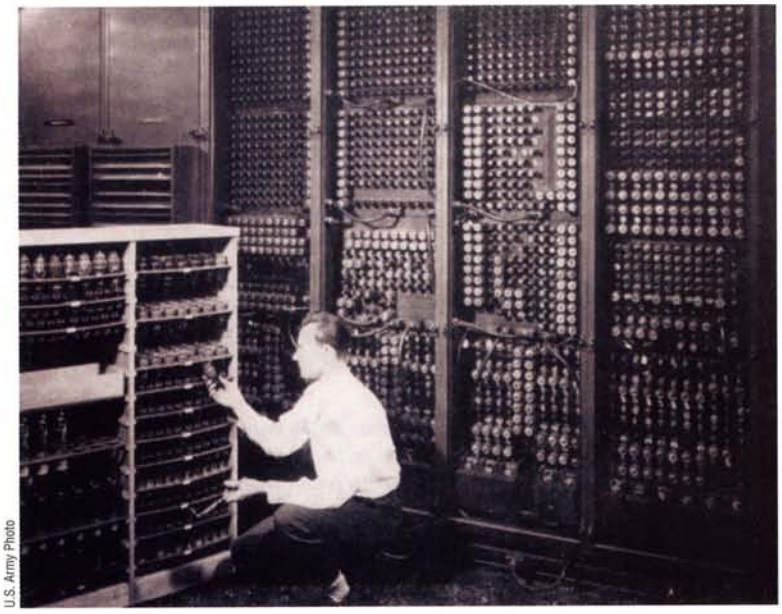
Photo from Eric Foxley Copyright HMSO

Le 15 février 1946, J. Presper Eckert et John William Mauchly, en s'inspirant de la machine d'Atanasoff, qu'ils ont rencontré de nombreuses fois, font fonctionner, à l'Université de Pennsylvanie aux États-Unis, une machine dont le contrat de construction avait été signé le 5 juin 1943, *l'ENIAC (Electronic Numerical Integrator and Computer)*. Cet ordinateur binaire, digital et électronique, qui doit être reprogrammé à chaque opération, est fondé sur un rapport de John von Neumann datant du 30 juin 1945, le *Draft Report on the Electronic Discrete Variable Automatic Computer* (EDVAC), dans lequel von Neumann se sert explicitement du concept de machine de Turing *(voir page 43)*.

Le 19 février 1946, Turing présente au *National Physical Laboratory* (NPL) de Teddington, en Grande-Bretagne, le projet *ACE (Automatic Computing Engine)*, ordinateur « Turing-complet » à programme interne. Une machine plus petite conçue sur le même principe, le *Pilot* ACE, ne sera achevée que le 10 mai 1950.

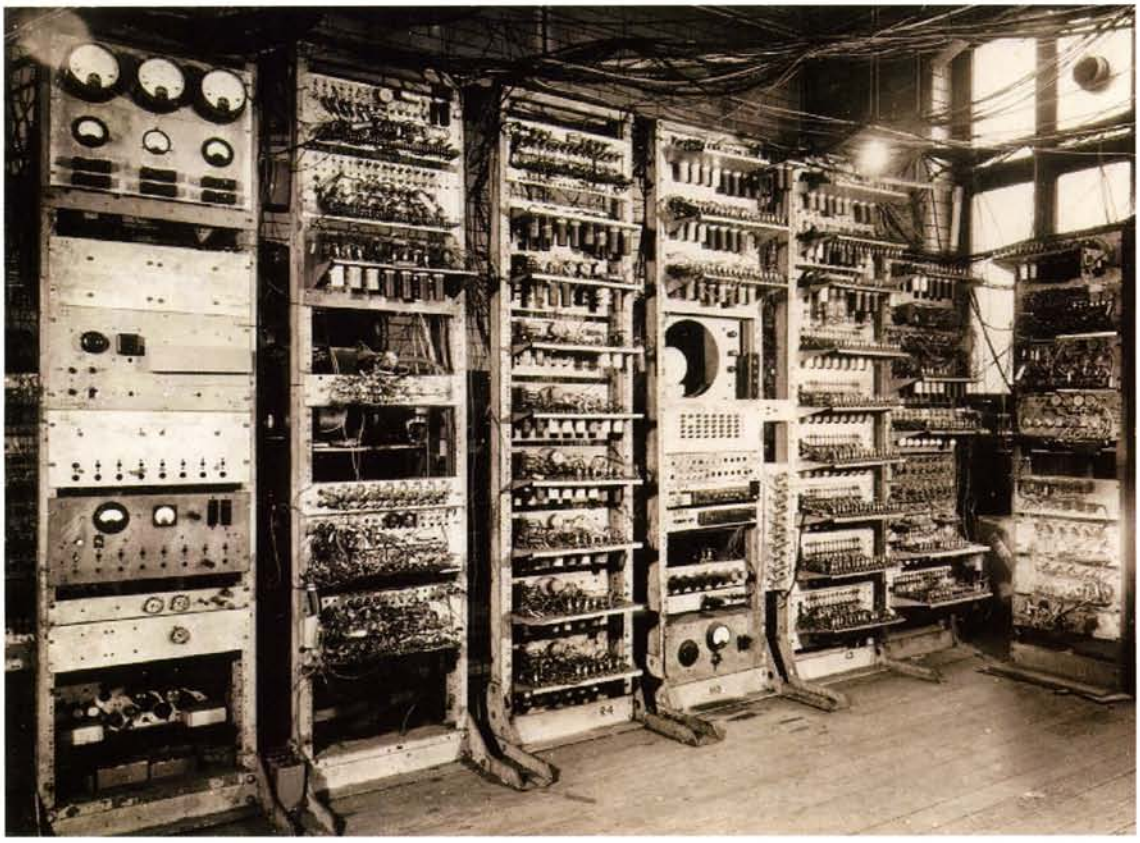
Le 21 juin 1948, le SSEM (*Small-Scale Experimental Machine*, surnommé « *Baby* »), premier ordinateur à programme intégré, conçu et réalisé par Frederic C. Williams et Tom Kilburn, devient opérationnel à l'Université de Manchester, en Grande-Bretagne. Turing fait partie de l'équipe à partir de juillet 1948. L'ordinateur est remplacé par un autre plus puissant, le *Manchester Mark I* (*Manchester Automatic Digital Machine*, appelé « MADM ») dès 1949.

Le 6 mai 1949 commence à fonctionner, à l'Université de Cambridge, Grande-Bretagne, l'EDSAC (*Electronic Delay Storage Automatic Calculator*), un ordinateur conçu sur les plans du rapport de John von Neumann de 1945. En août 1949, l'ordinateur conçu



U.S. Army Photo

Un technicien changeant un tube à vide défectueux de l'ENIAC, à rechercher parmi les 19 000 tubes qui composaient la machine. Le* Pilot ACE *(en haut) construit par le* National Physical Laboratory *en 1950, d'après le projet* ACE *présenté par Turing en 1946.



University of Manchester

par John von Neumann dans ce rapport, l'*EDVAC*, est enfin construit à l'Université de Pennsylvanie.

Dans ce foisonnement d'initiatives, Turing a joué un rôle important, moins en tant que maître d'œuvre d'un projet spécifique que par le fait qu'il est l'auteur du concept-clé de machine de Turing. À bien des égards, Turing est déjà décalé par rapport aux projets de construction technique de l'ordinateur et sa contribution dans ce domaine reste modeste. Ce qui l'intéresse au premier chef est d'incorporer ce projet technique dans un cadre théorique beaucoup plus général, celui du déterminisme, sous l'aspect des *rapports entre construction artificielle et croissance organique*.

Déjà en 1944, alors qu'il travaille sur sa machine à crypter la voix, il confie à l'un de ses assistants qu'il veut « construire un cerveau ». Or un cerveau ne se « construit » pas, puisqu'il est le résultat d'une croissance : ce dilemme entre construction artificielle et croissance organique oriente dorénavant sa recherche personnelle, beaucoup plus que la construction des ordinateurs ou celle d'une « intelligence mécanique ». C'est en gardant en mémoire les problèmes que posent les rapports entre construction et croissance que l'on apprécie pleinement la nature de sa participation à deux projets britanniques de construction de l'ordinateur qui se mettent en place dans les années 1945-1950.

Le moins de câbles possibles

Turing quitte *Hanslope Park* pour le *National Physical Laboratory*, basé à Teddington, le 1er octobre 1945. Le *NPL* est alors en train de se doter d'un département de mathématiques divisé en sections, dont l'une est réservée à Turing : celle qui a pour but la construction d'un ordinateur, l'*ACE*. Le projet de Turing se

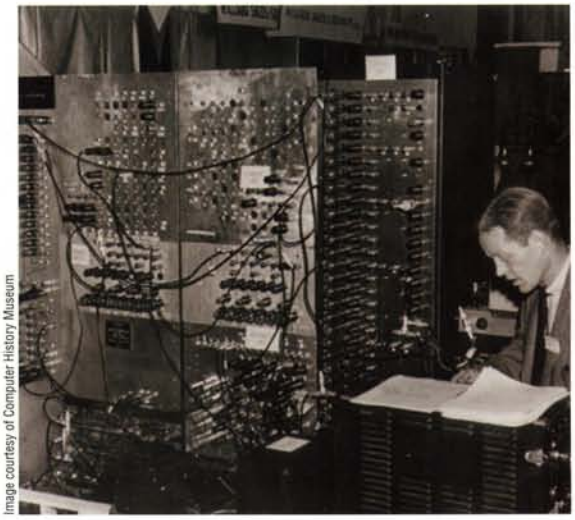
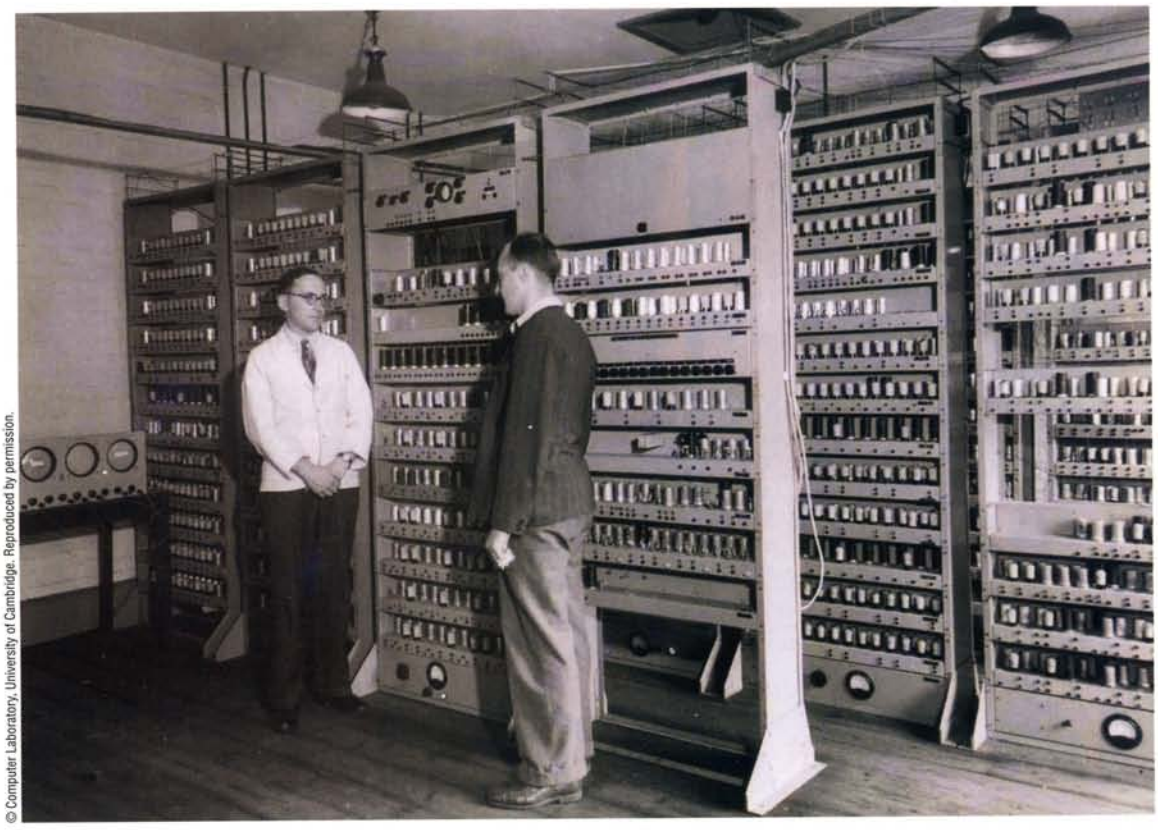


Image courtesy of Computer History Museum

L'EDVAC, ordinateur construit en 1949 à l'Université de Pennsylvanie à partir de plans de John von Neumann. Ci-dessus, l'ordinateur Manchester Mark I, construit en 1949 par l'équipe « informatique » de l'Université de Manchester, dont Turing faisait partie depuis 1948.



distingue des autres projets de l'époque, essentiellement américains, par le fait qu'il cherche à réduire au maximum tout câblage dédié à une opération particulière ; d'un point de vue logique, en effet, toute opération matérialisée dans un câblage peut être remplacée par des lignes supplémentaires de programmation. Ainsi, Turing appelle sa machine une « machine de papier », ses seules pièces matérielles vraiment importantes étant la *mémoire* et le *contrôle*.

À l'époque, le problème du stockage en mémoire est délicat parce que le rationnement impose de faire flèche de tout bois et d'utiliser des technologies soit surannées, soit peu fiables. Fort de son expérience récente avec sa machine à crypter la voix, Turing opte pour un composant dont la technologie est bien maîtrisée, le retardateur à mercure. Le retardateur à mercure relève de la physique des cordes vibrantes : il s'agit d'un tube rempli de mercure, relié à chaque bout à un transducteur piézo-électrique, dispositif qui reçoit, puis émet un signal. Chaque son reçu à un bout du tube, assimilé à un signal représentant un nombre, propage une onde dans le mercure, qui est reçue comme son par l'autre transducteur piézo-électrique. Il est alors possible, grâce

***M. V. Wilkes, l'un des pionniers du* Computer Laboratory *de l'Université de Cambridge, devant un retardateur à mercure. Ci-dessus, l'*EDSAC*, ordinateur construit à l'Université de Cambridge en 1949 d'après les plans fournis par John von Neumann dans son rapport sur l'*EDVAC.**

à un autre composant, de propager à nouveau le même son à travers le tube. Tant qu'il y a alimentation électrique, le tube fonctionne comme une mémoire relativement rapide, la vitesse de propagation du son dans le mercure étant élevée (1450 mètres par seconde). De nombreux signaux sont ainsi lancés à travers le tube, propageant des ondes dans le mercure. Reste alors à synchroniser le contenu de la mémoire avec l'ordre des opérations en cours dans l'ordinateur. Ce contrôle du caractère séquentiel des opérations est assuré par une horloge générale qui bat la mesure.

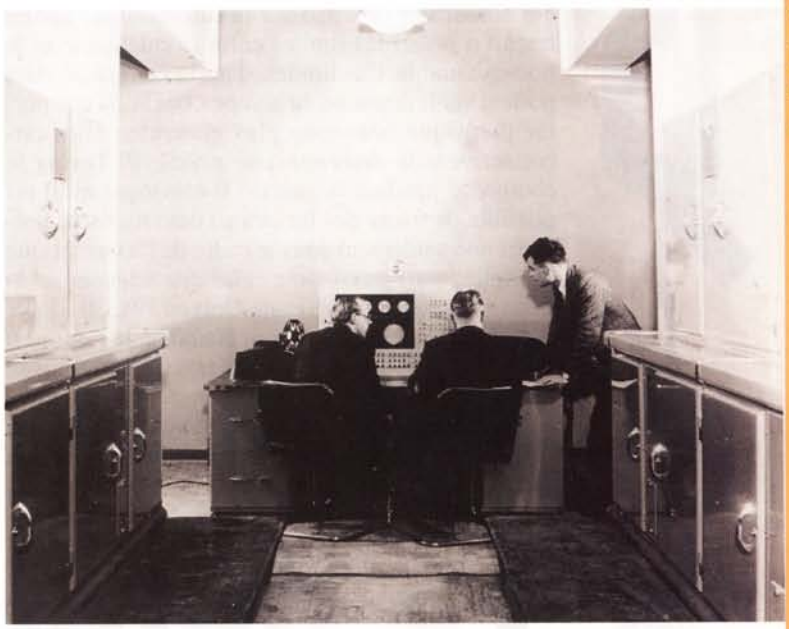
L'idée-phare du projet *ACE* de Turing, idée qu'il considère comme « très puissante » et qui fut aussi découverte par von Neumann, consiste à permettre à la machine elle-même d'effectuer différentes instructions selon l'étape du calcul en cours et à modifier ainsi son programme interne. En d'autres termes, Turing établit une hiérarchie dans les instructions des programmes et fait, autant de fois que nécessaire selon les signaux à traiter, des boucles à l'intérieur des lignes du programme.

Dans un rapport achevé fin 1945, Turing prophétise que l'*ACE* exécutera le travail de 10000 calculateurs humains. Il ajoute que tout travail programmé pour être traité par l'*ACE* pourra être réalisé à distance, grâce aux lignes téléphoniques. Il ne suffira alors que d'un nombre limité de personnes auprès de la machine pour s'occuper de sa maintenance. Malheureusement pour Turing, un peu trop en avance sur son temps, la réalisation du projet, décidée le 8 mai 1946, traîne en longueur pendant quatre ans, pour des raisons plus administratives que techniques. L'ordinateur *ACE* n'est inauguré qu'en novembre 1950.

« Construire un cerveau »

Pendant son séjour au *NPL*, Turing est envoyé du 7 au 10 janvier 1947 à une conférence à Harvard, aux États-Unis, où se tient un large symposium sur la construction des calculateurs. Il en revient sans avoir appris grand-chose de nouveau, sans doute parce que, comme il le confie à l'époque au neurologue W. R. Ashby, il ne s'intéresse pas directement à la mécanisation du calcul en tant que telle : « En travaillant à l'*ACE*, je suis plus intéressé par la possibilité de produire des modèles de l'activité du cerveau que par les applications pratiques du calcul. »

Cette problématique marginale l'éloigne de ceux qui, au *NPL*, sont préoccupés par la réalisation concrète du projet. Le 30 septembre 1947, Turing donne sa démission et repart à Cambridge pour l'année universitaire 1948-1949, en utilisant la fin de son allocation de recherche qu'il avait reçue en devenant *Fellow* et qu'il n'avait pas utilisée entièrement du fait de la guerre. Au même moment, il accepte de partir l'année suivante à Manchester, dans un nouveau laboratoire d'informatique, créé par son ancien professeur de Cambridge Max Newman. Contrairement à ce que l'on aurait pu croire, ce n'est pas pour faire des mathématiques ou pour participer à la constitution universitaire de l'informatique naissante qu'il repart à Cambridge, mais pour suivre des cours de physiologie : Turing reste fidèle aux questions théoriques qu'il se pose depuis la fin de la guerre sur les rapports entre la construction matérielle et la croissance organique.



Alan Turing penché sur la* Ferranti Mark I Console, *un ordinateur construit en 1951 par l'équipe informatique de Manchester. Turing est en compagnie de Brian Pollard et Keith Lonsdale, deux autres ingénieurs de l'équipe.

À Manchester, où il est nommé en juillet 1948, Turing arrive au sein d'une équipe déjà constituée et qui a déjà réussi le tour de force de faire « tourner » pour la première fois un programme, le 21 juin 1948. Il occupe un poste à part, créé pour lui.

Du point de vue logiciel, il se lance tout d'abord dans un travail de programmation visant à simplifier la machine. Toutefois, Turing travaille dans une arithmétique de base 32 (pour des raisons liées à la taille des tubes de mercure servant de mémoires), ce qui ne facilite ni la programmation ni les rapports avec ses collègues. Du point de vue matériel, il essaye de construire un générateur aléatoire. On devine ici la nature de ses préoccupations : Turing tente de réintroduire, dans la construction intégralement déterministe de la machine, une composante non déterministe permettant de simuler la nature *physique*. Pour ce faire, il utilisa sans doute son expérience liée à la conception de *Dalila* : *Dalila* contenait un générateur que l'on qualifierait aujourd'hui de pseudo-aléatoire, reposant sur l'usage d'algorithmes qui simulent l'aléatoire, mais, étant déterminés, ne peuvent pas être complètement aléatoires.

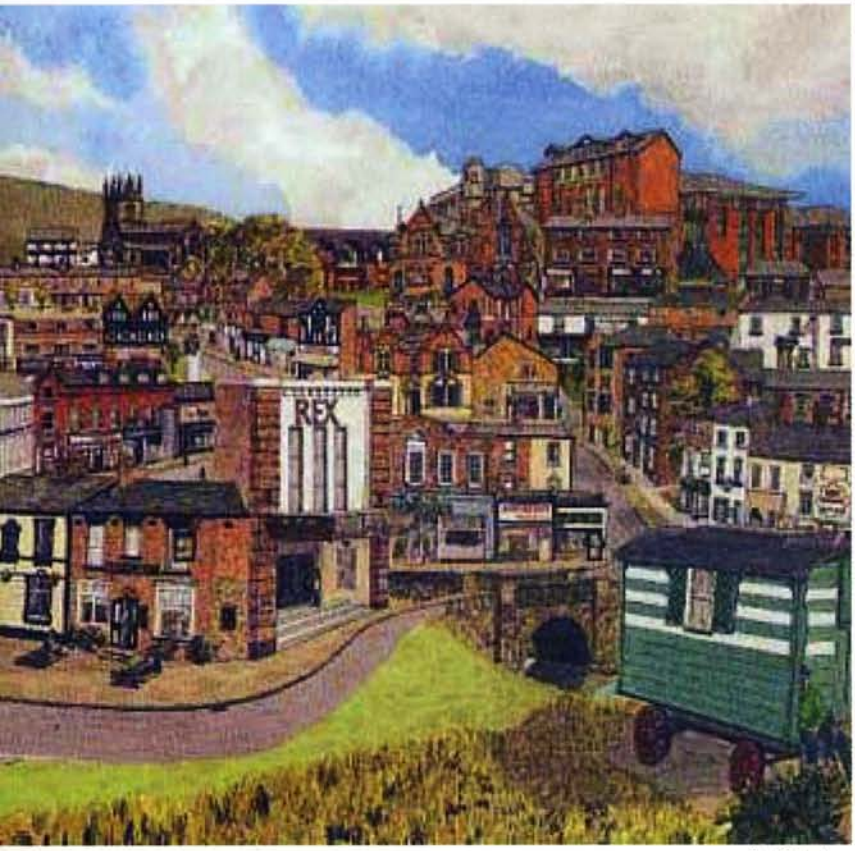
Ainsi, là encore, Turing se confronte avec le rapport entre le niveau informatique, de nature intégralement déterministe, et le niveau physique dans lequel l'aléatoire a une part objective. Nous verrons que c'est précisément ce rapport qui l'occupera, d'un point de vue théorique, jusqu'à la fin de sa vie. ■

Le déterminisme

Pourquoi les êtres vivants sont-ils similaires d'une génération à l'autre ? Pourquoi le cœur des fleurs est-il organisé ? Telles sont les questions auxquelles se consacre Turing à partir de 1947.

Dans son article de 1936 sur la calculabilité, Turing traçait *a priori* les limites entre le calculable et le non-calculable. Ces limites, dans la mesure où elles portent sur le domaine du nombre, ont aussi une portée théorique beaucoup plus générale : elles circonscrivent le *déterminisme prédictif*. Turing le comprend pendant la guerre : il remarque qu'il est possible de tracer des limites au déterminisme prédictif non seulement dans le cadre de l'axiomatique formelle, *mais aussi dans celui des sciences de la nature*. À son retour des États-Unis en 1943, il adopte donc la posture théorique qui était déjà la sienne en 1936, mais déplacée aux sciences de la nature : il se situe *des deux côtés de la limite* caractérisant le déterminisme prédictif, d'une part en participant à la *matérialisation* du déterminisme prédictif dans une machine – l'ordinateur – et, d'autre part, en explorant des domaines expérimentaux qui se situent *au-delà* du déterminisme prédictif, dont un en particulier, la morphogenèse biologique.

Le fruit de ce travail est son article de biologie théorique publié en 1952 dans les *Philosophical Transactions of the Royal Society* sous le titre *Chemical basis of morphogenesis (La base chimique de la morphogenèse)*, que Turing juge aussi fondamental que celui de 1936 : dans les deux cas, Turing invente des concepts nouveaux – celui de « machine de Turing » d'une part, et, nous allons le voir, celui de « structure de Turing » de l'autre. Ces concepts lui permettent de tracer des limites théoriques analogues entre processus prédictibles et processus non-prédictibles. Ce point de vue est fructueux, parce que Turing



Stuart Originals

The Chineese University of Hong Kong

ne se contente jamais d'étendre le domaine du déterminisme prédictif à un substrat déjà existant ; il articule des phases mutuellement exclusives du déterminisme *pour faire surgir de nouveaux phénomènes*. La fécondité de la science tient moins à la recherche de l'exhaustivité qu'à la construction de nouveaux objets scientifiques.

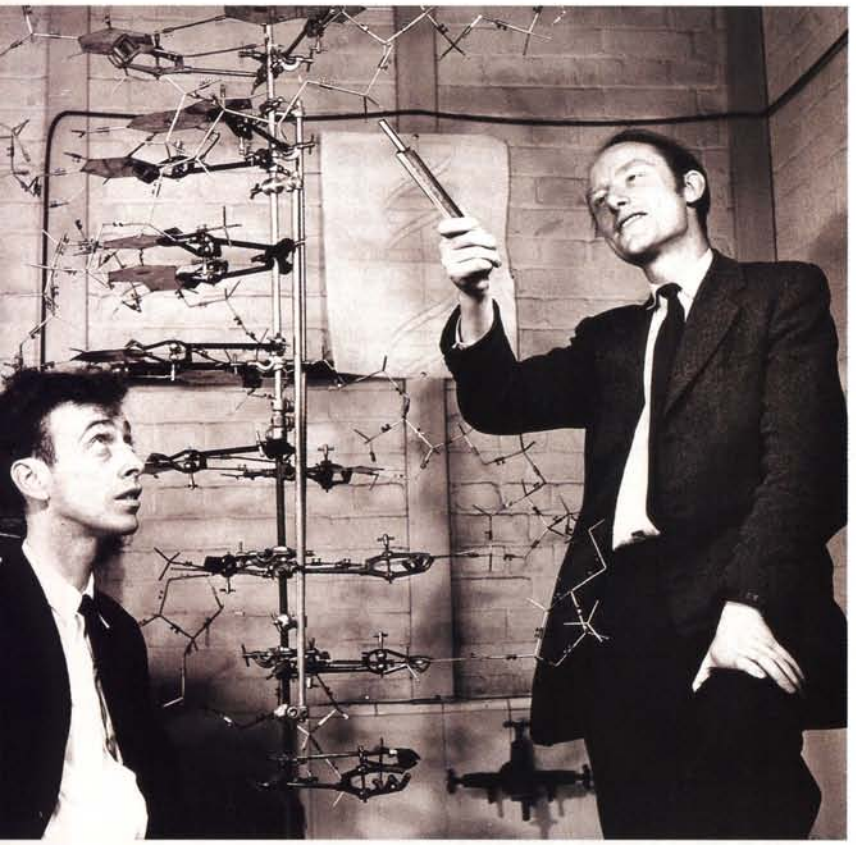
Les limites du déterminisme prédictif avaient déjà été explorées par Henri Poincaré (1854-1912). Le mathématicien et physicien français s'était notamment intéressé au problème des trois corps. En effet, toute prédiction *a priori* sur l'évolution d'un système physique possédant trois corps ou plus est impossible : on peut écrire les équations qui décrivent l'évolution d'un système constitué de trois corps mûs par leurs interactions gravitationnelles, mais on ne peut prédire leur trajectoire. Puis quelques mathématiciens russes, comme Alexandre Andronov (1901-1952), s'étaient penchés sur le problème. Toutefois, ces résultats étaient restés isolés, sans doute parce que la question des rapports entre les limites du déterminisme dans les sciences abstraites (mathématiques et logique) d'une part, et celles de la nature (physique et biologie) de l'autre, ne se posait pas encore avec une assez grande précision.

C'est cette question qui apparaît à Turing après 1943 : à partir du moment où la construction d'une machine déterministe au sein de la nature devient possible, les recherches sur le déterminisme dans les sciences abstraites et celles de la nature se rejoignent. Il devient alors possible de s'interroger sur les limites du déterminisme prédictif en général. Ainsi, l'année

et le vivant

À la fin des années 1940, Turing revient à ses amours de jeunesse : il s'interroge sur l'organisation dans le vivant. Pourquoi, par exemple, les moutons se ressemblent-ils de génération en génération ? Page ci-contre à gauche, une vue de Wilmslow, près de Manchester, où habite Turing à la fin de sa vie, sur une peinture de Martin Stuart Moore. Page ci-contre à droite, l'Université de Manchester, où Turing effectue ses recherches en biologie à la fin des années 1940.

© Mark Lord/Shutterstock



En 1953, Jim Watson et Francis Crick découvrirent la structure en double hélice de l'ADN. Cette découverte alimenta la métaphore du « programme génétique » selon laquelle tous les êtres vivants ont le même code génétique et seules les instructions d'exécution du programme changent en fonction des individus. Bien que la métaphore s'inspire de la machine de Turing, celui-ci ne chercha pas du tout à traduire son idée du déterminisme du vivant en termes de programme génétique.

1943 marque le début d'une tentative unitaire : Turing, tout en restant déterministe, adopte un point de vue à la fois prédictif-calculable (en logique avec la machine de Turing et en physique avec l'ordinateur) et non prédictif, non calculable (en logique avec le problème de l'arrêt et en biologie avec la morphogenèse). Décrivons les recherches biologiques qu'il mène à plein temps à partir de 1948, mais pour lesquelles il portait un intérêt depuis l'enfance.

Le déterminisme dans la nature

Turing quitte le *National Physical Laboratory* au printemps 1947 et passe une année sabbatique à Cambridge pendant l'année universitaire 1947-1948 pour se consacrer à ses recherches biologiques. Il accepte en outre un engagement au *Manchester University Computing Machine Laboratory* où Max Newman a créé pour lui un poste à la rentrée universitaire 1948. Jouissant d'une grande liberté dans ses recherches, il s'immerge dans la biologie du développement, même s'il fait un certain nombre d'exposés « grand public » sur ce qui s'appelait à l'époque l'« intelligence mécanique ». Turing est élu *Fellow* de la *Royal Society* le 15 mars 1951 à partir d'un rapport signé par Bertrand Russell et Max Newman. Ce rapport mentionnait son article de 1936, ce qui fit dire à Turing qu'il aurait pu être nommé à l'âge de 24 ans… Manchester sera son dernier poste universitaire, puisque c'est dans la maison qu'il achète à Wilmslow, près de Manchester, qu'il se suicidera trois ans plus tard, le 7 juin 1954.

Depuis 1943, Turing a pris une certaine distance critique à l'égard du paradigme formaliste du codage qui a cours dans l'axiomatique formelle ou la cryptographie. Il s'intéresse à l'origine des formes organiques de la nature, leur apparition première, mais aussi à la raison pour laquelle une forme organique donnée engendre une forme organique analogue à la génération suivante. Ce processus le fascine. Il est

Vue d'artiste du double pulsar **PSR J0737–3039** *découvert en 2003. Les pulsars sont des phares cosmiques : ces étoiles tournent sur elles-mêmes en émettant des faisceaux d'ondes radio qui balayent l'espace et apparaissent comme des éclairs périodiques aux observateurs. Les pulsars font partie des rares systèmes physiques déterministes prédictifs.*

déterministe, puisqu'une forme organique n'engendre pas n'importe quoi (un chat n'engendre pas une grenouille). Mais il ne résulte d'*aucun codage* : c'est l'organisation interne de la matière, sa déformation, qui fait surgir des formes et les maintient, avec des variations, à travers les âges. On est loin de l'axiomatique formelle ou de la cryptographie, où les formes (théorème ou message) relèvent d'un code (celui du codage numérique des propositions ou des messages). Mais de quel déterminisme s'agit-il, si la notion de codage en est absente ?

Avant de répondre à cette question, dissipons un malentendu. Une solution vient tout de suite à l'esprit aujourd'hui, et on imagine mal Turing passer à côté : la métaphore du « programme génétique ». Selon cette métaphore, le code génétique est le même chez tous les êtres vivants, et chaque individu est le fruit de la traduction du code en un programme génétique spécifique, c'est-à-dire en une liste d'instructions spécifiques, semblable à un programme informatique. En tant que théoricien de la notion de programme informatique, Turing aurait pu appliquer cette notion à la biologie. Or *il n'en est rien*. Certes, la métaphore est plus récente que les travaux de Turing, puisqu'elle s'est progressivement constituée à partir de la découverte de la structure de l'ADN par Watson et Crick en 1953 jusque dans les années 1970 avec les travaux des biologistes français Jacques Monod et François Jacob. Néanmoins, on peut dire rétrospectivement que *Turing l'a déjà dépassée* : il a compris que la question du déterminisme inhérent aux formes vivantes est une question limite où s'articulent le prédictif relevant du programme et le non-prédictif qui n'en relève pas.

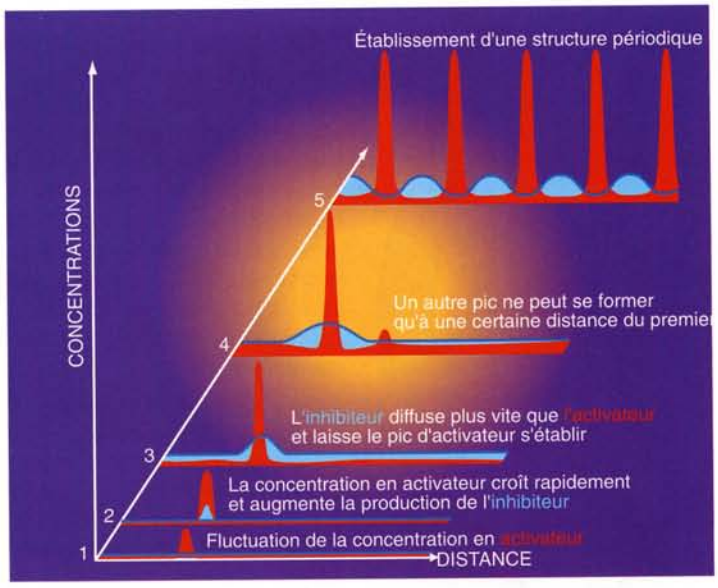
C'est d'ailleurs pour cette raison que la forme vivante entre dans son cadre méthodologique. Dans le cas des systèmes déterministes prédictifs, la prédiction est possible parce que ni l'espace ni les paramètres rendant compte de l'évolution du système considéré ne changent au cours du temps : le processus évolutif peut toujours être reconstitué, y compris si on inverse la direction du temps. Au contraire, dans de nombreux systèmes physiques et plus encore dans des systèmes vivants, *le nombre et la nature des paramètres changent au cours du processus*, modifiant la nature de l'espace où se déroule l'évolution. Ainsi, *l'itération*, si fondamentale en théorie de la calculabilité et en informatique, devient ici partiellement inopérante : aucun processus dans un système déterministe non prédictif n'est itérable sous les mêmes conditions initiales.

Seuls des systèmes physiques rudimentaires comme le pendule simple ou le pulsar ne sont pas « sensibles aux conditions initiales » dans la nature. En d'autres termes, seuls quelques rares systèmes physiques sont descriptibles à l'aide d'un programme d'ordinateur. La plupart des problèmes physiques ne relèvent du paradigme déterministe que par approximation et montrent donc la limite de la matérialisation d'une machine de Turing en un ordinateur. On retrouve ici la méthode chère à Turing, où il décrit *à la fois* le déterminisme calculatoire prédictif et ses limites intrinsèques.

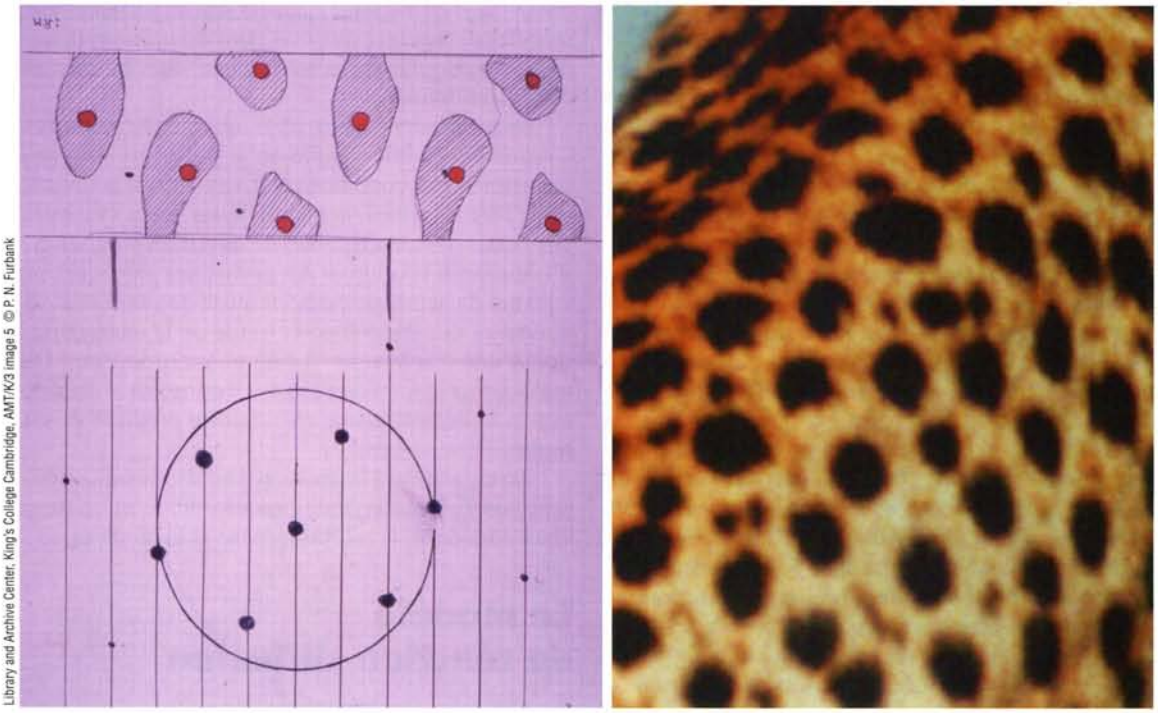
Turing répond à la question du déterminisme inhérent aux formes vivantes en *physicien* et ce dans deux domaines, la morphogenèse et la phyllotaxie.

Le modèle de réaction-diffusion

La morphogenèse, étude des mécanismes de croissance des formes biologiques à partir de leur fécondation, est une branche de la biologie du développement. Le modèle imaginé par Turing consiste, en partant d'un état homogène de la matière, à étudier, d'un point de vue mathématique, l'origine d'une organisation. Turing appelle cette origine « instabilité catastrophique » : elle est le résultat d'une singularité, décrite mathématiquement comme une transition infinitésimale entre deux états, transition pouvant provoquer des changements radicaux dans l'organisation de la matière, comme la transformation subite de l'eau en glace. En d'autres termes, cette instabilité catastrophique rend possible l'apparition d'une forme géométrique discrète au sein d'un milieu au départ amorphe et homogène.

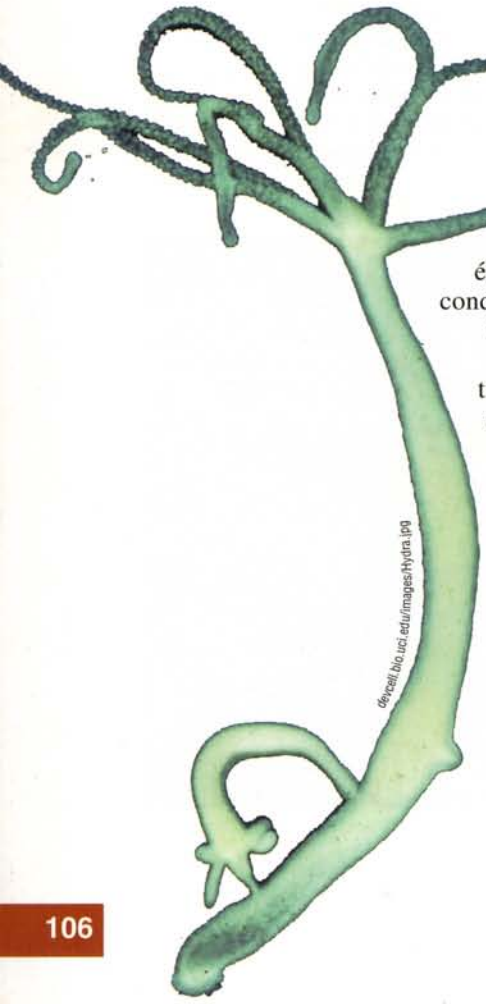


Naissance d'une structure de Turing. Dans un système activateur (rouge)-inhibiteur (bleu) homogène apparaissent des hétérogénéités de concentration. Sous l'effet de fluctuations du milieu, un léger excès d'activateur (1) se forme localement. La concentration locale en activateur croît alors (2), entraînant la production locale d'inhibiteur. L'inhibiteur diffuse plus rapidement (3) que l'activateur, empêchant la concentration d'activateur d'augmenter autour du pic d'activateur, qui est ainsi circonscrit. La concentration en activateur diminue de même autour du pic. D'autres pics d'activateur émergent (4), mais à distance du premier. Au final, une structure stationnaire périodique se forme (5).



Library and Archive Center, King's College Cambridge, AMT/K/3 image 5 © P. N. Furbank

Recherche d'Alan Turing sur la dispersion des tâches qui constituent les structures de Turing (à gauche). À droite, une structure de Turing de la nature : les motifs du dos d'un guépard. De nombreux montages expérimentaux reproduisant les structures de Turing ont été réalisés depuis ses travaux. Une façon d'en faire apparaître consiste à maintenir artificiellement l'instabilité catastrophique entre les substances chimiques inhibitrices et activatrices par une alimentation continue du système en substances selon des concentrations différentes. On se place alors expérimentalement dans l'intervalle de temps « catastrophique » et l'on rend visibles les phénomènes d'ondes stationnaires.



devcell.bio.uci.edu/images/Hydra.jpg

Hydra, *petit polype d'eau douce, ne semble pas perturbé par les endommagements. Il se répare et retrouve un état d'équilibre comme s'il ne l'avait jamais quitté. Son comportement est globalement prédictif.*

Le modèle, appelé par Turing « réaction-diffusion », se limite aux réactions chimiques et laisse de côté toutes les autres contraintes physiques comme les forces mécaniques ou électriques ; il décrit seulement les conditions de conversion, au sein du vivant, d'une dynamique de nature chimique en forme géométrique. Turing se donne deux substances idéales, appelées « morphogènes », qui diffusent avec des vitesses différentes au sein d'un système et qui ont respectivement un rôle activateur et inhibiteur. Au départ, les deux morphogènes sont dans un état stable, mais une perturbation aléatoire entraîne une production supplémentaire de l'un des morphogènes, production assez élevée pour que le système quitte l'état d'équilibre. La vitesse de propagation des substances chimiques a alors tendance à croître selon une « dérive exponentielle », explique Turing, qui rend impossible toute prédiction sur l'état futur du système. Si le système se conserve en tant que système, c'est que des mécanismes régulateurs internes évitent son explosion sous le coup d'une propagation trop rapide. Quels sont ces mécanismes ?

Turing s'en tient à l'étude des cas qui ne dévient pas trop de l'état d'équilibre. La compétition entre réaction et diffusion fait alors apparaître localement un état oscillant composé d'*ondes stationnaires*, phénomènes d'auto-organisation du milieu que l'on appellera plus tard des « structures de Turing ». En décrivant analytiquement les paramètres de contrôle des équations de propagation des deux morphogènes, on détermine les cas où les états oscillants apparaissent. Ainsi, en maintenant de façon artificielle le système dans un état proche de l'équilibre, on révèle des phénomènes physiques qui, sinon, seraient passés inaperçus. Turing remarque, dans son article de 1952 :

L'objet particulier de l'enquête consistait à étudier des phénomènes au moment où le système entrait dans une phase instable. Pour rendre le problème

mathématiquement traitable, il était nécessaire de faire l'hypothèse que le système ne déviait jamais très loin de son homogénéité originelle. Cette hypothèse était appelée « hypothèse de linéarité » parce qu'elle permettait de remplacer les fonctions générales de taux de réaction par des fonctions linéaires. L'hypothèse de linéarité est importante. Sa justification provient du fait que l'on s'attend à ce que les formes produites dans les premières étapes quand l'hypothèse est valide ont une forte similarité qualitative avec celles qui s'imposent dans les étapes ultérieures quand elle ne l'est plus.

L'hypothèse de linéarité est une hypothèse laplacienne, dans le sens où elle rend possible un déterminisme prédictif *local*. Elle n'est pas généralisable puisque, dans le cas général non-linéaire, c'est-à-dire dans les cas qui s'éloignent trop de l'équilibre, aucun traitement systématique n'est envisageable.

Turing ne décrit pas la façon dont les ondes stationnaires sont conservées dans la durée chez un organisme donné. Pour lui, ces ondes sont l'étape finale du développement de l'organisme au cours de l'histoire de son individuation et expliquent en partie les causes de sa forme. Les exemples étudiés par Turing sont la disposition des taches et des couleurs de pelages ainsi que la structuration en anneau des tentacules chez certains animaux.

Ce dernier exemple s'inscrit tout à fait dans l'hypothèse laplacienne : il porte sur un petit polype d'eau douce, *Hydra*, qui se dédouble en réarrangeant une partie sectionnée de lui-même pour former un nouvel organisme complet. À l'instar de certaines cellules qui, au cours de la morphogenèse des organismes s'adaptent au milieu dans lequel elles sont implantées à certaines phases précises du développement, *Hydra* retrouve un état d'équilibre global comme si, au moyen d'une régulation laplacienne, il conservait un état d'équilibre malgré toutes les perturbations « catastrophiques ».

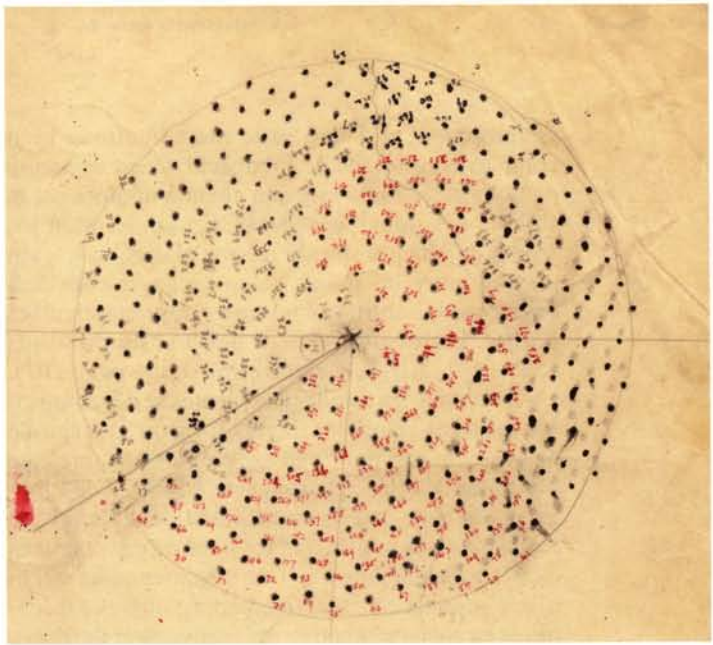
Cependant, la description mathématique présentée par Turing ne correspond pas exactement au cas auquel un biologiste s'attend. Ainsi, le biologiste de la morphogenèse C. H. Waddington écrit à Turing le 12 septembre 1952 que l'exemple des taches sur les pelages des animaux lui semble avoir une plus grande plausibilité expérimentale que celui des *Hydras*.

La phyllotaxie et la suite de Fibonacci

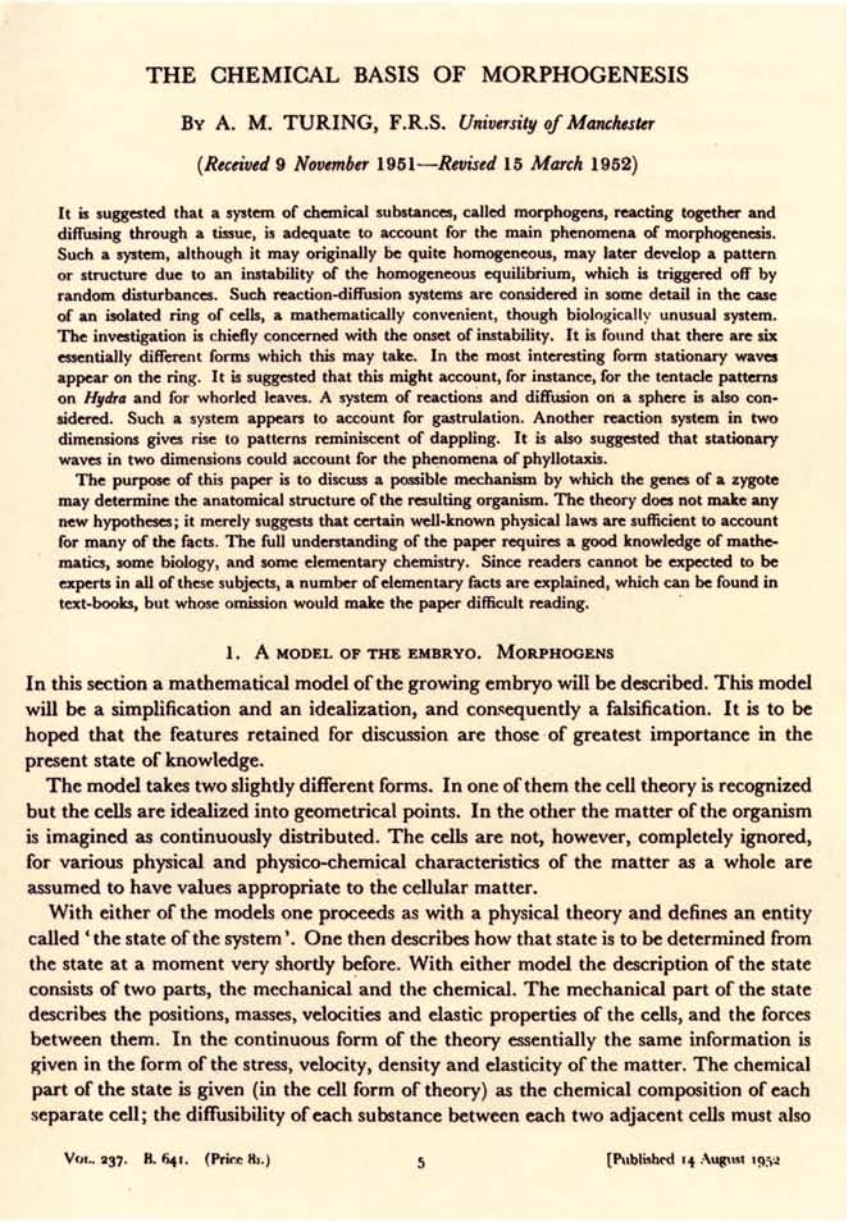
Pour tenter de comprendre la genèse des formes de la nature, Turing a aussi observé les plantes. Il s'intéresse à la phyllotaxie, laquelle consiste en l'étude de la disposition respective des parties des plantes. Dès le XIX^{e} siècle, les savants avaient noté que le cœur d'une fleur est composé d'étamines disposées sur le principe d'une double spirale. Ces spirales ont des propriétés particulières : ce qui préside à leur construction respecte le

Les graines de tournesol sur une inflorescence s'agencent en spirales très régulières (les unes dans un sens, les autres en sens inverse) que l'on peut compter. À chaque fois, on obtient deux nombres consécutifs de la suite de Fibonacci, une suite que l'on construit en partant de 0 et 1, et où tout nouveau terme est la somme des deux précédents (0, 1, 1, 2, 3, 5, 8, 13, 21, 35, 55, 89…). Turing étudia l'organisation des étamines du tournesol, comme en témoigne le dessin ci-dessous, où il recopia leur position et les numérota.

Photo Disc/ Belin - Pourlascience



Library and Archive Center, King's College Cambridge, AMT/C/25 © P. N. Furbank



THE CHEMICAL BASIS OF MORPHOGENESIS

By A. M. TURING, F.R.S. *University of Manchester*

(*Received* 9 *November* 1951—*Revised* 15 *March* 1952)

It is suggested that a system of chemical substances, called morphogens, reacting together and diffusing through a tissue, is adequate to account for the main phenomena of morphogenesis. Such a system, although it may originally be quite homogeneous, may later develop a pattern or structure due to an instability of the homogeneous equilibrium, which is triggered off by random disturbances. Such reaction-diffusion systems are considered in some detail in the case of an isolated ring of cells, a mathematically convenient, though biologically unusual system. The investigation is chiefly concerned with the onset of instability. It is found that there are six essentially different forms which this may take. In the most interesting form stationary waves appear on the ring. It is suggested that this might account, for instance, for the tentacle patterns on *Hydra* and for whorled leaves. A system of reactions and diffusion on a sphere is also considered. Such a system appears to account for gastrulation. Another reaction system in two dimensions gives rise to patterns reminiscent of dappling. It is also suggested that stationary waves in two dimensions could account for the phenomena of phyllotaxis.

The purpose of this paper is to discuss a possible mechanism by which the genes of a zygote may determine the anatomical structure of the resulting organism. The theory does not make any new hypotheses; it merely suggests that certain well-known physical laws are sufficient to account for many of the facts. The full understanding of the paper requires a good knowledge of mathematics, some biology, and some elementary chemistry. Since readers cannot be expected to be experts in all of these subjects, a number of elementary facts are explained, which can be found in text-books, but whose omission would make the paper difficult reading.

1. A MODEL OF THE EMBRYO. MORPHOGENS

In this section a mathematical model of the growing embryo will be described. This model will be a simplification and an idealization, and consequently a falsification. It is to be hoped that the features retained for discussion are those of greatest importance in the present state of knowledge.

The model takes two slightly different forms. In one of them the cell theory is recognized but the cells are idealized into geometrical points. In the other the matter of the organism is imagined as continuously distributed. The cells are not, however, completely ignored, for various physical and physico-chemical characteristics of the matter as a whole are assumed to have values appropriate to the cellular matter.

With either of the models one proceeds as with a physical theory and defines an entity called 'the state of the system'. One then describes how that state is to be determined from the state at a moment very shortly before. With either model the description of the state consists of two parts, the mechanical and the chemical. The mechanical part of the state describes the positions, masses, velocities and elastic properties of the cells, and the forces between them. In the continuous form of the theory essentially the same information is given in the form of the stress, velocity, density and elasticity of the matter. The chemical part of the state is given (in the cell form of theory) as the chemical composition of each separate cell; the diffusibility of each substance between each two adjacent cells must also

Vol. 237. B. 641. (Price 8s.) 5 [Published 14 August 1952

L'article publié par Turing en 1952 sur son système de réaction-diffusion qui sera nommé ensuite « structure de Turing ».

développement d'une suite mathématique bien connue, la suite dite de Fibonacci. C'est une suite de nombres entiers tels que chaque nombre est la somme des deux précédents. En numérotant les points de chaque spirale mise à plat et en traçant un trait entre les points selon un angle constant, on retrouve, entre les nombres représentant les points, le développement de la suite de Fibonacci : 1, 1, 2, 3, 5, 8, etc. *(voir la figure page 107).*

De même, les feuilles des plantes se développent sur les tiges selon des angles formant une spirale de pas constant pour une espèce donnée ; presque toutes les spirales appartiennent à la suite de Fibonacci. La suite de Fibonacci prédit donc l'angle de répartition des feuilles sur les tiges ou des étamines dans les fleurs sans que l'on comprenne le mécanisme sous-jacent. Turing espère éclaircir ce mécanisme en appliquant son modèle de réaction-diffusion.

En effet, la simple description géométrique du phénomène ou son expression arithmétique ne donnent aucun résultat.

Turing travaille à cette question après 1952, en collaboration avec son collègue de Manchester C. W. Wardlaw et son étudiant B. Richards, mais la plupart de ses recherches restent à l'état d'ébauche. Elles manifestent néanmoins le même souci que l'article publié en 1952 : voir dans quelle mesure, en l'absence de solutions générales, il est possible de trouver des solutions partielles grâce au modèle de réaction-diffusion, au prix d'hypothèses fortes quant à la prédictibilité du phénomène étudié.

On comprend ainsi la place accordée à l'hypothèse prédictive dans les travaux de Turing et, en particulier, la parenté entre l'article de logique de 1936 et l'article de biologie de 1952 : elle vient du fait que, dans les deux cas, le déterminisme prédictif est *local*, tandis que le comportement global du système (machine de Turing ou système physique) est non prédictif. Les deux perspectives diffèrent seulement par la façon dont nous apparaît cette distinction local/global : dans le cas de la machine de Turing, la différence local/global s'exprime à travers un théorème d'impossibilité logique traduit sous la forme du problème de l'arrêt. Ce théorème montre qu'il existe une classe de problèmes qui ne peuvent être résolus par la machine universelle de Turing. En revanche, dans le cas biologique, le caractère non prédictif tient à la croissance trop rapide des fonctions qui décrivent le système.

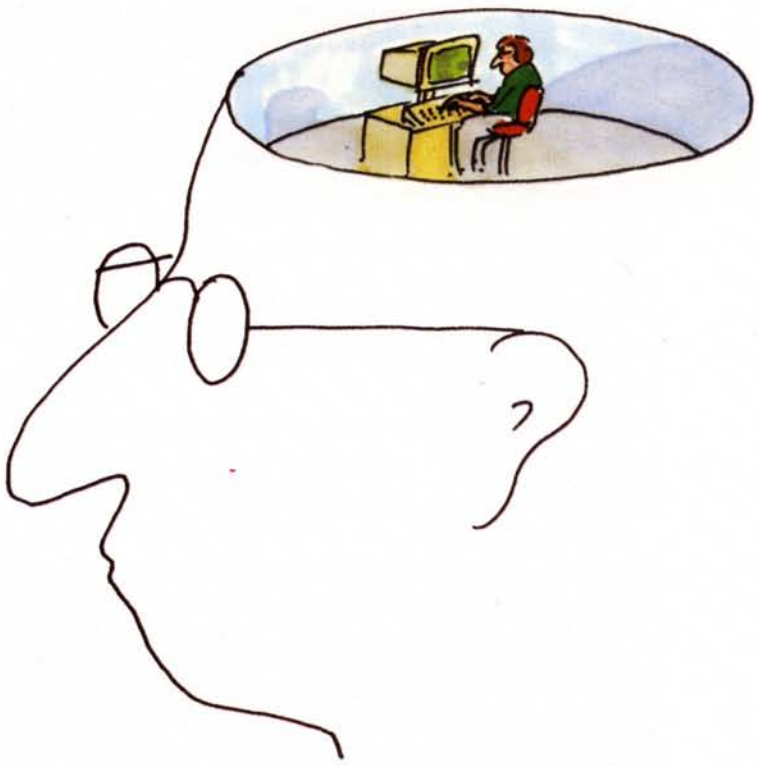
Ce point de vue théorique de Turing est des plus subtils, car il rapproche deux objets – l'auto-organisation d'une forme et la machine de Turing – que tout semble séparer à première vue, à commencer par leur nature. L'auto-organisation d'une forme possède, près de l'équilibre, un caractère de nécessité, tandis que la machine de Turing n'est qu'une notion, dont l'aspect arbitraire n'est compensé que par sa parenté avec d'autres notions remplissant la même fonction au regard de la calculabilité : le lambda-calcul de Church *(voir page 75)* et les fonctions calculables de Gödel *(voir page 70).*

La nature de la pensée

Pour quelles raisons Turing a-t-il développé un tel point de vue ? Pourquoi a-t-il tenu à se maintenir dans une posture duale à l'égard du déterminisme ? Turing s'en explique dans le seul article philosophique qu'il ait jamais écrit : *Computing machinery and intelligence*. Cet article, paru en 1950 dans la revue philosophique *Mind*, porte sur la nature de la pensée. Turing y *superpose* les deux phases du déterminisme. Loin d'être la défense et l'illustration du déterminisme prédictif dans la pensée, comme ses premiers lecteurs l'ont cru, l'article

articule au contraire avec beaucoup de subtilité deux projets dont un laisse la part belle au non prédictif : un projet « grand public » où le déterminisme prédictif semble être revendiqué, et un projet plus caché qui, tout en montrant les limites du premier, donne de la pensée une tout autre idée, celle d'un processus irréversible et non prédictif. La discrétion du second point de vue est due au fait que Turing ne le revendique pas : il affiche un point de vue déterministe sur la nature de la pensée, mais convoque toutes les ressources figuratives de la langue naturelle – jeux de mots, allusions littéraires, souvenirs, proverbes – pour en faire surgir son second point de vue. C'est ainsi, en quelque sorte, l'histoire de son propre itinéraire intellectuel que Turing raconte ici.

L'article de 1950 se présente comme une expérience de pensée dont le simple déroulement doit convaincre le lecteur que l'intelligence est un concept *déterministe et prédictif* ; le lecteur en conclut alors que l'intelligence s'applique aussi bien aux humains qu'aux ordinateurs, si ces derniers sont convenablement programmés. Et puisque le statut déterministe et prédictif rend ce concept *abstrait* applicable à des substrats divers (être humain, ordinateur), une science mécanique de l'intelligence devient alors concevable.



Peut-on faire de l'intelligence un concept abstrait, dissocié de tout substrat physique ? Telle est l'une des questions qui occupe Turing à la fin de sa vie.

Le jeu de l'imitation

L'article de Turing apparaît comme un exercice rhétorique : il vise à emporter la conviction du lecteur quant à la possibilité d'une *science mécanique de l'intelligence*. Cet exercice rhétorique prend l'aspect d'un *jeu* impliquant une prise de décision. De ce point de vue, il s'intègre parfaitement au questionnement général sur le statut du prédictif cher à Turing depuis son article de 1936. Cependant, contrairement au cas formel où la décision concernait le vrai et le faux, le jeu porte sur la différence *physique* maximale existant entre les êtres humains, à savoir la différence homme/femme.

Turing appelle le jeu qu'il imagine le « jeu de l'imitation ». Celui-ci se divise en deux phases successives. La première phase se joue entre un homme, une femme et un interrogateur séparé physiquement des deux autres joueurs. L'interrogateur pose des questions à l'homme et à la femme par l'intermédiaire d'un dispositif relevant de l'écrit (télétype ou imprimante, par exemple). Chacun doit essayer de dissimuler son sexe à l'interrogateur en imitant les réponses que donnerait l'adversaire. La deuxième phase débute quand, à l'insu de l'interrogateur, on remplace l'homme par un ordinateur programmé pour dissimuler à l'interrogateur « qui » il est : l'ordinateur est programmé pour imiter les réponses que donnerait l'homme (qui lui-même imite les réponses de la femme).

Devant l'échec de l'interrogateur à reconnaître le subterfuge après une durée de partie fixée à l'avance, le lecteur doit conclure que la dissimulation est efficace et que l'interrogateur ne parviendra jamais à deviner le sexe des joueurs. En d'autres termes, il doit se persuader que l'ordinateur est capable de *remplacer* l'être humain dans cette expérience de discrimination de l'homme et de la femme sans que sa présence soit décelable par un être humain. Convaincu de ce succès, le lecteur doit alors en déduire que, dans le futur, toutes les tâches qui requièrent une intelligence et qui étaient jusqu'à présent exécutées par un être humain pourront être remplacées par des ordinateurs au fur et à mesure des progrès de la programmation.

Le jeu vise donc deux buts : d'une part, à long terme, prouver que la notion d'intelligence est indépendante de tout substrat physique et qu'elle est de ce fait susceptible d'être incarnée dans les matériaux les plus divers, y compris les ordinateurs ; d'autre part, à court terme (même s'il ne s'agit que d'une expérience de pensée), montrer qu'un ordinateur convenablement programmé est capable de remplacer l'un des joueurs humains dans le jeu. Pourtant, en examinant le déroulement d'une partie, on conclut que *ces deux buts sont inaccessibles par le biais du jeu lui-même* et qu'il y a donc autre chose dans le jeu que la mise en place d'une science possible de l'intelligence.

Trois constatations conduisent à cette conclusion. D'abord, dans le cas de la première phase du jeu, Turing précise que la meilleure stratégie pour la femme « est sans doute de dire la vérité ». Or deux

Le jeu de l'imitation

« *Il se joue à trois, un homme* (A), *une femme* (B) *et un interrogateur* (C), *qui peut être de l'un ou l'autre sexe. L'interrogateur demeure dans une pièce différente de celle des deux autres joueurs. Le but du jeu, pour l'interrogateur, est de déterminer lequel des deux est l'homme et lequel la femme. Il les connaît sous les appellations* X *et* Y *et à la fin de la partie, il doit dire soit : "* X *est* A *et* Y *est* B *", soit : "* X *est* B *et* Y *est* A *". L'interrogateur a le droit de poser à* A *et* B *des questions telles que : "* X *pourrait-il ou pourrait-elle, s'il vous plaît, me dire la longueur de ses cheveux ? "*

Supposons que X *est vraiment* A *et qu'il lui faut donner une réponse. Le but de* A *dans le jeu est d'induire* C *en erreur. Sa réponse pourrait donc être : " Mes cheveux sont coupés à la garçonne et les mèches les plus longues font à peu près 20 centimètres. "*

Pour faire en sorte que les tons de voix ne viennent pas en aide à l'interrogateur, les réponses devraient être écrites, ou mieux encore, dactylographiées. La configuration idéale serait de disposer d'une téléimprimante communiquant à travers deux pièces. On peut aussi concevoir que questions et réponses soient répétées par un intermédiaire. Le but du jeu pour le troisième joueur (B) *est de venir en aide à l'interrogateur. La meilleure stratégie pour celle-ci est sans doute de donner des réponses vraies. Elle peut ajouter des remarques à ses réponses comme " je suis la femme, ne l'écoutez pas ! ", mais cela n'aboutirait à rien, car l'homme peut faire des remarques semblables.*

Nous posons maintenant la question : " Que se passera-t-il si l'on substitue une machine à A dans le jeu ? " L'interrogateur se trompera-t-il autant de fois quand le jeu est joué de cette manière que lorsqu'il est joué entre un homme et une femme ? Ces questions remplacent la question originelle, " Les machines peuvent-elles penser ? ". »

Alan Turing, Computing machinery and intelligence, *1950*

points méritent d'être soulignés à ce sujet : d'une part, cette stratégie a pour conséquence immédiate l'élimination de la femme dans le jeu, puisque dire toujours la vérité est une stratégie trop univoque pour échapper longtemps à la perspicacité de l'interrogateur ; d'autre part, attribuer une telle stratégie à la femme n'a aucun caractère de nécessité, car elle pourrait aussi bien échoir à l'homme. L'attribuer *systématiquement* à la femme a donc une autre fonction que celle d'être la « meilleure stratégie ».

Cette fonction repose sur une analogie induite dans le jeu entre la vérité-authenticité et la femme d'une part, et le mensonge-dissimulation et l'homme d'autre part. La question qu'une telle attitude soulève est alors la suivante : sur quel fondement repose l'analogie tracée par Turing entre la vérité et le mensonge d'un côté, et la femme et l'homme de l'autre ? Force est de constater que cette analogie est *illégitime* et qu'elle laisse penser que l'élimination de la femme a d'autres motivations que celles explicitées dans le jeu.

La deuxième constatation a trait à la deuxième phase du jeu : la conclusion qui doit être atteinte consiste à considérer que la notion d'intelligence est indépendante de tout substrat physique. Or pour l'établir, il faut décider, à un moment donné, que l'interrogateur ne parviendra *jamais* à faire la différence entre l'homme et la femme et qu'il est donc temps de remplacer l'homme par un ordinateur : c'est cette décision qui permet de faire passer la partie de la première à la seconde phase. Or *ce moment de décision ne peut pas être déterminé temporellement* parce qu'il est toujours possible que l'interrogateur pose la question qui lui révélera l'identité sexuelle d'un joueur.

Passer à la seconde phase de la partie implique donc d'être *déjà convaincu* de l'échec de l'interrogateur et du caractère indécidable du remplacement de l'homme par l'ordinateur, *avant même* le début

http://www.sciwrite.caltech.edu/journal03/tura.html

d'une partie. Les arguments qui ont conduit à cette conviction ne dérivent donc pas du déroulement d'une partie : il y a de toute évidence ici une *pétition de principe* qui repose sur des motivations d'un tout autre ordre que celles défendues ouvertement par Turing dans l'article.

En résumé, le jeu tel qu'il est décrit conduit, lors de sa première phase, à l'élimination de la femme et, lors de sa seconde, au préjugé de l'indépendance de la notion d'intelligence à l'égard de tout substrat physique particulier.

Argument de la continuité du système nerveux

Le système nerveux n'est sûrement pas une machine à états discrets. Une petite erreur concernant la taille de l'influx nerveux entrant dans un neurone peut avoir une grande conséquence sur la taille de l'influx qui en sort. On pourrait arguer, cela étant, que l'on ne peut pas imiter le comportement du système nerveux avec un système à états discrets. Il est vrai qu'une machine à états discrets doit être différente d'une machine continue. Mais si nous acceptons les condictions du jeu de l'imitation, l'interrogateur ne pourra pas tirer avantage de cette différence.

Alan Turin, *Computing machinery and intelligence*, 1950

Peut-on distinguer un homme et un ordinateur ?

La troisième constatation concerne l'enchaînement des deux phases du jeu. Cet enchaînement vise à supprimer la pertinence de tout substrat physique particulier eu égard à la notion d'intelligence : d'une part entre les hommes et les femmes, d'autre part entre les humains et les ordinateurs. Toutefois, cette conclusion dépend d'un point de vue *imaginaire* dans lequel on ne peut jamais se placer en réalité, car il exige *a priori* du lecteur à la fois de reconnaître la différence *physique* entre l'humain et l'ordinateur et de ne pas reconnaître cette différence. En effet, pour que le jeu de l'imitation puisse atteindre son but déclaré (séparer l'intelligence de tout substrat physique), il faudrait que chaque humain se place de lui-même en position d'interrogateur mis en échec, *tout en étant capable* de faire *physiquement* la différence entre humain et ordinateur (en tant qu'observateur du déroulement du jeu).

En d'autres termes, l'article de 1950 invite le lecteur à considérer que la différence physique entre humain et ordinateur est en même temps non pertinente et pertinente, selon qu'il se place en imagination à l'intérieur de la construction du jeu ou à l'extérieur : à l'intérieur du jeu, le lecteur s'identifie à l'interrogateur mis en échec et ne voit donc plus la différence entre humain et ordinateur ; à l'extérieur, la différence physique entre humain et ordinateur existe (c'est une donnée du jeu à laquelle le a accès en tant qu'observateur extérieur).

C'est la possibilité de ce va et vient entre intérieur et extérieur du jeu – autrement dit cet *indécidable quant à la différence physique* entre humain et ordinateur – qui n'est jamais explicitée par Turing. En outre, rétroactivement, si cet indécidable instaure une différence physique inassignable entre être humain et ordinateur, il doit le faire aussi entre homme et femme de par la construction du jeu. *La différence sexuelle n'est donc pas abolie dans le jeu : elle y survit sous la forme d'un indécidable.*

Le déroulement du jeu, dans la mesure où il fait intervenir de façon dissimulée un indécidable quant à la nature physique des joueurs, impose une double conclusion. D'une part, du point de vue du projet scientifique que le jeu est censé soutenir, on doit conclure que les *manifestations de l'intelligence humaine ne relèvent pas toutes du déterminisme prédictif* : un être humain (l'interrogateur) ne peut décider si les réponses qu'il obtient proviennent d'un homme, d'une femme ou d'une machine. D'autre part, il est possible de préciser ce sur quoi porte l'indécidable qui intervient dans le jeu : il s'agit de la différence physique maximale entre deux êtres parlants, c'est-à-dire de la différence sexuelle telle qu'elle est décelable à partir de leur comportement verbal rédigé sous forme écrite.

Le projet d'une science mécanique de l'intelligence bute donc sur la façon dont est interprétée la différence sexuelle dans le cadre du jeu. Un tel projet présuppose qu'un surcroît de programmation pourra toujours venir à bout de la différence sexuelle représentée verbalement ; or la présence d'un indécidable dans le jeu montre au contraire que les interactions entre les humains laissent toujours une trace physique par l'intermédiaire de leur comportement verbal. C'est donc *l'attitude à l'égard du verbal* qui conditionne l'une ou l'autre branche de l'alternative.

La part du diable

Turing a le projet déclaré, dans l'article de 1950, de montrer que le cheminement qui a été le sien pour en venir à penser la possibilité d'une science mécanique de l'intelligence peut être exécuté par tout lecteur. Le jeu de l'imitation est donc censé être le moyen qui permet au lecteur de l'article d'opérer un raccourci temporel. Il parviendrait alors, dès qu'il aurait compris le fonctionnement du jeu, au point où est arrivé Turing, sans avoir besoin de suivre son cheminement.

Cependant, à moins de supposer que la réalisation d'une science mécanique de l'intelligence ait été prévisible de toute éternité, il faut bien admettre que son apparition à un moment donné est *contingente*. Dès lors, Turing se sent contraint, pour convaincre son lecteur, de faire intervenir dans son article l'« ingrédient » qui justifie cette apparition, son récit personnel. Ce récit personnel est contenu dans les objections au projet d'une science mécanique de l'intelligence, objections que Turing expose, puis auxquelles il répond en mêlant de façon dissimulée des souvenirs personnels à ses réflexions.

Library and Archive Center, King's College Cambridge, AMT/K/7/38

Alan Turing dans les années 1940.

Ces réponses dépassent le cadre du style d'écriture scientifique et rendent accessible une tout autre dimension du rapport au verbal, faite de récits et de métaphores. Par leur aspect figuratif, elles dévoilent, dans la constitution de sa pensée, une *contingence* inaccessible à la programmation. Les ressources de la langue naturelle nous permettent alors de saisir les *événements constitutifs* qui ont participé à l'organisation personnelle de la pensée de Turing et qui dessinent son histoire individuelle, en particulier cette façon de mêler l'invention de l'ordinateur à sa verbalisation du rapport qu'il entretient avec la différence sexuelle. Dès lors, l'invention de l'ordinateur devient plus le résultat du parcours culturel d'un individu au sein d'une société que l'apparition, dans un agenda scientifique, d'un outil abstrait que l'on n'aurait pas besoin de s'approprier culturellement.

Sans prendre en considération tous les exemples, il est possible d'en donner deux qui éclaircissent la question. À la fin de la liste d'objections que Turing présente contre la thèse de la possibilité d'une science mécanique de l'intelligence, Turing use d'un argument au premier abord tout à fait étrange, « l'argument de la perception extra-sensorielle » : l'esprit ne peut pas être mécanisé car on ne peut pas mécaniser la perception extra-sensorielle. Que vient faire un argument pareil dans un texte épistémologique ? D'aucuns ont invoqué le caractère ironique de l'argument dans un texte rempli… de perceptions extra-sensorielles. D'autres y voient une image dont on peut tirer un véritable projet scientifique en termes d'interfaces cerveaux-machines. Ce sont des lectures possibles. Après tout, Turing ne semble pas hermétique à l'idée de télépathie : dans un article de 1953 où il raconte comment des vieilles dames de la *Société pour la recherche psychique* ont cherché à influencer un ordinateur qui jouait au jeu de Nim, il conclut que « les machines sont beaucoup moins coopératives que les être humains en matière d'influence télépathique » *(voir page 45)*.

On pourrait s'en tenir là. Mais on peut aussi remarquer que *la carrière scientifique de Turing a eu une expérience « télépathique » pour déclencheur.* Lors de ses années de lycée passées à l'internat de la *Sherborne Grammar School*, Turing eut un amour platonique envers l'un de ses camarades, Christopher Morcom, et leur terrain d'entente commun fut la science. Christopher Morcom, brillant élève promis à un grand avenir scientifique, fut reçu à l'examen d'entrée à Cambridge alors que Turing échoua et dut attendre un an pour le repasser. Entretemps, Christopher Morcom mourut d'une tuberculose bovine. Turing se sentit alors investi d'une mission : assumer le destin scientifique de son ami disparu. Il écrivit un texte intitulé *Nature de l'Esprit* qu'il envoya à la mère de son camarade décédé. Dans ce texte, il décrit le mécanisme qui retient l'esprit au corps jusqu'à la mort et la façon dont l'esprit détaché du corps « trouve tôt ou tard un autre corps, peut-être immédiatement ». Il se demande alors pourquoi les êtres humains ont des corps qui les empêchent de « vivre libres comme des esprits et de communiquer comme tels ». Ainsi, le décès de son camarade a probablement influencé la manière dont Turing a posé les questions qui ont jalonné son parcours intellectuel.

L'« apprentissage des machines »

Quelques pages plus loin, Turing s'interroge sur la façon dont les machines peuvent apprendre. Le thème de l'« apprentissage des machines » aura un rôle crucial par la suite dans ce qu'il est convenu d'appeler aujourd'hui « l'intelligence artificielle » et certains voient donc dans les remarques de Turing l'acte de naissance d'un thème scientifique promis à un grand avenir. On peut aussi être sensible à la façon dont Turing présente le thème en question. Il soulève tout d'abord une question d'ordre technique : quel type d'ingénierie doit servir à la construction des ordinateurs pour qu'ils soient susceptibles de participer au jeu de l'imitation ? On s'attend à ce qu'il réponde à la question en mathématicien ou en

ingénieur, mais il fait tout d'abord une étrange remarque qui semble, elle aussi, relever de l'ironie pure et simple :

On pourrait par exemple insister sur le fait que l'équipe d'ingénieurs devrait être toute du même sexe, mais ce ne serait pas vraiment satisfaisant, car il est sans doute possible de construire un individu complet à partir d'une seule cellule, disons de la peau d'un homme.

L'ironie vient du fait que la remarque vise à supprimer la possibilité d'une tricherie de la part de l'équipe d'ingénieurs, tricherie qui consisterait à faire passer pour une création artificielle une « machine » qui aurait été en fait obtenue par une fécondation et une gestation naturelles. En formant une équipe d'ingénieurs du même sexe, on supprime la différence sexuelle, ce qui va dans le sens du projet du jeu de l'imitation. Toutefois, on laisse le soin à la peau de jouer le rôle d'une création par parthénogenèse. Ce faisant, on construit une machine qui a les attributs de l'homme : l'homme en tant qu'opposé à la femme reste fantomatiquement présent.

On trouve dans l'article de 1950 de nombreuses autres allusions cachées à la vie de Turing : ces allusions permettent d'interpréter tout autrement le projet d'une science mécanique de l'esprit en l'intégrant à un parcours individuel dont le propre est d'être globalement imprévisible et dont la langue naturelle permet, par ses ressources propres, de reconstituer les étapes marquantes. Remarquons cependant que le jeu de l'imitation n'a pas seulement pour fonction de reconstituer les étapes du passé individuel de Turing : il préfigure à bien des égards la fin tragique qui sera celle de Turing quatre ans après la rédaction de l'article.

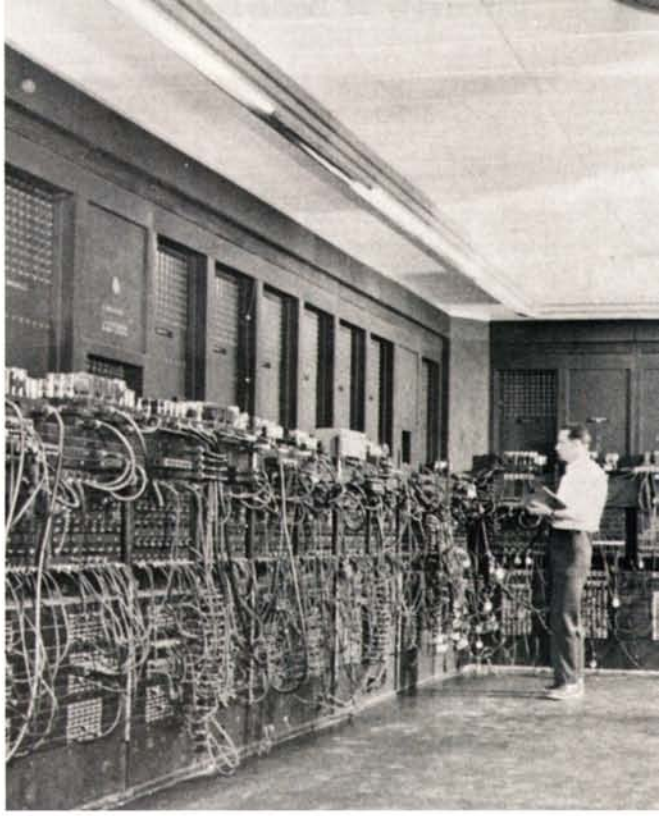
Wikimedia Commons

Une statue de Turing, une pomme à la main, érigée en 2001 à Manchester, dans le Sackville Park. Ci-dessous, une plaque commémorative posée sur la dernière demeure de Turing, à Wilmslow, en 2004.

Le suicide de Turing

Turing était homosexuel et ne s'en cachait pas, au moins dans le cercle libéral de Cambridge, même si la réserve était de mise ailleurs. En décembre 1951, il rencontre dans les rues de Manchester un jeune homosexuel, Arnold Murray, avec qui il a une aventure. Peu après, un vol d'argent est commis chez Turing par un tiers qui a eu vent par Murray de la disposition des lieux. Turing, naïvement, déclare le vol à la police qui reconstitue assez vite toute l'histoire. Murray et Turing sont condamnés pour homosexualité en mars 1952. On laisse à Turing, *fellow* de la *Royal Society*, le choix entre la prison et un traitement hormonal censé prévenir son homosexualité. Turing, pour des raisons liées à son travail à Manchester qu'il ne veut pas abandonner, décide de subir le traitement hormonal consistant en des injections d'hormones femelles censées faire baisser sa libido. Le traitement lui est administré d'avril 1952 à avril 1953 ; il devient temporairement impuissant et ses seins se mettent pousser. Le 7 juin 1954, Turing se suicide chez lui en ingérant une pomme ayant macéré dans du cyanure, selon le schéma de l'empoisonnement de Blanche-Neige dans le dessin animé de Walt Disney qui l'avait tant marqué avant-guerre.

Revenons un instant au jeu de l'imitation. Sa problématique est basée sur le fait de savoir s'il est possible de considérer la notion d'intelligence indépendamment de tout substrat physique. La réponse « publique » de Turing est positive, mais nous avons vu qu'elle nécessite de refouler dans l'indécidable certaines conditions du jeu. Ces conditions, qui ont à voir avec la nature physique des joueurs, sont celles qui rattrapent définitivement Turing le 7 juin 1954 : son traitement hormonal a fait de lui un homme qui ne peut plus décider s'il est un homme ou une femme. Il est dès lors difficile de ne pas voir dans les conditions du jeu de l'imitation non seulement le récit d'une histoire individuelle, mais la préfiguration de sa fin tragique. ■



Étrange destin que celui de l'œuvre de Turing (1912-1954) : jaugée à l'aune de la conception et de la réalisation des premiers ordinateurs, elle fut longtemps occultée sur ce terrain par la figure massive de von Neumann. Ce n'est qu'à la faveur du travail d'historien mené à bien par Andrew Hodges au début des années 1980 que la connaissance de son œuvre quitta le cercle étroit des spécialistes et réoccupa la place capitale qui lui est due dans l'histoire de la notion de calcul, dans celle des machines à calculer et, plus généralement, dans notre société informatique. Mais cette reconnaissance posthume, tout d'abord suscitée par l'engouement pour l'« intelligence artificielle » qui se voulait être, depuis les années 1960, l'héritière et la continuatrice de son œuvre, a également nuit à son intelligibilité : en s'appropriant la figure de Turing pour le camper en père fondateur, elle occulta les parties de ses travaux qui n'avaient pas directement à voir avec la construction des premiers ordinateurs ni avec les modèles informatiques des fonctions cognitives.

Turing ou l'expérience

Lorsque Turing conçoit sa machine, explique certaines formes du vivant ou définit la notion d'intelligence, une seule question le guide : quel est le périmètre de calculabilité du domaine considéré ?

Library and Archive Center, King's College Cambridge, AMT/K/7/19 © Walton Athletic Club

Alan Turing (à gauche) *se rendant en 1946 à une rencontre sportive avec d'autres membres du* Walton Athletic Club, *un club amateur d'athlétisme situé à Walton, dans le Surrey. Page ci-contre, l'*ENIAC, *un des premiers ordinateurs réalisés à partir du concept de machine universelle de Turing.*

des limites

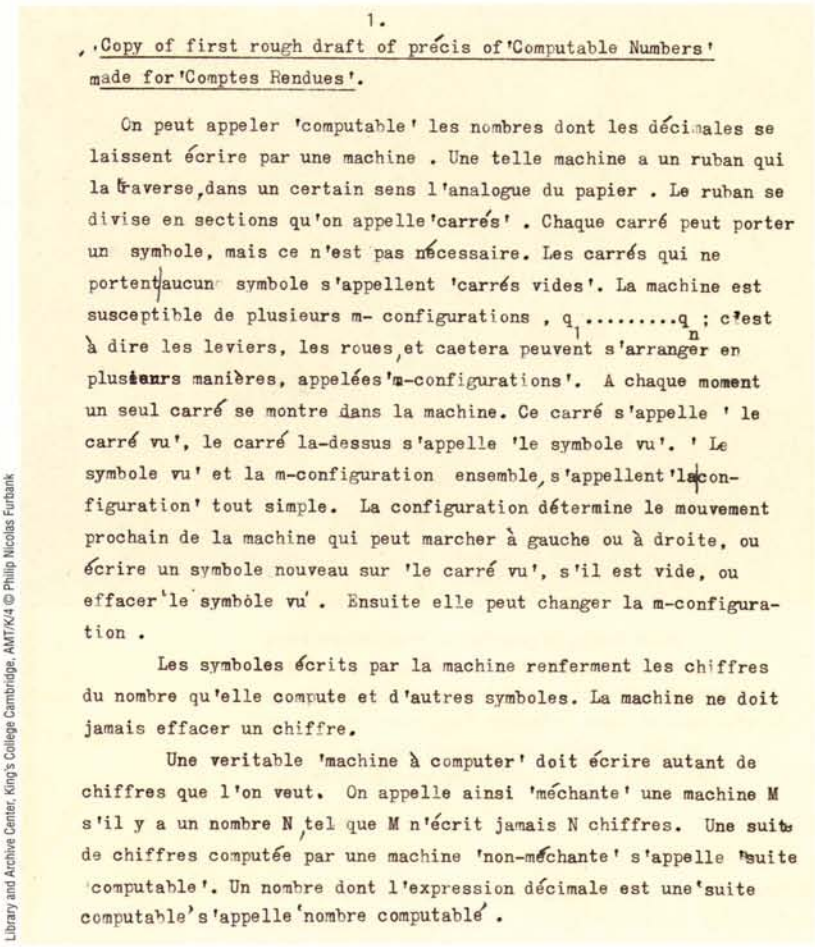
L'œuvre de Turing est en effet beaucoup plus diverse que ce que l'on en retient généralement. Elle est aussi beaucoup plus *romanesque* : écrivains et hommes de théâtre ne s'y sont pas trompés, eux qui ont reconnu très vite tout le parti qu'ils pouvaient tirer d'un itinéraire de vie aussi extraordinaire, tant du point de vue intellectuel que du point de vue des actions collectives auxquelles il participa. Qu'on en juge plutôt. Son parcours intellectuel a quelque chose d'un météore : en à peine 20 ans (son premier article date de 1935, son dernier de 1954), Turing obtient des résultats novateurs ou même révolutionnaires dans des domaines aussi divers que les mathématiques (calcul des probabilités, théorie des nombres, théorie des groupes), la logique (décidabilité, calculabilité), la construction des premiers ordinateurs ou la morphogenèse biologique.

Plus encore, dans l'urgence absolue de la lutte contre le nazisme, à un des moments les plus dramatiques de la Seconde Guerre mondiale où la Grande-Bretagne, isolée par un blocus sans précédent, résiste encore, Turing se distingue à nouveau : au départ quasi seul, il décode les messages cryptés envoyés de Berlin *via* Paris occupé aux sous-marins de la *Kriegsmarine* patrouillant dans l'Atlantique à la recherche des convois de ravitaillement alliés à couler. Il est, de ce point de vue, l'une des grandes figures de l'ombre du second conflit mondial et a puissamment contribué à préserver la Grande-Bretagne d'une invasion nazie. Enfin, sa condamnation pour homosexualité, qui a très certainement contribué à son suicide à l'âge de 42 ans, en 1954, nous paraît, avec le recul, non seulement dénuée de justice mais quasi incompréhensible.

La vie de Turing inspira nombre d'écrivains et hommes de théâtre. Ci-contre, les affiches de spectacles mis en scène par Jean-François Peyret : « Turing Machine » en 1999 à la MC93 de Bobigny et « Histoire naturelle de l'esprit (suite et fin) » en 2000 au Palais de Chaillot à Paris.

Il y a donc bien matière à *récit* dans cette vie énigmatique et emblématique de notre civilisation technologique où le « numérique » prend une place sans cesse croissante. Cependant, une première constatation s'impose : dans cette énumération rapide des résultats de Turing, il n'est nulle part question d'« intelligence artificielle », pas plus que de modélisation informatique des fonctions cognitives. Sa réputation serait-elle donc usurpée ? Pas du tout. Mais elle ne se situe pas seulement là où on la cantonne quand on privilégie ses contributions à la théorie de la calculabilité ou à la construction des premiers ordinateurs sans prendre en considération la totalité de l'itinéraire complexe de Turing. Quelle clé faut-il alors employer pour en décrypter la cohérence ?

La question qui a occupé Turing toute sa vie concerne le *périmètre de la calculabilité*, c'est-à-dire ce qui, dans tel ou tel domaine, est susceptible ou non de recevoir une détermination *numérique*, au sens traditionnel de rapport entre nombres. On pourrait arguer que cette question ne définit pas de façon



1.

Copy of first rough draft of précis of 'Computable Numbers' made for 'Comptes Rendues'.

On peut appeler 'computable' les nombres dont les décimales se laissent écrire par une machine . Une telle machine a un ruban qui la traverse, dans un certain sens l'analogue du papier . Le ruban se divise en sections qu'on appelle 'carrés' . Chaque carré peut porter un symbole, mais ce n'est pas nécessaire. Les carrés qui ne portent aucun symbole s'appellent 'carrés vides'. La machine est susceptible de plusieurs m- configurations , $q_1 \ldots\ldots\ldots q_n$; c'est à dire les leviers, les roues, et caetera peuvent s'arranger en plusieurs manières, appelées 'm-configurations'. A chaque moment un seul carré se montre dans la machine. Ce carré s'appelle ' le carré vu', le carré la-dessus s'appelle 'le symbole vu'. ' Le symbole vu' et la m-configuration ensemble, s'appellent 'la configuration' tout simple. La configuration détermine le mouvement prochain de la machine qui peut marcher à gauche ou à droite, ou écrire un symbole nouveau sur 'le carré vu', s'il est vide, ou effacer 'le symbole vu' . Ensuite elle peut changer la m-configuration .

Les symboles écrits par la machine renferment les chiffres du nombre qu'elle compute et d'autres symboles. La machine ne doit jamais effacer un chiffre.

Une veritable 'machine à computer' doit écrire autant de chiffres que l'on veut. On appelle ainsi 'méchante' une machine M s'il y a un nombre N tel que M n'écrit jamais N chiffres. Une suite de chiffres computée par une machine 'non-méchante' s'appelle 'suite computable'. Un nombre dont l'expression décimale est une 'suite computable' s'appelle 'nombre computable' .

suffisamment précise ses recherches personnelles puisque tout mathématicien se trouve confronté à elle : et en effet, déterminer numériquement un certain nombre de rapports réglés semble caractériser l'entreprise mathématique en général. En revanche, Turing est original dans la façon dont il *répond* à cette question, car sa réponse caractérise le style de détermination mathématique qu'il cherche à promouvoir.

L'attitude de Turing consiste à tracer une limite entre le calculable et le non calculable et, *d'un même mouvement*, à étendre aussi loin que possible le domaine du calculable en repoussant cette limite. Ce point de vue explique les hésitations qu'un lecteur de Turing peut avoir, tiraillé entre les deux facettes de sa démarche. Il devient alors tentant de privilégier un aspect de son œuvre plutôt qu'un autre, selon que l'on cherche à ranger celle-ci sous une bannière ou une autre, comme la théorie de l'extension indéfinie du domaine du calculable, de loin la plus répandue. Toutefois, il faut se garder de cette lecture unilatérale parce que l'extension maximale du domaine du calculable *n'a de sens que si elle est rapportée à la détermination préalable d'une frontière entre calculable et non calculable*.

Certes, cette frontière est mouvante : dans un système formel, le domaine de ce qui est accessible au calcul dépend de l'ingéniosité des mathématiciens à trouver les caractéristiques calculatoires de telle ou telle procédure. Néanmoins, elle existe bel et bien *a priori*, comme Turing l'a démontré. Il existe donc un processus de pensée, de nature énigmatique, sur lequel Turing s'interrogera toute sa vie, un processus qui dépasse la limite entre calculable et non calculable puisqu'il permet de la déterminer. Comprendre « l'expérience des limites » de Turing fait toute la difficulté et la richesse de son œuvre. Cette attitude n'est pas partagée par toute la communauté mathématique. D'autres styles de pensée la côtoient, dont certains ne privilégient en rien la dimension calculatoire de la détermination mathématique.

La limite du calculable

Trois exemples éclairent la démarche de Turing. Ces trois exemples correspondent à ses trois articles fondamentaux, écrits respectivement en 1936, 1950 et 1952. Ils représentent les trois étapes fondamentales de son itinéraire intellectuel.

Dans le premier, intitulé *On computable numbers, with an application to the* Entscheidungsproblem,

Le compte-rendu que Turing écrivit en 1935 pour l'Académie des sciences avec l'aide de sa mère, première version de son fameux article de 1936 (en haut). Ce compte-rendu passa inaperçu. Ci-contre, l'ordinateur* Deep Blue*, qui affronta Kasparov aux échecs en 1996. Le champion gagna la première manche en un coup et perdit la seconde. L'intelligence de l'ordinateur et celle de l'homme seraient-elles indiscernables lorsqu'il s'agit de jouer aux échecs, comme dans le jeu de l'imitation de Turing ?

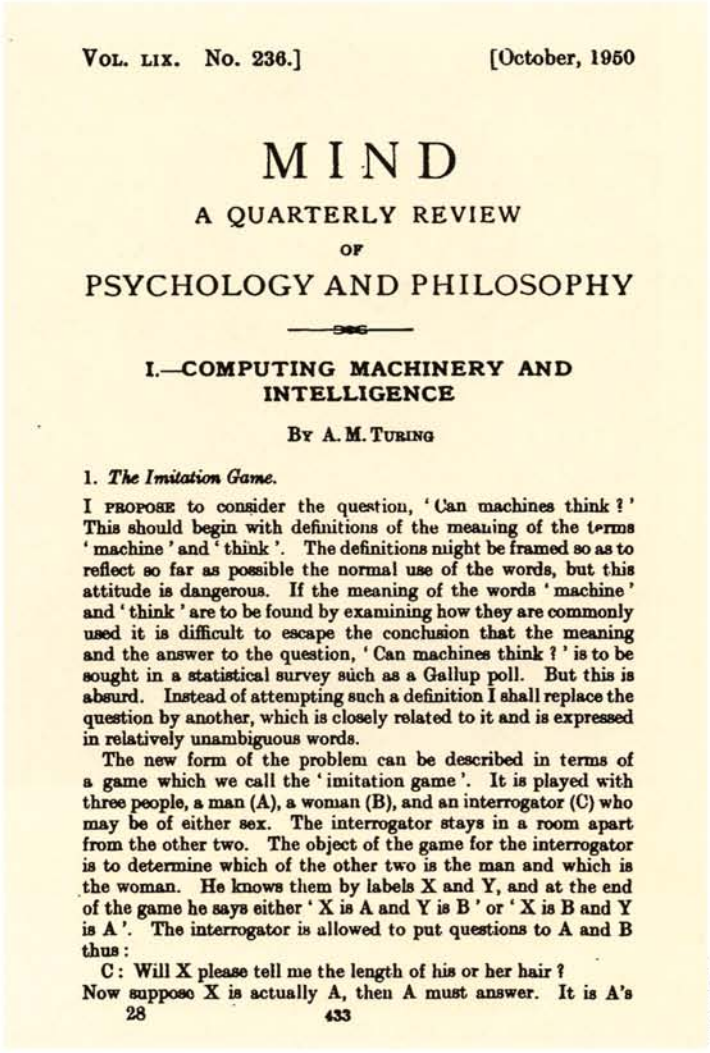
Turing, âgé de 24 ans, fonde la théorie de la calculabilité en démontrant plusieurs résultats capitaux liés à la représentation formelle des nombres. À l'aide d'un même concept mathématique, la « machine de Turing », il fixe une limite entre calculable et non calculable (et définit du même coup la calculabilité) tout en déplaçant continuellement cette limite en faveur du calculable.

Nous avons vu que sa première étape, la détermination d'une *limite* au domaine du calculable, passe par un raisonnement par l'absurde : Turing montre le caractère contradictoire d'une machine à calculer du type « machine de Turing » qui pourrait résoudre une certaine classe de problèmes en s'en tenant au domaine de l'explicitement calculable. Une fois cette contradiction atteinte, c'est-à-dire une fois explicitement exhibée la frontière entre calculable et non calculable sur une classe de problèmes servant de contre-exemple, Turing montre, dans une seconde étape, que l'on peut postuler que le concept de « machine de Turing » effectue néanmoins n'importe quelle tâche, pourvu que celle-ci soit réductible à un calcul : à charge aux mathématiciens d'opérer cette réduction sur des problèmes précis. Cette approche est particulière parce qu'elle ne se contente pas de solliciter l'esprit du lecteur en vue qu'il *comprenne* un nouveau concept ou une nouvelle démonstration. Elle le place dans une disposition d'esprit telle *qu'il cherche à étendre le domaine du calculable*. Cette *éthique du calculable* est le fond de la démarche de Turing en 1936 et ne le quittera jamais.

Lorsque l'ordinateur et l'homme sont indiscernables

Le deuxième exemple est lié au seul article que Turing écrivit pour une revue de philosophie. Dans ce texte de 1950, intitulé *Computing machinery and intelligence*, Turing s'interroge sur la nature du concept d'intelligence et se demande dans quelle mesure ce concept est ou non transférable à des machines. Pour répondre à ces questions, il propose un jeu de son invention, qu'il appelle le « jeu de l'imitation » et qui vise à montrer expérimentalement – si tant est qu'une expérience de pensée soit assimilable à une expérience – qu'il est possible de dissocier le concept d'intelligence du substrat physique particulier propre aux êtres humains. Les règles du jeu mettent en lumière une situation dans laquelle un être humain ne peut pas distinguer une tâche exécutée par un ordinateur (et donc par définition réductible à un calcul) de la même tâche exécutée par un être humain, alors que ce dernier est censé être discernable d'un ordinateur (son intelligence est censée ne pas se réduire à un calcul).

Ainsi, en isolant le concept d'intelligence de tout support physique particulier, le jeu permet d'assimiler intelligence humaine et intelligence « mécanique » : si l'intelligence humaine ne se distingue pas des formes d'intelligence ayant un tout autre support matériel,



VOL. LIX. No. 236.] [October, 1950

MIND

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

BY A. M. TURING

1. *The Imitation Game.*

I PROPOSE to consider the question, 'Can machines think?' This should begin with definitions of the meaning of the terms 'machine' and 'think'. The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous. If the meaning of the words 'machine' and 'think' are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, 'Can machines think?' is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

The new form of the problem can be described in terms of a game which we call the 'imitation game'. It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either 'X is A and Y is B' or 'X is B and Y is A'. The interrogator is allowed to put questions to A and B thus:

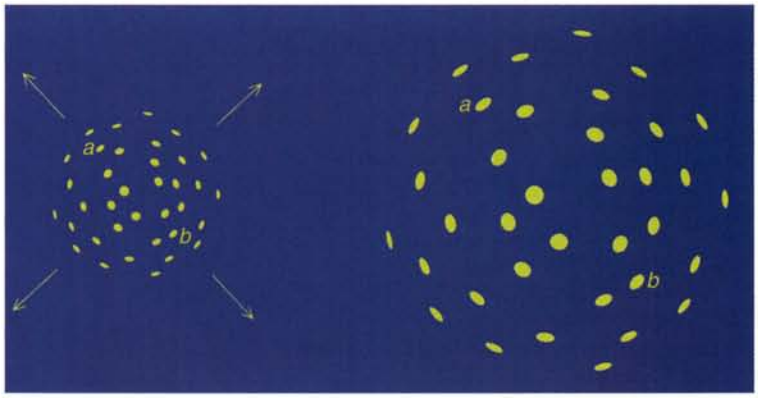
C: Will X please tell me the length of his or her hair?

Now suppose X is actually A, then A must answer. It is A's

28 433

Mind Association

L'article Computing machinery and intelligence *publié par Turing en 1950 dans la revue* Mind (ci-dessus). *Le périmètre de calculabilité défini par Turing ressemble à « l'univers-ballon » des cosmologistes* (ci-dessous), *modèle qui représente l'expansion uniforme de l'Univers (chaque point à la surface du ballon représente une galaxie). Comme les cosmologistes, Turing délimite un domaine, puis l'étend au maximum.*



H. Meinhardt

Les motifs pigmentaires qui décorent les coquilles de gastéropodes et de mollusques bivalves (en bas à gauche) sont des « structures de Turing ». Ce sont des enregistrements d'un processus dynamique se déroulant sur le bord de la coquille à mesure qu'elle croît. Aujourd'hui, des modèles et des simulations sur ordinateur reconstituent les processus de formation de ces motifs.

mathématiquement les équations de propagation de substances chimiques idéales et la compétition entre la réaction et la diffusion de ces substances s'opérant à vitesse variable. Le résultat le plus remarquable est l'apparition, dans certaines conditions transitoires, d'ondes stationnaires, appelées aujourd'hui « structures de Turing » : ce phénomène d'auto-organisation du milieu rend compte de l'apparition de certaines formes naturelles dans le processus de développement, comme les tâches et les rayures sur la peau de certains mammifères ou de certains coquillages.

Turing semble effectuer ici un virage complet dans ses recherches, car rien ne paraît rapprocher ce thème de ses travaux antérieurs. C'est d'ailleurs pourquoi toute la fin de sa carrière intellectuelle est généralement sous-estimée ou passée sous silence. Pourtant, si l'on examine ce dernier axe de recherche en gardant en tête le projet fondamental de Turing – la détermination du périmètre de la calculabilité selon les deux étapes décrites précédemment –, on s'aperçoit qu'il n'est pas aussi éloigné des précédents qu'il en a l'air : Turing cherche à montrer que *la matière elle-même* passe par ces deux étapes. Le raisonnement de nature si énigmatique qui parvient à réunir les deux faces de la notion de calcul, calculable et non calculable, a donc une *contrepartie physique*. Dans les deux premiers articles, la constitution d'une limite entre calculable et non calculable était acquise quand on exhibait un problème particulier qui n'était pas soluble au moyen du concept de machine de Turing. Dorénavant, ce n'est pas un cas particulier, mais la nature toute entière qui est non calculable : le non calculable est la règle plutôt que l'exception dans le domaine de la matière physique. L'exception réside dans l'apparition des formes individuées au sein du vivant : il est possible de calculer, au moins idéalement, les processus de différenciation qui aboutissent à cette apparition. L'extension du domaine du calculable passe donc par *l'étude de la production des formes dans la nature*.

Ainsi, les trois étapes de l'itinéraire intellectuel de Turing dessinent un parcours tout à fait original : héritier de la science classique déterministe, Turing a poussé aussi loin que possible le périmètre de sa validité. Cependant, il a aussi rompu avec ce paradigme et contribué à en engendrer un nouveau, celui de la science contemporaine : le nôtre. ■

[BIBLIOGRAPHIE >>|

Turing A., ***Collected Works of A. M. Turing 1 : Pure Mathematics***, Ed. J. L. Britton, Elsevier Science Publishers, 1992.

Turing A., ***Collected Works of A. M. Turing 2 : Mathematical Logic***, Ed. R. Gandy & C.E.M. Yates, Elsevier Science Publishers, 1995.

Turing A., ***Collected Works of A. M. Turing 3 : Mechanical Intelligence***, Ed. Darrel Ince, Elsevier Science Publishers, 1992.

Turing A., ***Collected Works of A. M. Turing 4 : Morphogenesis***, Ed. P. T. Saunders, Amsterdam, Elsevier Science Publishers, 1992.

Turing A., ***The Essential Turing : Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma***, Oxford University Press, 2004.

Turing A., ***On Computable Numbers with an Application to the Entscheidungsproblem***, Proceedings of the London Mathematical Society, 42, pp. 230-265,1936. En ligne à : http://web.comlab.ox.ac.uk/oucl/research/areas/ieg/e-library/source/tp2-ie.pdf

Turing A., ***Proposal for the Development in the Mathematics Division of an Automatic Computing Engine (ACE)***, Executive Committee National Physical Laboratory (HMSO) : 1-20, 1945. En ligne à : http://www.emula3.com/docs/Turing_Report_on_ACE.pdf

Turing A., ***Computing Machinery and Intelligence***, Mind LIX, 236 : 433-460, 1950. En ligne à : http://loebner.net/Prizef/TuringArticle.html

Turing A., ***Théorie des nombres calculables, suivie d'une application au problème de la décision*** dans ***La machine de Turing***, J.-Y. Girard, Seuil, pp. 49-104, 1995.

Turing A., ***Les ordinateurs et l'intelligence***, in ***La machine de Turing***, J.-Y. Girard, Seuil, pp. 135-175, 1995.

Auroux S., ***La révolution technologique de la grammatisation***, Mardaga, 1994.

Bailly F., G. Longo., ***Mathématiques et sciences de la nature. La singularité physique du vivant***, Hermann, 2006.

Bourgine P., A. Lesne (sous la direction de), ***Morphogenèse. L'origine des formes***, coll. « Échelles », Éditions Belin, 2006.

Gandy, R., ***The Confluence of Ideas in 1936***, in ***The Universal Turing Machine; a Half-Century Survey***, R. Herken. Oxford University Press : 55-111, 1988.

Gödel K., ***Some remarks on the undecidability results***, 1972, in Collected Works, Oxford University Press. II : pp. 305-306.

Goldstine H., ***The Computer from Pascal to von Neumann***, Princeton University Press, 1972.

Good I. J., ***A. M. Turing's Statistical Work in World War II***, in Biometrika 66 (2) : 393-396, 1979.

Good I. J., ***Introductory Remarks for the Article «A. M. Turing's Statistical Work in World War II»*** (1992), Collected Works of A. M. Turing : Pure Mathematics, J. L. Britton, North-Holland. 1 : 211-223.

Herken, R. (Ed.), ***The Universal Turing Machine; a Half-Century Survey***, Oxford University Press, 1988.

Herrenschmidt C. (1999), ***Écriture, monnaie, réseaux. invention des Anciens, inventions des Modernes***, Le Débat n°106, Gallimard.

Hilbert, D., ***Die logischen Grundlagen der Mathematik***, in Math. Annal 88 (1923) : pp. 151-165 et ***Über das Unendliche***, Math. Annal 95 (1926) : pp. 161-190 ; Trad. franç. dans ***Logique mathématique*** de J. Largeault, Armand Colin, 1972.

Hodges A., ***Alan Turing; The Enigma***, Allen & Unwin, 1983.

Hodges A., ***Alan Turing and the Turing Machine***, in ***The Universal Turing Machine; a Half-Century Survey***, R. Herken, Oxford University Press : pp. 3-15, 1988.

Hodges A., ***Alan Turing ; a Natural Philosopher***, Phœnix, 1997.

Lassègue J., ***Turing***, Paris, Belles Lettres, 2003.

Nagel E. & Newman J. R., Gödel K., Girard J.-Y., ***Le théorème de Gödel***, Seuil, 1989.

Rejewski, M, ***How Polish Mathematicians Deciphered the Enigma***, in ***Annals of the History of Computing***, 3 (3) : pp. 213-234, 1981.

Saunders, P. T. ***Introduction*** in ***Collected Works of A. M. Turing***, vol. 4, «Morphogenesis», North-Holland : XI-XXIV, 1992.

Thompson D'Arcy W., ***On Growth and Form***, Cambridge University Press , 1917.

Von Neumann, J., ***Preliminary Discussion of the Logical Design of an Electronic Computing Instrument***, in ***John von Neumann Collected Works*** volume V, Pergamon Press. V : pp. 34-79, 1946.

Von Neumann J., ***Theory of Self-Reproducing Automata***, University of Illinois Press, 1966.

http://www.turing.org.uk/
Site de référence maintenu par Andrew Hodges.

POUR LA SCIENCE

8, rue Férou 75278 PARIS CEDEX 06 • Tél. : 01-55-42-84-00 • www.pourlascience.com
Commande de numéros (Génies ou magazine) au 08-92-69-12-88

Directrice de la rédaction et rédactrice en chef : Françoise Pétry
Pour la Science :
Rédacteurs en chef adjoints : Maurice Mashaal, Loïc Mangin
Rédacteurs : François Savatier, Philippe Ribeau-Gesippe, Bénédicte Salthun-Lassalle
Dossiers Pour la Science :
Rédactrice en chef adjointe : Bénédicte Leclercq
Rédacteur : Daniel da Rocha
Génies de la Science :
Rédactrice : Marie-Neige Cordonnier
Cerveau & Psycho :
Rédacteur : Sébastien Bohler

Directrice artistique : Céline Lapert
Secrétariat de rédaction et maquette : Annie Tacquenet, Pauline Bilbault, Pascale Thollier-Dumartin, Sylvie Sobelman, Raphaël Queruel, Aurore Carlier.
Site internet : David Martin
Marketing et Publicité : Philippe Rolland, assisté de Marina Ballini
Direction financière : Anne Gusdorf
Direction du personnel : Jean-Benoît Boutry
Fabrication : Jérôme Jalabert, assisté de Christine de La Rochère et Marianne Sigogne
Presse et communication : Susan Mackie
Directeur de la publication et Gérant : Marie-Claude Brossollet
Conseillers scientifiques : Philippe Boulanger et Hervé This

Ont collaboré à ce numéro : Laurence André, Aurélie Dureuil

PUBLICITÉ France
Directeur de Publicité : Jean-François Guillotin
(jf.guillotin@pourlascience.fr)
Tél. : 01 55 42 84 28. Télécopieur : 01 43 25 18 29

SERVICE ABONNEMENTS
Ginette Grémillon : 01 55 42 84 04.

DIFFUSION DE POUR LA SCIENCE
Canada : Edipresse : 945, avenue Beaumont, Montréal, Québec, H3N 1W3 Canada. Suisse : Servidis : Chemin des chalets, 1979 Chavannes - 2 - Bogis. Belgique : La Caravelle : 303, rue du Pré-aux-oies - 1130 Bruxelles. Autres pays : Éditions Belin : 8, rue Férou - 75278 Paris CEDEX 06.

Bibliographie complémentaire sur le site Pour la Science, www.pourlascience.com

Imprimé en France par Groupe Opale-Istra - 67300 Schiltigheim – Dépôt légal : novembre 2006 – N° d'édition : 075078 ISBN 2-84245-078-7
Commission paritaire n° 0907K86975 du 21-09-05
Distribution : NMPP – ISSN 1298-6879 – Directeur de publication et Gérant : Marie-Claude Brossollet.