

Calculer avec l'ADN

LEONARD ADLEMAN

L'utilisation d'ADN pour résoudre des problèmes mathématiques conduit à repenser la notion de «calcul».



Quand on évoque les ordinateurs, on pense claviers, écrans, souris, mémoire vive, circuits micro-électroniques, gigaoctets et mégahertz. Les succès de l'informatique classique ont fait croire que seules des puces de silicium pouvaient recevoir des circuits capables d'effectuer des calculs, mais est-ce exact? L'ordinateur que vous utilisez inconsciemment pour lire ces mots, votre cerveau, n'a pas grand-chose à voir avec les ordinateurs classiques. Peut-être avons-nous une vision trop étroite de ce qu'est le calcul. Ne gagnerait-on pas à élargir la notion d'ordinateur? Certains peuvent-ils être des liquides, où les calculs résulteraient d'interactions de molécules? De telles machines existent : ce sont les ordinateurs à ADN.

Je m'intéresse à ces systèmes depuis un séjour dans un laboratoire de biologie moléculaire, en 1993. Mathématicien et informaticien, j'ai d'abord fait de la recherche médicale, contribuant notamment à des études du SIDA. Comment montrer aux biologistes l'intérêt des modèles mathématiques ou informatiques? Sous la houlette de Nicolas Chelyapov, j'ai commencé à apprendre les méthodes de la biologie moderne.

Ce travail est fascinant : on crée de l'ADN qui n'existe pas dans la nature et on l'introduit dans des bactéries où

il déclenche la production de protéines qui peuvent changer la nature même de l'organisme. Pendant cette période d'apprentissage, je lus la *Biologie moléculaire du gène*, un manuel classique de biologie dû à James Watson, l'un des deux découvreurs de la structure en double hélice de l'ADN, et ma conception de la biologie en fut bouleversée : la biologie moléculaire n'était plus la science des produits odorants conservés dans des réfrigérateurs ; elle gagnait en profondeur et en puissance explicative. La biologie est aujourd'hui l'étude de l'information stockée dans l'ADN par des enchaînements de quatre lettres, A, T, G et C, les quatre bases qui sont l'adénine, la thymine, la guanine et la cytosine ; la biologie étudie aussi les modifications de ces informations dans la cellule. Ces modifications se font selon des règles mathématiques.

Un soir, dans mon lit, alors que je lisais le livre de J. Watson, j'en arrivai à la description de l'ADN polymérase, la reine des enzymes, celle qui crée la vie : dans un milieu approprié, cette enzyme fabrique, à partir d'un brin d'ADN, un ADN complémentaire, où chaque C est remplacé par un G, chaque G par un C, chaque A par un T et chaque T par un A. Par exemple, en partant

d'une molécule dont la séquence est CATGTC, l'ADN polymérase engendre une nouvelle molécule de séquence GTACAG. L'ADN polymérase assure la reproduction de l'ADN, ce qui permet aux cellules de se reproduire et, indirectement, à l'espèce humaine de se développer : pour un biologiste réductionniste, la réplication de l'ADN par l'ADN polymérase est l'essence même de la vie.

L'ADN polymérase est une machine étonnante : à elle seule, cette molécule s'accroche à un brin d'ADN, se déplace dessus en «lisant» chaque base rencontrée et en «écrivant» son complément, fabriquant ainsi un nouveau brin d'ADN. Alors que je pensais à cette enzyme remarquable, je compris qu'elle fonctionnait comme une machine décrite en 1936 par le mathématicien anglais Alan Turing. Ce dernier, ainsi que Kurt Gödel, Alonzo Church et Stephen Kleene, avait analysé la notion de «calculabilité». Ces travaux théoriques, qui précédaient de plus d'une décennie les premiers ordinateurs, ont abouti à certains des résultats les plus importants des mathématiques du XX^e siècle (voir, par exemple, *Le hasard en théo-*



1. LES MOLÉCULES D'ADN stockent de l'information. Ce sont des enchaînements d'adénine, de thymine, de guanine et de cytosine, respectivement représentées par les lettres A, T, G et C. La molécule dessinée ici en couleurs, de séquence GCAGTCGGACTGGGC-TATGTCGA, code la solution du problème du chemin hamiltonien représenté page suivante.

rie des nombres, par Gregory Chaitin, *Pour la Science*, septembre 1988).

Turing avait inventé un ordinateur abstrait que l'on nomme aujourd'hui «machine de Turing». Cette machine est composée de deux bandes, ainsi que d'un système nommé contrôleur à états finis, qui se déplace le long d'une bande pour y lire des données et, simultanément, se déplace le long de la deuxième bande pour y écrire d'autres données. Le contrôleur se programme à l'aide d'instructions simples, de sorte qu'il est aisé d'écrire un programme qui lit sur la bande d'entrée une chaîne composée de A, de T, de C et de G, et qui écrit la chaîne complémentaire sur la bande de sortie. La similitude avec l'ADN polymérase est patente.

La similarité est intéressante, parce que la machine de Turing est universelle, c'est-à-dire qu'elle peut être programmée pour calculer tout ce qui est calculable. Cette idée est la «thèse de Church». Elle me fit penser que l'on pourrait programmer une machine de Turing pour produire une chaîne complémentaire, pour factoriser des nombres ou pour jouer aux échecs. Je me dressai soudain dans mon

lit et fis remarquer à mon épouse : «Bon sang! Ces choses peuvent calculer.» Je ne dormis pas de la nuit, cherchant un moyen d'utiliser l'ADN pour résoudre des problèmes.

Je voulus d'abord concevoir un ordinateur à ADN analogue à une machine de Turing, où le contrôleur à états finis serait remplacé par une enzyme. Une idée identique avait déjà été proposée dix ans plus tôt par des chercheurs de la Société IBM (voir *Les limites physiques du calcul*, par Charles Bennett et Rolf Landauer, *Pour la Science*, septembre 1985). Malheureusement, si une enzyme (l'ADN polymérase) permet de créer des ADN complémentaires, où trouverait-on les molécules qui assureraient d'autres calculs importants, telle la factorisation des nombres?

Les biologistes moléculaires ont la chance inouïe de pouvoir utiliser le contenu des cellules. Nous sommes très loin de savoir créer *ex nihilo* des machines moléculaires aussi perfectionnées que l'ADN polymérase, mais trois ou quatre milliards d'années d'évolution ont engendré des cellules pleines de petites merveilles. La biotechnologie moderne emploie certaines de ces

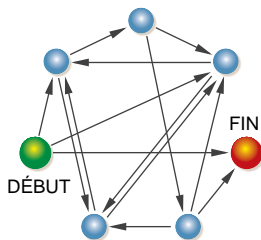
molécules, mais l'évolution n'a jamais produit de cellules qui jouent aux échecs. Aussi, pour construire des machines cellulaires aux capacités de calcul intéressantes, je devais utiliser les outils disponibles. Ces outils sont essentiellement les suivants :

1. L'appariement des bases. À chaque brin d'ADN correspond une molécule complémentaire. Dans une solution, lorsqu'un brin d'ADN rencontre l'ADN complémentaire, les deux brins s'apparient, c'est-à-dire qu'ils s'enroulent l'un autour de l'autre pour former une double hélice. Les brins ne réagissent pas chimiquement, mais se lient parce que des forces faibles, des «liaisons hydrogène», s'établissent entre les bases complémentaires. Quand un brin d'ADN rencontre un autre brin d'ADN qui ne lui est pas complémentaire ou qui ne possède pas de longues parties complémentaires, alors les deux brins ne se lient pas.

2. Des polymérases dupliquent l'information d'une molécule en créant une molécule complémentaire. Par exemple, l'ADN polymérase fabrique un brin d'ADN complémentaire à partir de n'importe quelle molé-

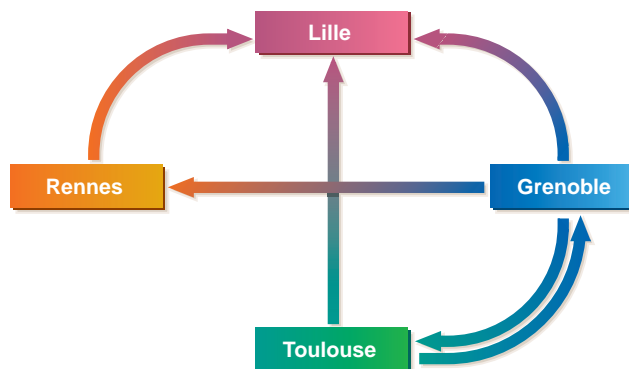
Le chemin hamiltonien codifié par des bases

Considérons des villes reliées par un certain nombre de vols sans escale (*en bas*). Dans l'exemple représenté, on peut aller de Grenoble à Lille, mais pas l'inverse. On cherche s'il existe un chemin qui part de Toulouse et se termine à Lille, en passant une fois seulement dans les autres villes. On attribue à chaque ville une séquence d'ADN (ACTTGCAG pour Toulouse) qui peut être considérée comme un prénom (ACTT) et d'un nom propre (GCAG). Les codes des vols peuvent alors être définis par la mise bout à bout du nom propre de la ville d'origine et du prénom de la ville de destination (*en bas, à droite*).



Les complémentaires des noms des villes s'obtiennent par le remplacement de C par G, de G par C, de A par T et de T par A. Pour ce problème particulier, un seul chemin hamiltonien relie Toulouse à Lille : il passe successivement par Grenoble et Rennes, dans cet ordre. Dans le calcul, ce chemin est représenté par GCAGTCGGACTGGGC-TATGTCCGA, une séquence d'ADN de 24 bases de longueur. Le problème représenté ci-contre, à

gauche, a été résolu par un ordinateur à ADN : il fait intervenir 7 villes et 14 vols.



VILLE	ADN	COMPLÉMENTAIRE
TOULOUSE	ACTTGCAG	TGAACGTC
GRENOBLE	TCGGACTG	AGCCTGAC
RENNES	GGCTATGT	CCGATACA
LILLE	CCGAGCAA	GGCTCGTT
VOL		CODE ADN DU VOL
TOULOUSE - GRENOBLE		GCAGTCGG
TOULOUSE - LILLE		GCAGCCGA
GRENOBLE - RENNES		ACTGGGCT
GRENOBLE - LILLE		ACTGCCGA
GRENOBLE - TOULOUSE		ACTGACTT
RENNES - LILLE		ATGTCCGA

Slim Films

cule d'ADN. L'ADN polymérase commence cette copie quand elle a rencontré un signal de départ qui lui indique le début de la séquence à copier. Ce signal est fourni par une amorce, une séquence d'ADN particulière et souvent courte. Dès que l'ADN polymérase trouve une amorce appariée au modèle, elle commence à ajouter des bases à l'amorce pour créer une copie complémentaire du modèle.

3. Des enzymes nommées ligases raboutent les molécules d'ADN. Lorsque deux molécules d'ADN sont proches, la ligase fait réagir une extrémité d'une molécule avec une extrémité de l'autre molécule : grâce à la liaison covalente ainsi établie, la ligase lie les deux molécules. Les cellules utilisent les ligases pour réparer les molécules d'ADN coupées par les rayonnements ultraviolets ou par d'autres rayonnements ionisants.

4. Des enzymes nommées nucléases coupent les molécules d'ADN. Par exemple, les enzymes de restriction cherchent une succession spécifique de bases dans un brin d'ADN et, quand elles l'ont trouvée, elles coupent l'ADN en deux parties. L'enzyme EcoRI, par exemple, est une enzyme de restriction qui coupe l'ADN de la bactérie *Escherichia coli* après le G de la séquence GAATTC ; elle ne coupe jamais

à un autre endroit. On a supposé que ces enzymes de restriction défendaient la bactérie contre les virus bactériophages, qui injectent leur ADN parasite dans la cellule bactérienne. La bactérie *Escherichia coli* protège son propre ADN de l'action de l'enzyme EcoRI par des méthylations (l'addition de groupes méthyle), lors de la répllication, mais tout virus qui posséderait la séquence GAATTC est inactivé, parce que son ADN est découpé en morceaux. Mon ordinateur à ADN n'utilise pas d'enzymes de restriction, mais celles-ci ont été utilisées, dans de nombreuses expériences, par d'autres équipes de recherche.

5. Des gels pour électrophorèse. Ceux-ci ne sont pas des constituants des cellules. On sépare des molécules d'ADN placées dans une solution en déposant une petite quantité de la solution sur un gel et en appliquant une différence de potentiel entre les deux extrémités du gel. Les molécules d'ADN négativement chargées migrent alors vers l'anode en se frayant un chemin à travers la structure réticulée du gel, à une vitesse qui dépend de leur longueur, les molécules les plus courtes se déplaçant plus vite que les molécules plus longues. Ainsi, les biologistes séparent les brins d'ADN selon leur longueur : l'ajout d'une substance fluorescente qui se

lie à l'ADN révèle la répartition finale des molécules d'ADN en fonction de leur longueur.

6. La synthèse de l'ADN. Aujourd'hui, on peut écrire une séquence d'ADN sur un bout de papier, l'envoyer à un laboratoire de synthèse, et recevoir quelques jours plus tard, un tube à essai qui contient approximativement 10^{18} molécules d'ADN, dont la plupart possèdent exactement la séquence demandée. On atteint facilement des longueurs de 100 bases et, pour des molécules dont la longueur est de 20 bases, le coût n'est que de 150 francs. Les molécules sont fournies déshydratées dans un tube à essai ; elles forment un petit grain, blanc et amorphe.

Comment ces divers ingrédients aident-ils à factoriser les nombres ou à jouer aux échecs ? Dans les années 1930, les mathématiciens ont démontré que deux ingrédients seulement sont indispensables pour construire un ordinateur : un moyen de stocker de l'information et un système pour effectuer quelques opérations simples. La machine de Turing stocke l'information au moyen de lettres écrites sur une bande et la manipule grâce aux instructions simples que possède le contrôleur à états finis. Un ordinateur électronique stocke l'information dans sa mémoire sous la forme de sé-

quences de 0 et de 1, et il la manipule grâce à l'unité de calcul. N'importe quelle façon de stocker l'information et n'importe quel jeu d'instruction suffisent.

Suffisent à quoi? À effectuer n'importe quel calcul. Pour synthétiser un ADN complémentaire ou pour déterminer des coups aux échecs, il suffit de stocker la bonne information et d'appliquer la bonne séquence d'opérations, c'est-à-dire d'exécuter un programme. Or, l'ADN est un excellent système de stockage de l'information : les cellules l'utilisent depuis des milliards d'années pour stocker le modèle de la vie. Les enzymes telles que les polymérases et les ligases agissent sur cette information. Était-ce suffisant pour construire une machine à calculer universelle? Grâce aux mathématiciens des années 1930, je savais que oui.

Le problème du chemin hamiltonien

Quel problème résoudre? Il ne devait pas être inventé spécialement pour la machine, et il devait démontrer les possibilités de cette nouvelle manière de calculer. J'ai choisi de résoudre le problème du chemin hamiltonien, une variante du problème du voyageur de commerce.

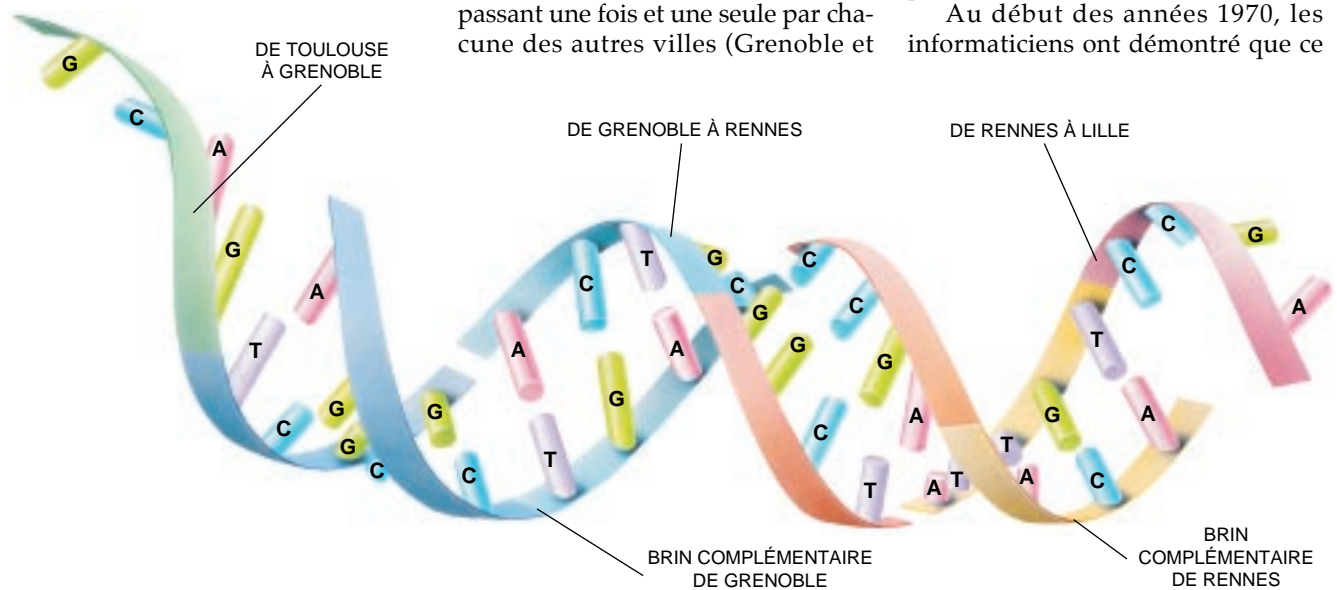
Les chemins hamiltoniens sont dus à William Rowan Hamilton, qui était astronome royal en Irlande, au milieu du XIX^e siècle. Le problème qui porte son nom est décrit dans l'encadré de la page 58 : soit une carte (graphe) où figurent des villes reliées par des vols sans escale (chemins orientés). Par exemple, on peut aller de Grenoble à Rennes, mais pas de Rennes à Grenoble. On cherche le chemin qui part de Toulouse (le sommet de départ) et se termine à Lille (le sommet d'arrivée) en passant une fois et une seule par chacune des autres villes (Grenoble et

Rennes). Un tel trajet est un chemin hamiltonien. Dans cet exemple, on voit rapidement qu'un seul chemin hamiltonien relie Toulouse à Lille ; c'est : Toulouse, Grenoble, Rennes, puis Lille. Entre Lille et Toulouse, il n'existe pas de chemin hamiltonien.

Plus généralement, étant donné un graphe orienté, un sommet de départ et un sommet d'arrivée, alors il existe un chemin hamiltonien si et seulement si il existe un chemin qui passe par tous les sommets exactement une fois. Le problème du chemin hamiltonien est la recherche de ce chemin.

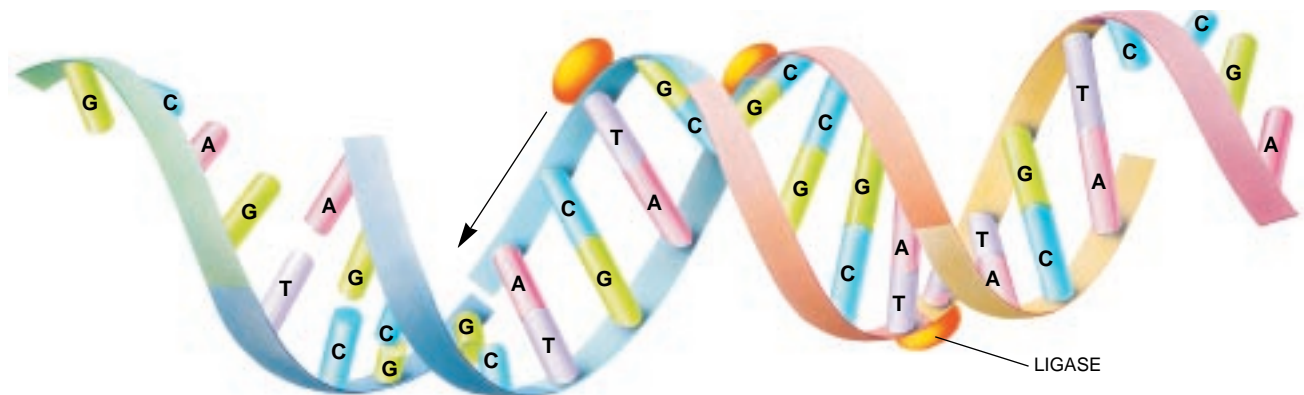
Ce problème a été très étudié par les informaticiens, mais aucun algorithme efficace (c'est-à-dire rapide) n'a été trouvé. On connaît des graphes de moins de 100 sommets pour lesquels, même en utilisant le meilleur algorithme et le meilleur ordinateur, la résolution du problème du chemin hamiltonien prendrait des centaines d'années.

Au début des années 1970, les informaticiens ont démontré que ce



2. L'ASSOCIATION DE MOLÉCULES D'ADN par appariement des C et des G, des A et des T conduit à un brin d'ADN représentant les vols (ici de Toulouse à Grenoble, de Grenoble à Rennes, et de Rennes à

Lille). Ces brins sont appariés à d'autres brins qui codent le complémentaire du nom des villes (ici Grenoble et Rennes). Des centaines de milliards d'appariements se produisent chaque seconde.



3. LES LIGASES RELIENT des molécules. Chaque fois que ces protéines trouvent deux molécules d'ADN à proximité, elles déclen-

chent une réaction des deux extrémités et relient les deux molécules en une seule par une liaison covalente.

Tomo Narashima

problème était «NP-complet» : la plupart pensent qu'on ne trouvera jamais d'algorithme rapide pour résoudre ce problème, bien que la preuve de cette supposition reste à fournir (voir *Les lois du tout ou rien*, par Jean-Paul Delahaye, *Pour la Science*, juillet 1995). Le problème du chemin hamiltonien n'est pas insoluble, mais le temps de calcul risque d'être excessivement long.

Considérons par exemple l'algorithme suivant :

Soit un graphe possédant n sommets.

1. Engendrer un ensemble de chemins aléatoires.
2. Pour chaque chemin de l'ensemble :
 - a. Vérifier si le chemin part du sommet de départ et se termine au sommet d'arrivée. Sinon, le retirer de l'ensemble.
 - b. Vérifier si ce chemin passe exactement par n sommets. Sinon, le retirer de l'ensemble.
 - c. Pour chaque sommet, vérifier que ce chemin y passe. Sinon, retirer ce chemin de l'ensemble.
3. Si l'ensemble n'est pas vide, alors il existe un chemin hamiltonien. Si l'ensemble est vide, alors il n'existe pas de chemin hamiltonien.

Cet algorithme est imparfait, mais si l'ensemble des chemins est engendré de manière suffisamment aléatoire et si l'ensemble résultant est suffisamment grand, alors cet algorithme donnera probablement la solution. C'est cet algorithme que

j'ai mis en œuvre dans la première expérience de calcul avec l'ADN.

Sept jours dans un laboratoire

Pour cette première expérience, j'ai cherché un problème suffisamment simple pour qu'un prototype d'ordinateur à ADN le résolve, mais suffisamment important pour constituer une preuve indubitable de l'intérêt de ces nouvelles machines. J'ai choisi un graphe comportant sept villes et 14 vols, représenté page 58. À la main, on met en moyenne une minute pour trouver l'unique chemin hamiltonien de ce graphe (vous pouvez commencer maintenant...).

Pour examiner la technique utilisée, considérons le graphe qui contient quatre villes seulement : Toulouse, Grenoble, Rennes et Lille, reliées par six vols. On cherche un chemin hamiltonien qui part de Toulouse et qui finit à Lille.

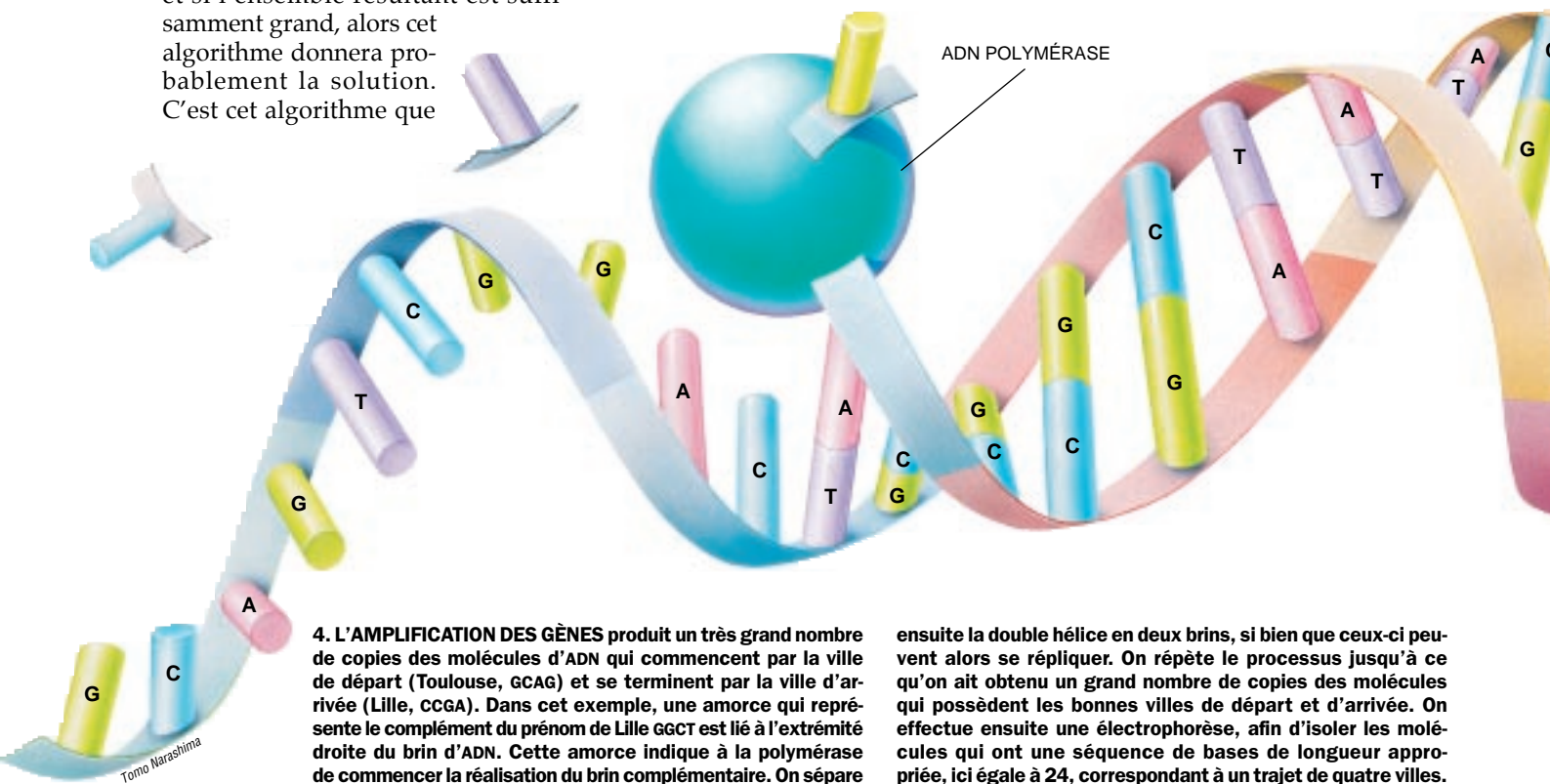
J'ai commencé par attribuer une séquence aléatoire d'ADN à chacune des villes : à Toulouse, j'ai associé ACTTGCAG, et à Grenoble TCGGACTG. J'ai alors imaginé que la première moitié des séquences était le prénom des villes et la seconde moitié leur nom propre. Ainsi le nom propre de Toulouse est GCAG, et le prénom de Grenoble TCGG.

J'ai ensuite codé chaque vol en mettant bout à bout le nom propre de la ville d'origine avec le prénom de la ville d'arrivée. Dans notre exemple, le code du vol reliant Toulouse à Grenoble est GCAGTCGG.

Comme on peut former un ADN complémentaire de tout brin d'ADN, chaque ville possède un nom complémentaire. Le nom complémentaire de Toulouse est TGAACGTC.

Après avoir effectué ces codages, j'ai synthétisé les séquences d'ADN complémentaires des noms de ville et les séquences des codes des vols. J'ai ensuite pris une pincée de chaque séquence, environ 10^{14} molécules que j'ai placées dans un tube à essai. Pour que le calcul commence, j'ai simplement ajouté de l'eau, de la ligase, du sel et quelques autres composants qui reproduisent les conditions qui règnent à l'intérieur d'une cellule. En tout, j'ai utilisé un cinquième de cuillère à café de solution et, une seconde plus tard, j'avais la solution du problème du chemin hamiltonien.

Afin de comprendre le déroulement du calcul, examinons les réactions dans le tube. Le numéro du vol reliant Toulouse à Grenoble (GCAGTCGG), par exemple, et le nom complémentaire de Grenoble (AGCCTGAC) peuvent se rencontrer par hasard. Or, par construction, la première séquence se termine



4. L'AMPLIFICATION DES GÈNES produit un très grand nombre de copies des molécules d'ADN qui commencent par la ville de départ (Toulouse, GCAG) et se terminent par la ville d'arrivée (Lille, CCGA). Dans cet exemple, une amorce qui représente le complément du prénom de Lille GGCT est lié à l'extrémité droite du brin d'ADN. Cette amorce indique à la polymérase de commencer la réalisation du brin complémentaire. On sépare

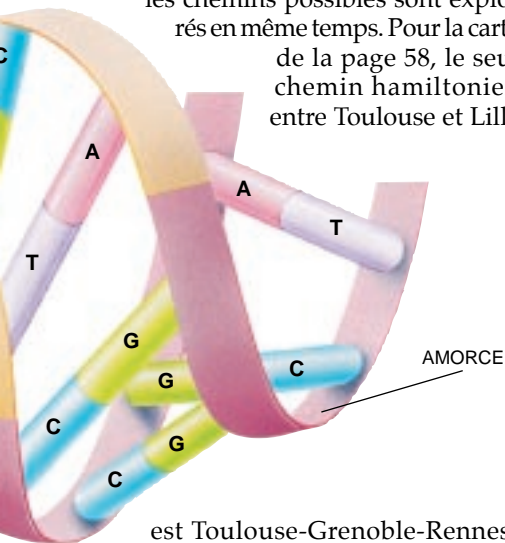
ensuite la double hélice en deux brins, si bien que ceux-ci peuvent alors se répliquer. On répète le processus jusqu'à ce qu'on ait obtenu un grand nombre de copies des molécules qui possèdent les bonnes villes de départ et d'arrivée. On effectue ensuite une électrophorèse, afin d'isoler les molécules qui ont une séquence de bases de longueur appropriée, ici égale à 24, correspondant à un trajet de quatre villes.

par TCGG, et la deuxième commence par AGCC. Ces séquences étant complémentaires, elles s'associent. La molécule complexe ainsi formée peut ensuite rencontrer la molécule associée au vol Grenoble-Rennes (ACTGGGCT), qui se liera à son tour au complexe, parce que la fin de celui-ci (TGAC) est complémentaire du début du code du vol (ACTG). Les molécules s'allongent ainsi à mesure que les codes des vols s'associent aux noms complémentaires des villes. Les ligases du mélange relient en permanence les chaînes d'ADN codant les vols. Dans le tube, les molécules restantes représentent des chemins aléatoires passant par les différentes villes. La première étape de notre algorithme est terminée.

Comme on introduit initialement un très grand nombre de molécules et que le problème ne fait intervenir qu'un petit nombre de villes, on obtient très vraisemblablement au moins une molécule qui code le chemin hamiltonien cherché. Ainsi la solution d'un problème mathématique est stockée dans une simple molécule!

Remarquons aussi que tous les chemins sont créés en même temps grâce aux interactions simultanées de centaines de milliers de milliards de molécules. Cette réaction biochimique est un gigantesque calcul parallèle : tous les chemins possibles sont explorés en même temps. Pour la carte

de la page 58, le seul chemin hamiltonien entre Toulouse et Lille



est Toulouse-Grenoble-Rennes-Lille. La molécule qui code cette solution possède donc la séquence GCAGTCGGACTGGGCTATGTCCGA. Cette solution est présente dans le tube sous la forme double brin, c'est-à-dire avec la séquence complémentaire appariée.

Toutefois, cette molécule se trouvait parmi les centaines de milliards

Cristal d'ADN

La science est très loin de pouvoir créer des ordinateurs moléculaires et, sans l'exemple donné par la vie elle-même, il ne serait pas évident qu'une telle organisation est possible. Toutefois, l'expérience décrite dans cet article a montré comment les bases complémentaires assurent l'auto-assemblage des molécules d'ADN en une structure linéaire. Des structures autres que linéaires peuvent-elles être construites de la sorte?

Les travaux de Nadrian Seeman, de l'Université de New York, sur les structures d'ADN branchées apportent un élément de réponse. Les structures d'ADN branchées, des molécules d'ADN avec plusieurs extrémités, sont présentes dans la cellule lors de la réplication de l'ADN et pendant la recombinaison. Avec ses collègues, N. Seeman a appris à façonner des molécules d'ADN en des structures branchées. Sa construction d'un cube dont les

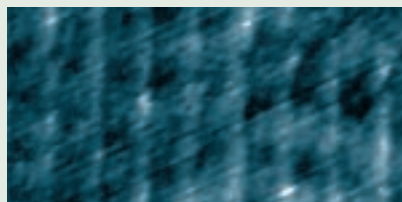
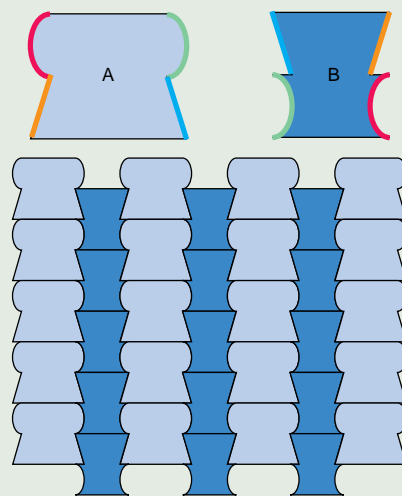
arêtes sont des molécules d'ADN marque le début de ce que l'on nomme «les nanotechnologies de l'ADN».

En 1997, je fis le lien entre les travaux de N. Seeman et ceux de L. Adleman grâce à la théorie des pavages (voir les pavages de Penrose dans l'article de Martin Gardner, page 86). La théorie des pavages est intéressante non seulement par sa nature géométrique, mais aussi parce qu'elle est liée à la logique et au calcul. Dans les pavages, la forme des bords des pavés détermine quels pavés peuvent être assemblés. Dans un brin d'ADN, la séquence des bases détermine quels brins s'associent. Le passage aux «pavés moléculaires» constitués de brins d'ADN était alors naturel.

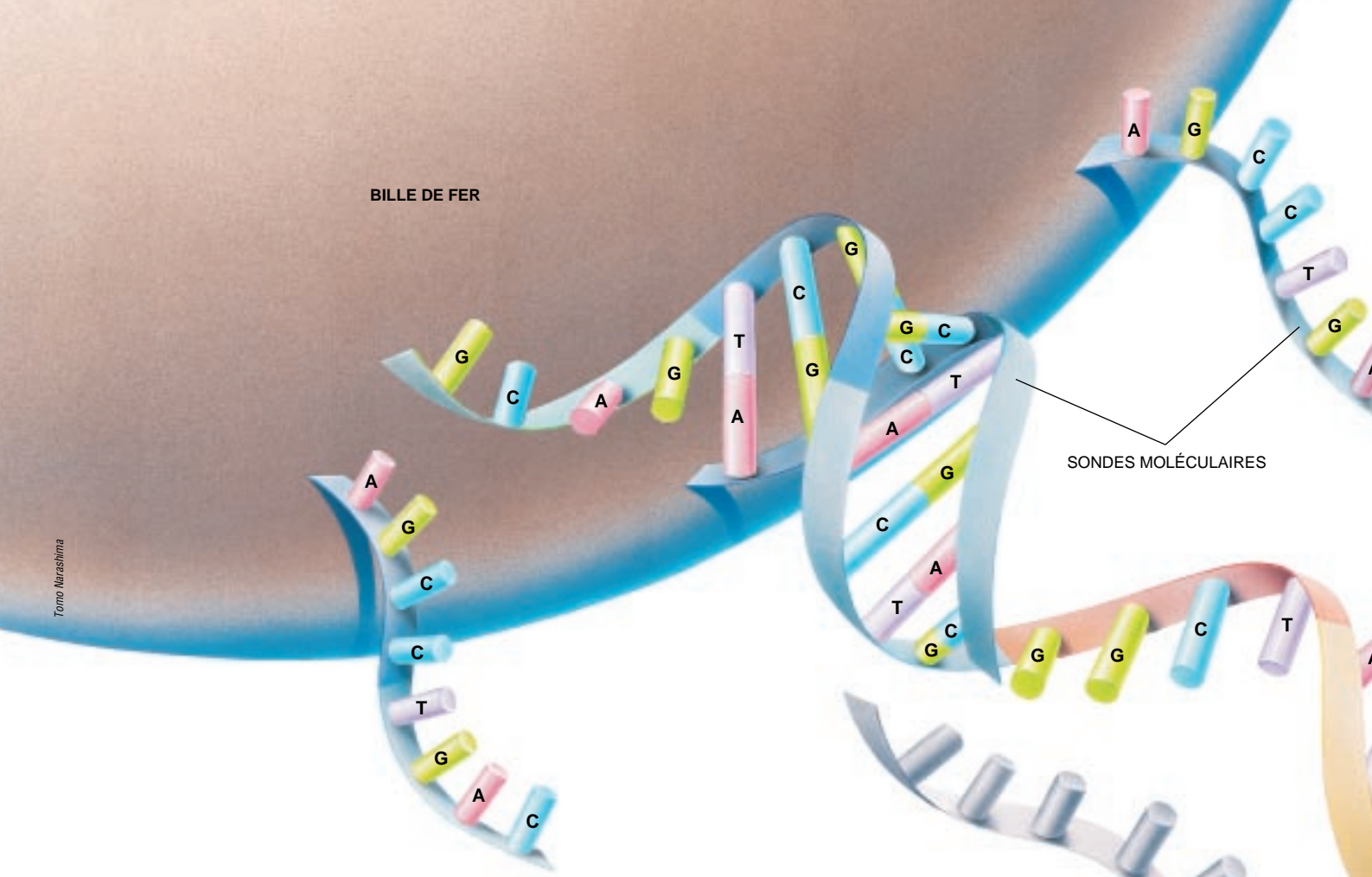
Avec le groupe de N. Seeman, nous avons construit le premier pavage moléculaire en laboratoire. Deux pavés moléculaires A et B (deux structures branchées), s'assemblent d'une façon unique pour produire un pavage rayé. Chaque pavé moléculaire comporte quatre extrémités avec des séquences uniques ; chaque extrémité du pavé A est complémentaire d'une extrémité du pavé B, de sorte que, lorsque l'on mélange de nombreux pavés dans une solution, ils s'assemblent spontanément en un cristal bidimensionnel d'ADN (voir la figure).

À première vue, ce simple motif rayé ne ressemble en rien à du calcul. Toutefois, les pavés moléculaires sont «programmables» : on peut modifier facilement les extrémités d'assemblages de chaque pavé pour déterminer leurs interactions. En fait, on peut construire ainsi une infinité de pavés, dont des pavés qui, comme dans l'expérience d'Adleman, simuleraient une machine de Turing par la «logique» de leur assemblage. L'auto-assemblage des pavés d'ADN peut fonctionner comme un ordinateur universel.

Erik WINFREE,
Institut de technologie de Californie



Le premier pavage bidimensionnel d'ADN est obtenu en mélangeant deux sortes de pavés d'ADN, des structures d'ADN qui comportent des extrémités complémentaires (segment colorés, en haut). Ces pavés s'assemblent selon une structure rayée unique (au centre), révélée par microscopie à force atomique (en bas).



Tomo Narashima

5. À L'AIDE DE SONDÉS MOLÉCULAIRES, on identifie les brins d'ADN qui codent des chemins passant par les villes intermédiaires (Grenoble et Rennes). Les molécules sondes qui comportent le complémentaire du nom de Grenoble (AGCCTGAC) sont attachées à une bille de fer que l'on met en suspension dans le liquide. En raison des forces entre bases complémentaires, ces molécules captent les brins

d'ADN qui contiennent le nom de Grenoble (TCGGACTG). Les brins qui ne contiennent pas le nom de Grenoble sont alors enlevés. On répète ce processus avec des molécules sondes qui codent le complémentaire du nom de Rennes. Lorsque toutes les étapes sont terminées, les brins qui restent sont ceux qui codent la solution du problème du chemin hamiltonien : GCAGTCGGACTGGGCTATGTCGGA.

de molécules qui codent des chemins qui ne sont pas hamiltoniens. Comment les éliminer? Pour enlever les molécules qui ne commencent pas par la ville de départ et qui ne se terminent pas par la ville d'arrivée, je me suis servi de la technique d'amplification des gènes (PCR). Cette technique utilise de nombreux exemplaires de deux séquences «amorces» complémentaires des deux extrémités de la séquence d'ADN que l'on désire amplifier. Ainsi, l'ADN polymérase effectue plusieurs cycles de réplication, alternativement sur chacun des brins. Les amorces utilisées sont, d'une part, le nom propre de la ville de départ (GCAG pour Toulouse) et, d'autre part, le complémentaire du prénom de la ville d'arrivée (GGCT pour Lille). Ces deux amorces agissent simultanément : la première prévient l'ADN polymérase de copier les séquences qui possèdent la bonne ville de départ,

la seconde déclenche la duplication de molécules qui codent la bonne ville d'arrivée.

Pour amplifier l'ADN, on effectue des cycles à haute et à basse température. À haute température, on dissocie les séquences double brin pour libérer les séquences complémentaires des amorces. À basse température, on apparie les amorces, et l'ADN polymérase effectue la réplication, recréant une molécule double brin.

Ainsi, les molécules qui possèdent les bonnes villes de départ et d'arrivée se multiplient à une vitesse très élevée (exponentielle). En revanche, les molécules qui possèdent la bonne ville de départ, mais pas la bonne ville d'arrivée, ou réciproquement, se multiplient à une vitesse inférieure (linéaire). Les séquences d'ADN qui ne possèdent ni la bonne ville de départ, ni la bonne ville d'arrivée ne se reproduisent pas du tout. Après la réaction,

le tube à essai contient beaucoup de molécules dont les villes de départ et d'arrivée sont les bonnes, et peu de molécules ne satisfaisant pas ces deux critères. L'étape 2a de l'algorithme est alors terminée.

J'utilise ensuite un gel d'électrophorèse pour identifier les molécules qui possèdent la bonne longueur, 24 dans notre exemple ; toutes les autres molécules sont éliminées. C'est l'étape 2b de notre algorithme.

Afin de vérifier que les séquences qui restent codent bien des chemins qui passent par les villes intermédiaires, j'utilise la complémentarité des séquences d'ADN dans une procédure nommée séparation par affinité. On utilise des sondes d'ADN qui codent le complémentaire du nom d'une ville particulière, par exemple Grenoble. Ces sondes sont fixées à de microscopiques sphères de fer d'un micromètre de diamètre environ.

Quand on met ces billes en suspension dans le tube contenant les molécules restantes, seules les molécules contenant le nom de la ville désirée (Grenoble) se fixent sur les sondes. On place ensuite un aimant contre la paroi du tube à essai pour attirer et maintenir les boules métalliques. Puis on jette le liquide qui ne contient plus que des molécules ne possédant pas le nom désiré.

On ajoute alors un autre solvant et l'on retire l'aimant, afin de remettre les billes en suspension. Une augmentation de la température du mélange permet aux molécules de se séparer des sondes et de se dissoudre à nouveau dans le liquide. À l'aide de l'aimant, on attire les billes contre la paroi du tube à essai, mais, cette fois, aucune molécule n'y est fixée. Le liquide contient alors les brins d'ADN désirés : ceux qui codent des chemins passant par Grenoble. Il peut être versé dans un autre tube pour un autre examen. On répète la procédure pour

velle amplification des gènes, puis une autre électrophorèse. Cette analyse finale montra que les molécules restantes codaient effectivement le chemin désiré. Après sept jours de laboratoire, le premier calcul utilisant de l'ADN était terminé.

L'émergence d'un nouveau domaine

Comment perfectionner ce système? Les ordinateurs moléculaires ont de nombreuses propriétés intéressantes. Leurs capacités de stockage sont considérables : un gramme d'ADN sec occupe un volume de un centimètre cube environ et stocke autant d'informations qu'un millier de milliards de disques compacts. En outre, les ordinateurs moléculaires fonctionnent de manière très parallèle. Même dans notre petite expérience conduite avec un cinquième de cuillerée à café de solution, 10^{14} codes de vols étaient mis bout à bout chaque seconde. L'ordinateur micro-électronique le plus puissant du monde n'accomplira vraisemblablement jamais une telle tâche aussi rapidement.

Les ordinateurs moléculaires consomment extrêmement peu d'énergie : un joule suffit, en principe, pour effectuer 2×10^{19} opérations de liaison. C'est remarquable, car le deuxième principe de la thermodynamique fixe la limite à 34×10^{19} opérations irréversibles par joule à la température ambiante.

Les supercalculateurs existants en sont très loin : ils ne dépassent pas 10^9 opérations par joule.

De nombreux théoriciens et expérimentateurs étudient aujourd'hui les ordinateurs moléculaires. Réussiront-ils à construire des ordinateurs moléculaires qui rivaliseront avec les ordinateurs électroniques? De gigantesques investissements en hommes et en argent ont fait de l'ordinateur électronique une merveille de notre époque ; il sera difficile à battre.

Cependant, ce serait une erreur de considérer seulement l'intérêt pratique des ordinateurs moléculaires. Mon expérience est la première d'un nouveau domaine des sciences. Partout, cette nouvelle «science moléculaire» fait des émules. Par exemple, Gerald Joyce, qui travaille à l'Institut Scripps, en Californie, et Laura Landweber, de l'Université Princeton, élèvent des milliards de molécules d'ARN, génération après génération, jusqu'à ce que les «meilleures» aient les propriétés de catalyse désirées (voir *L'évolution moléculaire dirigée*, par Gerald Joyce, *Pour la Science*, février 1993). À l'Institut de technologie du Massachusetts, Julius Rebek crée des molécules qui peuvent se reproduire, ce qui nous donne des indications sur la façon dont la vie est apparue sur Terre (voir *Qu'est-ce que la vie?*, par Julius Rebek, *Pour la Science*, septembre 1994). À l'Institut de technologie de Californie, Erik Winfree synthétise des complexes moléculaires «intelligents», qui peuvent être «programmés» pour s'auto-assembler en des structures prédéterminées de complexité arbitraire (voir l'encadré de la page 61). Le potentiel de la biochimie est considérable.

La biologie et l'informatique, qui se sont épanouies durant le dernier demi-siècle, progresseront encore beaucoup et seront à l'origine de développements industriels importants au siècle prochain. Entre la biologie et l'informatique, entre la vie et le calcul, de grandes découvertes restent à faire.

les autres villes intermédiaires, Rennes dans notre exemple. Après une journée complète de travail au laboratoire, je terminai cette procédure itérative ; c'était la partie la plus fastidieuse de l'expérience.

L'étape 2c de l'algorithme se termine avec la fin de la séparation par affinité : dans le tube se trouvent alors les molécules d'ADN qui codent le chemin hamiltonien. Ainsi, si le tube contenait un seul type de molécule, je pouvais en conclure qu'un chemin hamiltonien existait dans le graphe sinon, un tel chemin n'existait pas. Comment le savoir? En utilisant une nou-

Leonard ADLEMAN travaille au département d'informatique de l'Université de Californie du Sud.

James D. WATSON, *Biologie moléculaire du gène*, InterÉditions, 1978.

Leonard M. ADLEMAN, *Molecular Computation of Solutions to Combinatorial Problems*, in *Science*, vol. 266, pp. 1021-1024, 11 novembre 1994.

David K. GIFFORD, *On the Path to Com-*

putation with DNA, in *Science*, vol. 266, pp. 993-994, 11 novembre 1994.

Richard J. LIPTON, *DNA Solution of Hard Computational Problems*, in *Science*, vol. 268, pp. 542-545, 28 avril 1995.

Jean-Paul DELAHAYE, *Les lois du tout ou rien*, in *Pour la Science*, juillet 1995.

D'autres informations sur les ordinateurs à ADN peuvent être trouvées sur notre site : <http://www.pourlascience.com/>