DNA computing on a chip

Mitsunori Oghara and Animesh Ray

In a DNA computer, the input and output are both strands of DNA. A computer in which the strands are attached to the surface of a chip can now solve difficult problems quite quickly.

“...how often have I said that when you eliminate the impossible, whatever remains, however improbable, must be the truth?” exalted the great sleuth. The principle of arriving at the truth by elimination is ancient; but on page 175 of this issue, Liu et al. report a new technique for massively parallel elimination, which harnesses the power of DNA chemistry and biotechnology to solve a particularly difficult problem in mathematical logic.

The difficulty of finding solutions to mathematical problems is classified by the speed at which the best algorithm can compute their solutions. ‘Easy’ problems have algorithms with ‘running times’ that scale as a polynomial function of the number of variables (polynomial time or P problems). There is also a class of problems characterized by proofs that are easy to verify (non-deterministic polynomial time or NP problems), such as the famous travelling salesman problem. In the worst case, ‘hard’ NP problems have running times that grow exponentially with the number of the variables. For example, finding a factor of a given natural number N cannot be done in polynomial time, but verifying that another number d is a factor of N is easy. Computer scientists have been intensively studying whether sequential algorithms can solve all NP problems in polynomial time, but the answer is still unknown.

In 1994, Leonard Adleman shocked the computing world by presenting a DNA-based polynomial-time method for the Hamilton path problem (Fig. 1a), the problem of finding an airline flight path between several cities on a map such that each city is visited only once. This NP problem is known to be one of the hardest. In order to achieve the small computation time, Adleman traded space (the amount of DNA needed) for time (the number of biochemical steps to be used). His key insight was that cities on a map, and paths between pairs of cities, may be encoded in strands of DNA. Millions of DNA strands, diffusing in a liquid, can self-assemble into all possible flight-path configurations, from which a judicious series of molecular manoeuvres can fish out the correct solution. Adleman, combining elegance with brute force, could isolate the one true solution out of many possibilities.

Every NP problem can be seen as the search for a solution that simultaneously satisfies a number of logical clauses, each composed of three variables (which can be true or false), connected by ‘or’ statements: for example (x1 OR x2 OR x3) AND (x1 OR x4 OR x6). This particular problem, known as 3-SAT, is the hardest of all NP problems. Liu et al. show how to solve a simple case of 3-SAT in a reasonable amount of time by using a brute-force search made possible by the parallel nature of their DNA computing techniques.

They begin with a string of binary numbers representing the variables in a given 3-SAT formula. Such a binary string can be represented by a unique sequence of nucleotides in single-stranded DNA; for example, TGGCGG might stand for 001. For n variables, there are 2ⁿ unique answers (or Watson strands), so for three variables you need eight Watson strands. For each Watson strand, there is also a complementary Crick strand created by the base-pairing rule — A bonds to T, and C bonds to G. The goal is to identify those strings out of a library of eight that satisfy all the clauses of a particular 3-SAT formula (Fig. 1b).

Liu et al. first immobilized the Watson DNA strands corresponding to all candidate solutions on a specially treated gold surface. Next they added all possible Crick strands that will stick to a Watson string satisfying the first clause. Such pairing creates double-stranded DNA. The remaining single-stranded molecules are those that do not satisfy the first clause, and these are destroyed by enzymes. The surface is then heated to melt away the complementary strands, washed and a fresh collection of Crick strands is paired with strings satisfying the second clause. This cycle is repeated for each of the remaining clauses. At the end, only those strands whose sequence satisfies the original formula survive.

In this system, the DNA ‘answers’ are attached randomly to the surface (rather than in an ordered array) so, to read out the answer, the surviving strands first have to be amplified using the polymerase chain reaction. Their identities are then determined by pairing with an ordered array of strings identical to the original set of sequences. Not counting the number of steps required to produce the DNA molecules in the first place, the algorithm takes only (3k + 1) steps, where k is the number of clauses, for a brute-force evaluation of all 2ⁿ possible answers. This represents a remarkable improvement over the best conventional...
news and views

Liu and colleagues’ DNA computer. These results are remarkable, but scaling up this technique to solve larger 3-SAT problems is still unrealistic. The authors claim that their surface-based approach is designed to scale up to problems of practical significance, but solving a problem that would embarrass a conventional computer is a long way off. First, there is the important issue of correcting errors arising from the inherent sloppiness of DNA chemistry, although ways of dealing with this issue have been suggested. Second, there is the high cost of tailor-made DNA sequences. For example, a 50-variable 3-SAT problem will require at least $10^{15}$ unique DNA strands. It remains to be seen whether massively parallel synthesis of DNA binary strings, using combinatorial chemistry, could further speed up the technique and reduce costs. One also needs to consider the non-trivial combinatorial chemistry, could further speed up the technique and reduce costs.

The most serious remaining issue is the exponentially increasing number of DNA molecules needed to compute even small 3-SAT problems. Alternative strategies that use self-assembly of complex DNA nanostructures also suffer from similar weaknesses. Perhaps a compromise may be achieved by reducing the search space through heuristics. An alternative may be to examine whether a surface-based attack on the more manageable P problems, such as Boolean circuit evaluation, would yield better returns in terms of speed and cost. If the DNA strings are spatially encoded and attached to an organic semiconductor surface, computation and read-out may also be combined.

It is foolish to attempt to predict the future of technology, but it may be that the ideal application for DNA computation does not lie in computing large NP problems. Some day there may be a need for fully organic computing devices implanted within a living body that can integrate signals from several sources and compute a response in terms of an organic molecular-delivery device for a drug or signal. Progress in DNA computing may pave the way in this direction.

Mitsunori Ogihara is in the Department of Computer Science, University of Rochester, Rochester, New York 14627-0226, USA.
e-mail: ogihara@cs.rochester.edu
Animesh Ray is at Akkadix Corporation, 11099 North Torrey Pines Road, Suite 200, La Jolla, California 92037, USA.
e-mail: ray@akkadix.com


Aerodynamics

From insects to microvehicles

Robin Wootton

Insects are the world’s smallest and in many respects most perfect flying machines. They can hover, fly slowly and manoeuvre with great precision. They can navigate through narrow gaps and around complex spaces, constantly receiving detailed information about their surroundings—but keeping it to themselves. How useful it might be, in surveillance, industrial fault location and so on, if we could build little machines with similar capabilities to detect and transmit information to us.

Advances in our understanding of insect aerodynamics, energetics and structural mechanics have brought this objective within reach. Collaborative research between engineers and specialists in insect flight is under way in at least three countries to develop micro-air-vehicles (MAVs) capable of this kind of investigatory flight in enclosed situations. Charles Ellington, of the Zoology Department, University of Cambridge, has published an intriguing account of the aspects of insect flight most applicable to MAV development, predicting how such a machine might be designed.

In technology, the rotary aerfoils of helicopters are the traditional solution to slow, manoeuvrable flight and stationary hovering, but at the insect/MAV scale flapping is a viable alternative. Both systems operate by accelerating a mass of air, predominantly downwards to support the weight, but with a smaller backward, forward or lateral component to generate forward, backward or sideways thrust as needed. To achieve this, insects oscillate and twist their wings, and typically vary the direction of the airflow by altering the angle between the plane of the wing stroke and the horizontal: the greater the angle, the faster the flight. The wings themselves deform semi-automatically, optimizing aerodynamic forces during the cycle.

Flight stability is maintained, and manoeuvres initiated, by finely controlling the details of the stroke. An asymmetric change in amplitude, or in the timing of twisting of the wings on one side of the body, can initiate a turn. An upward or downward shift of the line of action of the mean aerodynamic force can tilt the insect forward or backward, so altering the stroke-plane and in turn the speed.

Mimicking all these features in tiny vehicles presents many challenges: light enough materials; a miniature power plant; sophisticated controls; a controllable mechanism to flap and twist the wings at appropriate frequencies; and wings of appropriate shape, size and flexibility to maximize lift for minimum energy expenditure.

Two decades of research, much of it by Ellington and his group, have shown how insects make wide use of unconventional aerodynamic mechanisms which exploit the unsteady airflow resulting from the flapping motion to obtain far more lift than conventional, steady-state aerodynamics would allow. Some at least would be applicable to MAVs, particularly the ‘dynamic stall’ phenomenon: as the wing accelerates into the stroke, an intense vortex develops above the leading edge, and spirals outward towards the tip. This allows the wing to...