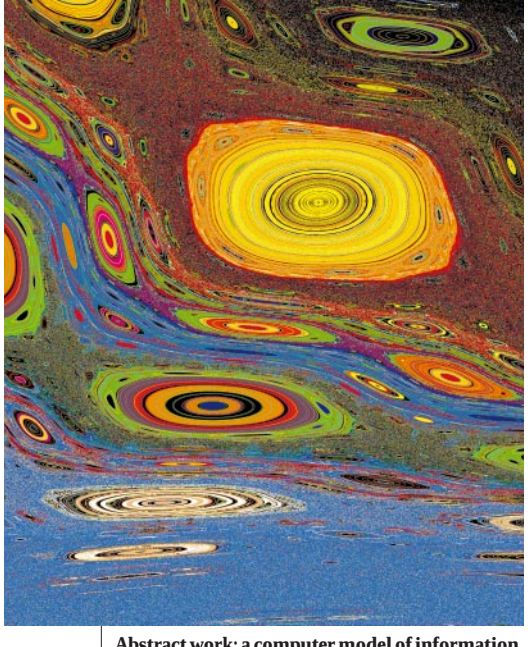
Cells as computation

Aviv Regev and Ehud Shapiro

Although we are successfully consolidating our knowledge of the 'sequence' and 'structure' branches of molecular cell biology in an accessible manner, the mountains of knowledge about the function, activity and interaction of molecular systems in cells remain fragmented. Sequence and structure research use computers and computerized databases to share, compare, criticize and correct scientific knowledge, to reach a consensus quickly and effectively. Why can't the study of biomolecular systems make a similar computational leap? Both sequence and structure research have adopted good abstractions: 'DNA-as-string' (a mathematical string is a finite sequence of symbols) and 'protein-as-three-dimensional-labelled-graph', respectively. Biomolecular systems research has yet to find a similarly successful one.

The hallmark of scientific understanding is the reduction of a natural phenomenon to simpler units. Equally important understanding comes from finding the appropriate abstraction with which to distill an aspect of knowledge. An abstraction — a mapping from a real-world domain to a mathematical domain — highlights some essential properties while ignoring other, complicating, ones. For example, classical genetic analysis uses the 'gene-as-hereditary-unit' abstraction, ignoring the biochemical properties of genes as DNA sequences. A good scientific abstraction has four properties: it is relevant, capturing an essential property of the phenomenon; computable, bringing to bear computational knowledge about the mathematical representation; understandable, offering a conceptual framework for thinking about the scientific domain; and extensible, allowing the capture of additional real properties in the same mathematical framework.



Abstract work: a computer model of information transfer through a network of coloured points.

For example, the DNA-as-string abstraction is relevant in capturing the primary sequence of nucleotides without including higher- and lower-order biochemical properties; it allows the application of a battery of string algorithms, including probabilistic analysis using hidden Markov models, as well as enabling the practical development of databases and common repositories; it is understandable, in that a string over the alphabet A, T, C, G is a universal format for discussing and conveying genetic information; and extensible, enabling, for example, the addition of a fifth symbol denoting methylated cytosine.

We believe that computer science can provide the much-needed abstraction for biomolecular systems. Advanced computer science concepts are being used to investigate the 'molecule-as-computation' abstraction, in which a system of interacting molecular entities is described and modelled by a system of interacting computational entities. Abstract computer languages, such as Petrinets, Statecharts and the Pi-calculus, were developed for the specification and study of systems of interacting computations, yet are now being used to represent biomolecular systems, including regulatory, metabolic and signalling pathways, as well as multicellular processes such as immune responses. These languages enable simulation of the behaviour of biomolecular systems, as well as development of knowledge bases supporting qualitative and quantitative reasoning on these systems' properties.

Processes, the basic interacting computational entities of these languages, have an internal state and interaction capabilities. Process behaviour is governed by reaction rules specifying the response to an input message based on its content and the state of the process. The response can include state change, a change in interaction capabilities, and/or sending messages. Complex entities are described hierarchically — for example, if *a* and *b* are abstractions of two molecular domains of a single molecule, then (*a* parallel *b*) is an abstraction of the corresponding two-domain molecule. Similarly, if *a* and *b* are abstractions of the two possible behaviours of a molecule in one of two conformational states, depending on the ligand it binds, then (*a* choice *b*) is an abstraction of the molecule, with the choice between *a* and *b* determined by its interaction with a ligand process.

Cellular abstractions

Computer science could provide the abstraction needed for consolidating knowledge of biomolecular systems.

Using this abstraction opens up new possibilities for understanding molecular systems. For example, computer science distinguishes between two levels to describe a system's behaviour: implementation (how the system is built, say the wires in a circuit) and specification (what the system does, say an 'AND' logic gate). Once biological behaviour is abstracted as computational behaviour, implementation can be related to a real biological system, for example the detailed molecular machinery of a circadian clock, and the corresponding specification to its biological function, such as a 'black-box' abstract oscillator. Ascribing a biological function to a biomolecular system is thus no longer an informal process but an objective measure of the semantic equivalence between low-level and high-level computational descriptions. Equivalence between related implementations in different organisms can also be a measure of the behavioural similarity of entire systems, complementary to sequence and structure similarity.

Computer and biomolecular systems both start from a small set of elementary components from which, layer by layer, more complex entities are constructed with ever-more sophisticated functions. Computers are networked to perform larger and larger computations; cells form multicellular organisms. All existing computers have an essentially similar core design and basic functions, but address a wide range of tasks. Similarly, all cells have a similar core design, yet can survive in radically different environments or fulfil widely differing functions. Of course, biomolecular systems exist independently of our awareness or understanding of them, whereas computer systems exist because we understand, design and build them. Nevertheless, the abstractions, tools and methods used to specify and study computer systems should illuminate our accumulated knowledge about biomolecular systems. ■

Aviv Regev and Ehud Shapiro are in the Department of Computer Science and Applied Mathematics, and the Department of Biological Chemistry, Weizmann Institute of Science, Rehovot 76100, Israel.

FURTHER READING

www.wisdom.weizmann.ac.il/~aviv

Milner, R. *Communicating and Mobile Systems: The Pi-calculus.* (Cambridge Univ. Press, 2000).

Fontana, W. & Buss, L. W. in *Boundaries and Barriers* (eds Casti, J. & Karlqvist, A.) 56–116 (Addison-Wesley, New York, 1996).

© 2002 Nature Publishing Group