## Biochemical computers

Biochemical computers use the immense variety of feedback loops that are characteristic of biological chemical reactions in order to achieve computational functionality. Feedback loops in biological systems take many forms, and many different factors can provide both positive and negative feedback to a particular biochemical process, causing either an increase in chemical output or a decrease in chemical output, respectively. Such factors may include the quantity of catalytic enzymes present, the amount of reactants present, the amount of products present, and the presence of molecules that bind to and thus alter the chemical reactivity of any of the aforementioned factors.

Given the nature of these biochemical systems to be regulated through many different mechanisms, one can engineer a chemical pathway comprising a set of molecular components that react to produce one particular product under one set of specific chemical conditions and another particular product under another set of conditions. The presence of the particular product that results from the pathway can serve as a signal, which can be interpreted—along with other chemical signals—as a computational output based upon the starting chemical conditions of the system (the input).

## Biomechanical computers

Biomechanical computers are similar to biochemical computers in that they both perform a specific operation that can be interpreted as a functional computation based upon specific initial conditions which serve as input. They differ, however, in what exactly serves as the output signal. In biochemical computers, the presence or concentration of certain chemicals serves as the output signal. In biomechanical computers, however, the mechanical shape of a specific molecule or set of molecules under a set of initial conditions serves as the output. Biomechanical computers rely on the nature of specific molecules to adopt certain physical configurations under certain chemical conditions. The mechanical, three-dimensional structure of the product of the biomechanical computer is detected and interpreted appropriately as a calculated output.

## Bioelectronic computers

Biocomputers can also be constructed in order to perform electronic computing. Again, like both biomechanical and biochemical computers, computations are performed by interpreting a specific output that is based upon an initial set of conditions that serve as input. In bioelectronic computers, the measured output is the nature of the electrical conductivity that is observed in the bioelectronic computer. This output comprises specifically designed biomolecules that conduct electricity in highly specific manners based upon the initial conditions that serve as the input of the bioelectronic system.

## Network-based biocomputers

In networks-based biocomputation , self-propelled biological agents, such as molecular motor proteins or bacteria, explore a microscopic network that encodes a mathematical problem of interest. The paths of the agents through the network and/or their final positions represent potential solutions to the problem. For instance, in the system described by Nicolau et al. , mobile molecular motor filaments are detected at the "exits" of a network encoding the NP-

complete problem SUBSET SUM. All exits visited by filaments represent correct solutions to the algorithm. Exits not visited are non-solutions. The motility proteins are either actin and myosin or kinesin and microtubules. The myosin and kinesin, respectively, are attached to the bottom of the network channels. When adenosine triphosphate (ATP) is added, the actin filaments or microtubules are propelled through the channels, thus exploring the network. The energy conversion from chemical energy (ATP) to mechanical energy (motility) is highly efficient when compared with e.g. electronic computing, so the computer, in addition to being massively parallel, also uses orders of magnitude less energy per computational step.

# Engineering biocomputers

A ribosome is a biological machine that uses protein dynamics on nanoscales to translate RNA into proteins. The behavior of biologically derived computational systems such as these relies on the particular molecules that make up the system, which are primarily proteins but may also include DNA molecules. Nanobiotechnology provides the means to synthesize the multiple chemical components necessary to create such a system. The chemical nature of a protein is dictated by its sequence of amino acids—the chemical building blocks of proteins. This sequence is in turn dictated by a specific sequence of DNA nucleotides—the building blocks of DNA molecules. Proteins are manufactured in biological systems through the translation of nucleotide sequences by biological molecules called ribosomes, which assemble individual amino acids into polypeptides that form functional proteins based on the nucleotide sequence that the ribosome interprets. What this ultimately means is that one can engineer the chemical components necessary to create a biological system capable of performing computations by engineering DNA nucleotide sequences to encode for the necessary protein components. Also, the synthetically designed DNA molecules themselves may function in a particular biocomputer system. Thus, implementing nanobiotechnology to design and produce synthetically designed proteins—as well as the design and synthesis of artificial DNA molecules—can allow the construction of functional biocomputers (e.g. Computational Genes).

Biocomputers can also be designed with cells as their basic components. Chemically induced dimerization systems can be used to make logic gates from individual cells. These logic gates are activated by chemical agents that induce interactions between previously non-interacting proteins and trigger some observable change in the cell.

Network-based biocomputers are engineered by nanofabrication of the hardware from wafers where the channels are etched by electron-beam lithography or nano-imprint lithography. The channels are designed to have a high aspect ratio of cross section so the protein filaments will be guided. Also, split and pass junctions are engineered so filaments will propagate in the network and explore the allowed paths. Surface silanization ensures that the motility proteins can be affixed to the surface and remain functional. The molecules that perform the logic operations are derived from biological tissue.