Mathematical Modelling, Vol. 7, pp. 1513-1577,1986

COMPUTER MODELS AND AUTOMATA THEORY IN BIOLOGY AND MEDICINE:

COMPUTER SIMULATION AND COMPUTABILITY OF BIOLOGICAL SYSTEMS

Ion C. Baianu and updated by Hsiao Chen Lin (2004)
University of Ulinois at Urbana
Physical Chemistry and NMR Laboratories
567 Bevier Hall, 905 S. Goodwin Ave
Urbana, Illinois 61801

(Received 27 *February* 1985; *revised* 12 *September* 1985,)

The ability to simulate a biological organism by employing a computer is related to the ability of the computer to calculate the behavior of such a dynamical system, or the "computability" of the system.* However, the two questions of computability and simulation are not equivalent. Since the question of computability can be given a precise answer in terms of recursive functions, automata theory and dynamical systems, it will be appropriate to consider it first. The more elusive question of adequate simulation of biological systems by a computer will be then addressed and a possible connection between the two answers given will be considered.

A. *Are biological systems recursively computable?*

An answer to this question was recently given by Conrad and Rössler [219] who showed that although a system can be computation universal it may not be effectively programmable if its translator has "chaotic" dynamics; such chaotic dynamics were encountered in certain models of biomolecular reaction kinetics [220]. At this point, let us introduce the concepts of recursive function, recursive computer, computation and program in order to be able to formally discuss recursive computability of a system, be it biological or nonbiological.

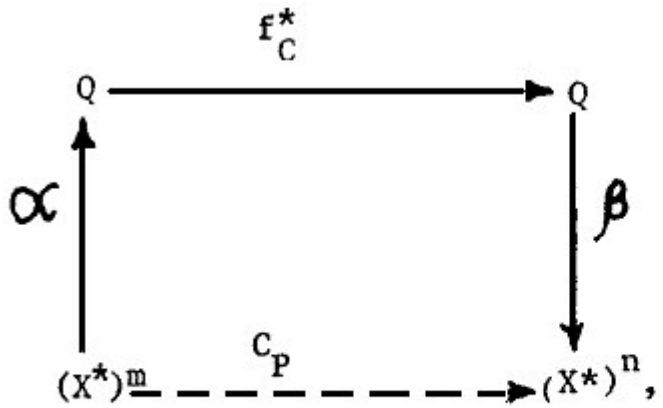
A function is called *recursive* if there is an effective procedure, or computation, for calculating it (p. 211 in Ref. [155]).

* The contributions made to this section by an anonymous referee are gratefully acknowledged.

A *recursive computer* C [155] on the alphabet *Y* is a partial recursive function $f_c : Y_b^* \to Y_b^* . Q = Y_b^*$ is called the set of complete states of C. (The qualifier "partial" refers to the fact that the recursive function may not be defined for all its values.) If at an instant *t* the computer has *n* registers, the *j*th register containing the word Y_j, one can say that the *complete state* of the computer is the word $Y_{1b} ... bY_n$ of *Q;* also if the state of the computer C is $\xi \in Q$ at time *t,* the state of C will be $f_C(\xi)$ at time $(t + 1)$, and may be therefore undefined, although not necessarily so. The computer is in a halting state $\xi \in Q$ when $f_C(\xi) = \xi$.

A *computation* by the computer C [155] is defined as any finite sequence $\xi_0, \xi_1, \xi_2, \xi_3 ..., \xi_n$ of elements of *Q* such that $\xi_{j+1} = f_C(\xi_i)$ for any $j < n$ and $j \geq 0$; $\xi_n = f_C(\xi_n)$ and $f_C^*(\xi)$ is defined as the end result of a computation starting with $\xi \in Q$. If f_C is total, the end result of a computation is unique. and is always defined.

A program *P* for the computer C is defined [155] as a pair of mappings $\alpha: (x^*)^m \to Q$, $\beta: Q \to (x^*)^n$, which are partial recursive on the union set $(X \cup Y)h$. α is used to read in the data and appropriate instructions into memory, which will continue until the computation stops; then β is used to "read" the results from the computer registers. The partial recursive function *f~* defining the computer is related to a, 13, and a partial function $Cp: (x^*)^m \to (x^*)$, which is said to be "computed by C with the program *P*". This relationship is given by the following commutative diagram, which also defines *Cp:*

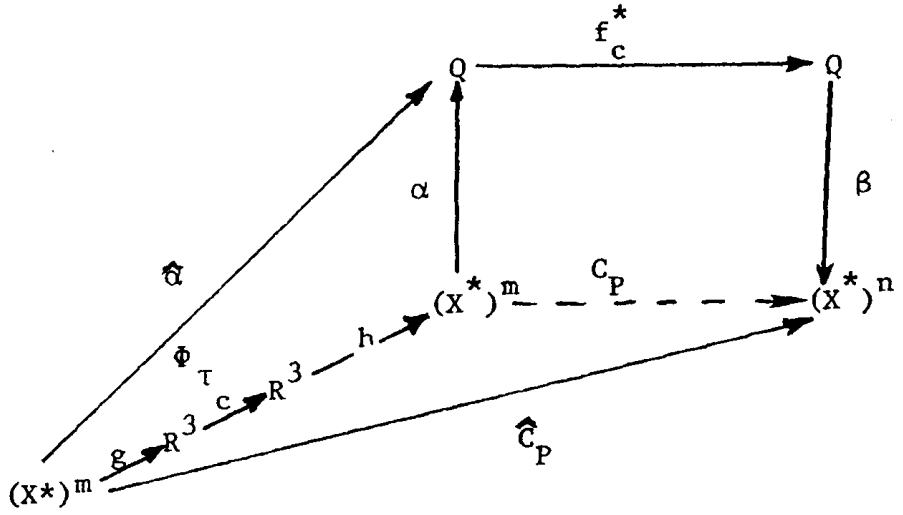


(39a)

That is, $Cp = \alpha \circ f_c^* \circ \beta$

The question at the beginning of this section is, therefore, if the function $Cp = \alpha \circ f_c^* \circ \beta$ is sufficient to mimic or simulate the dynamics of any biological system. The argument introduced by Conrad and Rössler [219] is that the read-in function should be a translator with "chaos" in order to be able to simulate certain biomolecular reactions in a biological system. Thus, suppose that one programs a digital computer in the usual sense; this can be represented by the commutative diagram (39a) above, in which *Q* is the state set, or "state space", of the

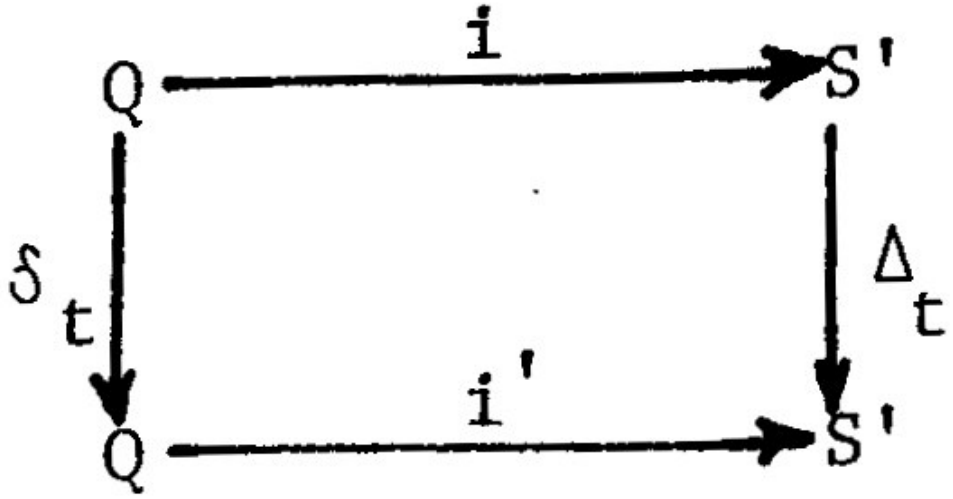
computer and with the functions α and β being used by the programmer to implement the function *Cp.* The idea of the Conrad-Rössler paper [219] was to interpose a "chaotic translator", between the programmer and the computer; such a chaotic translator can be constructed, or envisioned, as follows. Let *g* be a map $X \longrightarrow R^3$ (with R^3 being the three-dimensional Euclidean space), and let *h:* $R^3 \longrightarrow X$ be a (left) inverse of *g,* that *is,* $h \circ g = 1_X$, *with* 1_X being the identity map $X \longrightarrow X$ and " $\circ$ " the usual map composition law. Furthermore, let us consider a dynamical flow induced by the Lorenz equations $\Phi_{\tau c}: R^3 \longrightarrow R^3$, that *is,* with $\Phi_{\tau c}$ *(x, y, z)* being defined as the point *(x', y', z')* to which the system flows after the interval of time τc if started at the point *(x, y, z).* When coupled to the programming diagram (39a) such a dynamical flow induces the following representation:



Because arbitrarily small differences will produce permanently distinct aperiodic trajectories, there is no single partial recursive function f^*_c that will allow the computer to calculate the encoding function $\hat{\alpha}$ if $\tau c > Q \mid \Delta t$. Therefore, a system that does possess a dynamical flow which is induced by the Lorenz equations would not be recursively computable.

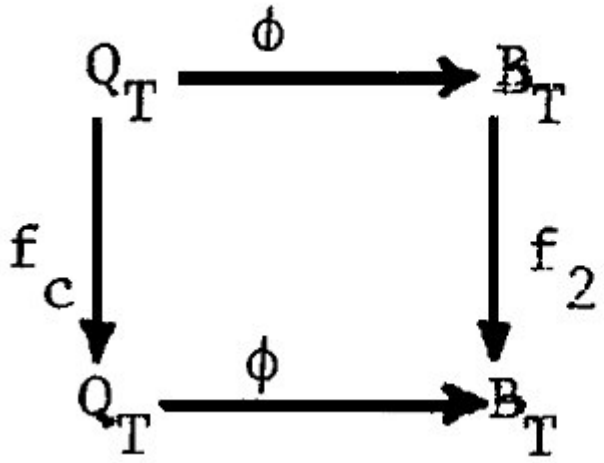
B. *Can One Simulates All Biological Systems with Computers?*

This important question of universal *computation* of biological *systems* can be answered uniquely only by giving an appropriate definition of *the concept* of "simulation". If one defines formally "computer simulation" *to* be *restricted to* (partial) recursive *computation* of biological dynamics by a digital computer, then *it* would follow from Sec. 8A *that* such a simulation is *not* generally possible for biological systems. However, if one employs dynamical analogy [221] *to* define simulation, then the computer prospects for simulation may appear somewhat brighter since one can now define a class of *systems S' that* are analogous to a computer *Q,* as represented by the commutative diagram



(40)

by selecting i and i' to be isomorphisms. If the computer is digital, then $\delta_t = f_c^*$ is the partial recursive function defined in Sec. 8A that will allow one to compute, or "simulate", the behavior of a certain class of dynamical systems. Such recursively computable systems may be certain components or subsystems of a biological system that do not incorporate any chaotic behavior. Furthermore, if one were to lift the restriction that Q be a discrete set, and define instead a new kind of "computer" with a topological state space (incorporating chaos), then a dynamic analogy can be considered between such a novel "topological computer" and an intact biological system. This type of topological computer may include, for example, topological automata that have topological semigroups [222] for state spaces instead of algebraic semigroups as the conventional automata or sequential machines [155]. The *analogy* will be replaced in this case by *conjugacy* (p. 508 in Ref. [223]) as a form of similarity between dynamical systems. A topological computer would thus be endowed with a topological semigroup and could simulate a biological system if its computation function f_c, was topologically conjugate [222] to the transition function f_2 of the biological system ($f_c \tilde{\Phi} f_2$), that is, $\Phi f_2(X) = f_c(\Phi(x))$, $x \in T$, with Φ being a homeomorphism



(41)

Topological conjugacy ensures that if $f_2 \tilde{\Phi} f_c$ for some given parameter set, then the functions f_2 and f_c have equivalent dynamical behavior over that parameter set. The general case of state spaces with both algebraic and topological structure was also considered by means of *adjointness of functors* defining dynamical equivalences between systems possessing such complex state spaces [224].

CONJECTURE. There is a universal simulation of biological systems by an algebraic-topological "computer," by means of a pair of adjoint functors which defines the dynamical equivalence of the computer with any selected biological system.

9. COMPUTER SIMULATIONS AND ALGEBRA OF GENETIC NETS

A. *Binary Genetic Nets*

Metabolic stability and epigenesis in genetic systems were modelled by Kauffman with formal' "binary genes", [160]. Such models are particularly applicable to bacterial genetic control systems [225]. Kauffman found that large genetic nets of binary elements, which have very 1arge numbers of possible states, exhibit remarkably ordered and stable behavior without requiring any special structural organization. Such "genetic nets" enter rapidly their stable limit cycles in their state space and have very few different limit cycles.

A Kauffman (genetic) net of size N and connectivity k consists of N interconnected binary elements (Fig. 35), each having k inputs and only one output. The elements of a Kauffman net were described [160] as "formal genes". The inputs and outputs of each formal gene take only one of the two possible values denoted by 0 (zero) and 1 at any instant in time t. The interconnection of formal genes is random, each input is connected to one and only one output and each input has an equal chance of being connected to any output. Therefore, as in the case of neural binary networks, the input-output relation of each formal gene is defined by one of 2^{2k} possible Boolean mappings of k binary variables. Furthermore, the assignment of Boolean mappings is such that each element in the net has an equal chance of being assigned any of the 2^{2k} Boolean mappings.

The state of a Kauffman net is specified at any instant in time by the states of the N element outputs, which means that this genetic net has $2N$ distinct states. In terms of automata theory (see Sec. 10) the Kauffman net is a deterministic, "autonomous", finite state sequential machine; the machine is autonomous in the sense that there are no external inputs to the genetic net [173]. Although the Kauffman net is random, its state transitions are not probabilistic events. Kauffman [160] also imposed the restriction that the network is in some well-defined state at time t, and operates on a discrete time scale, like a clock with a period t. The behavior of a Kauffman net can be also represented in the form of a state transition diagram (Fig. 36), which is a directed graph whose vertex set is the state space of the net, and whose arcs are the state transitions.

Kauffman [160] used computer simulation to study the behavior of genetic binary nets with low connectivity (k = 2 or 3), and numbers of elements between 15 and 8192. For k = 2, Kauffman found that the number of limit cycles r of a net with N elements was $r = N^{a}$. with a =

0.5, and that the median limit cycle length was $r = N^b$, with $0.3 < b < 0.6$. When $N = 1000$, $r = 30$, but the number of possible states is $n = 2^N - 10^{300}$, which is astronomical. Nets with such a large number of states entered very quickly a very small number of very short cycles. Sherlock [173] proposed an analytical description of Kauffman nets in terms of the state transition matrix. An example of such a matrix is reproduced from Ref [173] in fig37. The matrix provides a complete specification of the state transition graph.

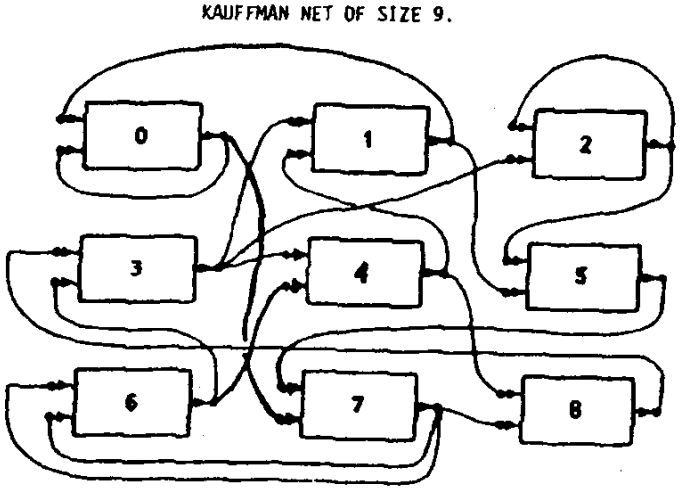


Fig. 35. A Kauffman net of connectivity $k = 2$ and size $N = 9$ (according to Sherlock[173]).

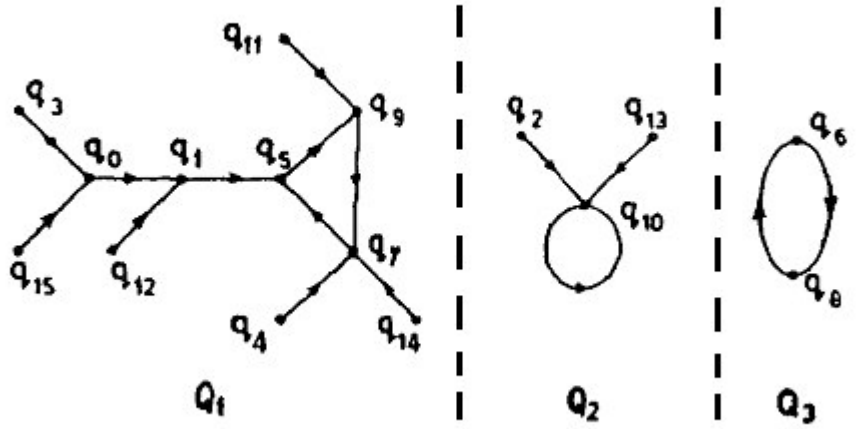


Fig. 36. A possible state transition diagram for a $N = 4$ Kauffman net (from Ref. [173]).

The states of the Kauffman net can be represented as vectors: $q(t+\tau) = Aq(t)$ where A is the transpose of the adjacency matrix [187] specifying the state transition digraph; in terms of the digraph, A is defined as

$$A = \{a_{ij}\}, \qquad a_{ij} = 1 \tag{42}$$

if there is an arc from the vertex *j* to the vertex i, and $a_{ij} = 0$ otherwise. This matrix form is convenient for an algebraic analysis of the dynamics of the genetic nets. The representation also has the advantage that the matrix elements a_{ij} of *A* can be calculated from a specified "wiring diagram" (Fig. 35) of the network, together with the assignment of the Boolean mappings to the network elements. With the matrix representation the number of states involved in the limit cycles are readily calculated [173] through matrix element manipulation. Specifically, the fractional distribution $f(l)$ of limit cycle lengths is

$$f(l) = \sum_{j[N_c]} n_l^j \left(\prod_{l=1}^{N_c} l^{n_l^j} n_l^j ! \right)^{-1}, \tag{43}$$

where n_l^j is the number of occurrences of integer *l* in the partition *j* of N_c into the sum $N_c = \Sigma n_l^j$ which corresponds to a partitioning of Q_c into any of the sets of disjoint limit cycle subsets C_J that give rise to n_1 stable states. n_2 cycles of length 2. etc. The number of occurrences of cycles of length *1* was calculated [173] as

$$M(l) = N_c f(l) \tag{44}$$

where N_c is the number of different of distributing N_c distinct objects among the partitions. Using a computer algorithm, Sherlock [173] evaluated numerically $f(l)$ for N_c all up to 20. Wotjom tje 24 bit (~ seven decimal digit) precision of the digital computer, Sherlock [173] found that

$$f(l) = 1/l, \tag{45}$$

for $1 \leqslant l \leqslant N_c$, $N_c \leqslant 20$. This *1/1* distribution of cycle lengths is also observed in a computer simulation of Kauffman nets for cycle lengths up to 50 (Fig. 7 on p.699 in Ref. [173]). Sherlock [173] also found that $N_c = L^2$, where *L* is the median cycle length and, therefore, with $L \simeq \sqrt{N}$ [160] he obtained $N_C \simeq N$; that is, for $k = 2$ random nets with *N* elements the cardinality of the cycle state set is approximately equal to *N*. With this value of N_c .the fraction F_c of the total number of states involved in the limit cycle is

$$F_c = \frac{N_c}{n} \simeq \frac{N}{2^N}. \tag{46}$$

This relation is in agreement with Kauffman's "experimental" findings [l60] by computer simulation of formal genetic nets. However, for connectivities higher than 2, the fraction F_c

calculated algebraically was no longer $\sim N / 2^N$, as in the computer experiments (simulation of random nets). The deviation from the behavior obtained by Kauffman for $k = 2$ increased rapidly with the size of the net and, therefore, it is possible that biologically relevant models of genetic systems may not have $F_c = N / 2^N$ (see Fig. 7 on p. 722 in Ref. [173]), but may have instead $\log F_c \sim (- N)$. Even more complex behavior is expected for n-ary genetic nets; such nets are discussed next.

B. *N-ary Genetic Networks*

A genetic network, or net, is defined as an assembly or aggregate of interacting genes. Genetic interactions can be either direct, as in the case of gene "clustering", or they can be indirect - via intermediates, products and metabolic pathways. Perhaps the best known simple example of a genetic net is the one introduced by Jacob and Monod [225] for the genes related to lactose metabolism in the Gram - negative bacterium *Escherichia coli.* Such genes were shown to lie near one another in the same region of the bacterial chromosome and were considered to act as a functional unit, or "operon". Within the operon, a "regulator" gene, three "structural" genes and an "operator" gene were postulated to exist and to play different functional roles. The three structural genes are under the control of the same operator gene, while the operator gene itself responds to, or is controlled by, the regulator gene. In the case of two interacting operons, which define a two-component genetic net, the situation can be represented schematically as in Fig 38.

Let us consider the possible dynamics of the genetic network in Fig. 38. If we postulate discrete states for the genes, then one can readily calculate the dynamics for "all-or- none" type of genes (that is, the case in which the genes are either fully active or completely inactive). The more general case of *n* discrete states, which are defined by *n* discrete levels of gene activities, is more difficult to treat and requires the introduction of special

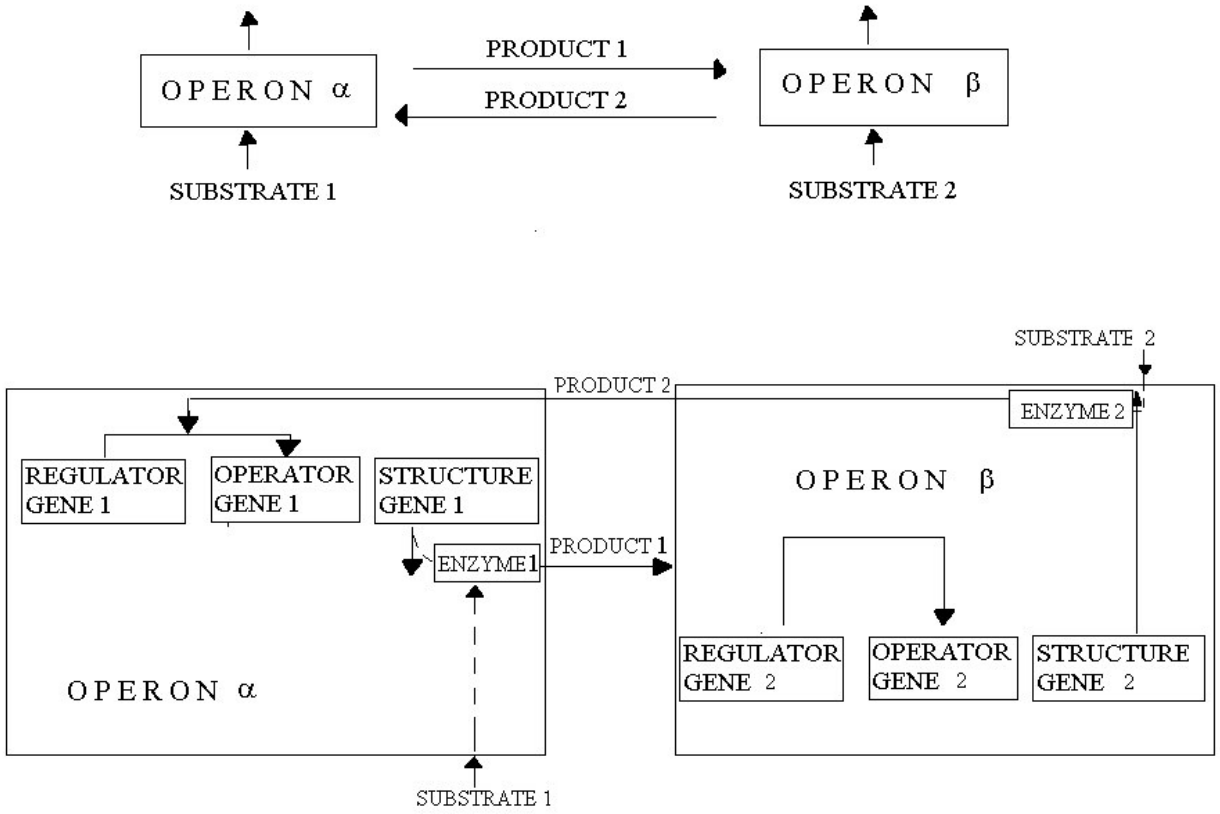


Fig. 38. Two-component genetic net with two operons IX **and** /3 **(from Ref. [226]).**

mathematical concepts. A general, two-component genetic net with *n* states is presented in Fig. 39 and the generalization to *m-component genetic nets* is now transparent. This n-state black-box representation can be used in principle to determine the behavior of any genetic net. However, it has the disadvantages of being tedious and limited in its scope for deriving general theorems concerning genetic network dynamics. An alternative approach was, therefore, introduced in which genetic nets and their activities are defined in terms of n-valued logic and their associated Lukasiewicz algebras [226]. Mainly because the *n* levels of activity of a gene are ordered (in the sense that the lowest level corresponds to no activity or 0, while the higher levels or corresponding states are in the order of increasing activity up to full activity which is labeled by I), the statements concerning such genetic activities are also ordered under a *symbolic ordering relation* $\subset$ forming therefore a distributive lattice of a special kind. Such a distributive lattice is called an *n- valued Lukasiewicz algebra* (or *L-algebra).* Certain maps $\sigma_i : L \longrightarrow L$ and $N: L \longrightarrow L$, which are endomorphisms of an L-algebra associated with an n-state genetic net [226] are employed to model tile dynamics of genetic systems in the absence of mutations. In order to represent genetic mutations certain transformations of L-algebras are defined. These are morphisms $f: L_1 \longrightarrow L_2$ of L-algebras and possess certain specific properties such as

$$f*N = N*f \quad \text{and} \quad f*\sigma_i = \sigma_i*f \quad \text{for any } i = 0, 1, \ldots, n-1 \qquad (47)$$

(see also details on p. 254 of Ref. [227]). All "genetic" L-algebras together with the possible L-morphisms representing genetic mutations are then forming 11 *subcategory* G_L of the *category*

Luk$_n$ of all n-valued Lukasiewicz algebras and L-morphisms between them. The basic properties of G_L were reported in Ref. [226]. In the particular case of two-state genes, the n-valued algebras are reduced to Boolean ones and the category G_L^B of such Boolean algebras is equivalent to a special subcategory called the category of centered L-algebras G_L^C. This equivalence between G_L^B and a full subcategory of G_L^C is expressed by two *adjoint functors* C and D:

$$G_L \xrightarrow{C} G_L^B \xrightarrow{D} G_L, \tag{48}$$

with the functor C being full and faithfull (see also p. 254 in Ref. [226] and references cited therein). The *centers* of a centered Lukasiewicz algebra are *(n* -2) elements (statements about gene activities), $a_1, a_2, \ldots, a_{n-1} \in L$, for which

$$\sigma_i(a_j) = \begin{cases} 0 & \text{for } 1 \leqq j < n - j, \\ 1 & \text{for } n - j < i \leqq n - 2. \end{cases} \tag{49}$$

The corresponding genes would have degenerate states in the sense that *(n* -j) states will all be characterized by no activity while the remaining states will yield full activity. An organism at a given stage of its development may have some genes of the all-or-none type, as well as some nondegenerate ones. Therefore, the collection of genes of an organism, or its corresponding genetic net, will be generally represented by a mixture of centered Ł-algebras and noncentered ones. The corresponding category of genetic nets of an organism and their dynamic transformations will be a subcategory of G_L , and therefore of *Lukn,* but will not be in general a subcategory of G_L^B, or *of BI,* the category of Boolean algebras. A simple example of a nondegenerate genetic net is provided by the multiple genes which were discovered in the lampbrush chromosomes in oocytes of the crested newt *Triturus cristatus carnifex* [228], after we postulated the genetic n-states [226]. The transcription of multiple genes can be readily represented as a particular dynamic process of an *m-state* genetic network with its 0 state corresponding to all multiple genes being inactive and the 1 state corresponding to all *m* genes being fully operational. There is strong experimental evidence that the transcription of multiple genes, or "satellite" DNA, on lampbrush loops in oocytes does indeed occur and correlates somehow with structural heteromorphism of the organism, as well as a lack of chiasma formation in the chromosome arm in *Triturus cristatus* [228]. The evolutionary significance of this "unusual" transcription of satellite DNA appears obscure at present, and further hybridization experiments involving *DNA /* nascent RNA transcripts from clone pTcS 1 will be necessary in order to determine the transcription mechanism in such cases. Another particularly interesting example of n-state transcription units was recently reported for the rDNA gene sequences of *Physarum* [229]. In this case, the subunits on the ribosomal gene transcription unit were shown to be in a molecular configuration which was distinct (in some unspecified way) from the oligomers of the nucleosomes present on the inactive rDNA regions. Different chromatin subunit structures could be isolated from different functional regions of a single gene [229]. These experiments open up the exciting possibility of carrying out detailed kinetic studies

of individual genetic activities and should allow for a direct comparison with the theoretical predictions based on abstract molecular models of such genetic systems. In particular, the polymerase binding rates and certain transient conformational changes of nucleosomes on functional genes could be derived from computer simulations.

10. AUTOMATA THEORY AND METABOLIC-REPLICATION MODELS IN BIOLOGY

Before considering the applications of automata theory in biology we shall introduce the basic concepts needed for such models.

A. *Algebraic Theory of Automata* [230, 155]

An *automaton* is a quintuple $M = (X, Y, Q, \delta, \lambda)$, where X is a finite input alphabet, or set of inputs $\{a_1, a_2, ..., a_m\}$, Y is a finite set of outputs $\{Y_1, Y_2, ..., Y_n\}$, Q is a finite set of memory states $\{q_1, q_2, ..., q_k\}$, δ: $Q \times X \rightarrow Q$ is the next state (transition) function and λ: $Q \times X \rightarrow Y$ is the output function.

Automata can also be represented by state transition graphs, similar to those of random nets, or by Boolean matrices and output vectors. A more powerful, algebraic representation of an automaton is its description as a semigroup of state transition maps; the state transition maps associated with single-input symbols can be multiplied to find the state transition maps produced by input sequences (strings of inputs). The set of state transition maps of the automaton when endowed with the multiplication operation of the transition maps has a semigroup structure. The algebraic theory of automata which deals with such semigroup structures is useful for solving the following problems:

1. Automata decomposition into the simplest, "irreducible" components;
2. Finding the standard version of any automaton;
3. Classifying automata according to their most general and essential properties.

B. *Basic Theorem for Decomposition of Automata*

The state transition maps are of two basic types:
(a) collapsers-two distinct states are carried onto the same next state under the input;
(b) permutations'"-states are being only permuted under such maps; the permutations are I-I onto functions of the state set.

Krohn, Rhodes and Tilson [231, 232] have found the basic decomposition theorem of automata: *any automaton can be decomposed into irreducible components that have either only permutations or are amongst four types of elementary collapser automata.* Per- mutations form a group and, therefore, *the automata decomposition theorem states that any automaton, or "machine" semigroup, can be decomposed into a group and collapser semigroups of four basic types .*

C, *Tessellation Automata and Biological Development*

Cellular automata have been used as models for the development and growth of biological organisms. An example, provided by Arbib [233] will be used to illustrate this approach. In a similar vein is the preceding work of Smith [234], Wagner [235] and Arbib [236].

A. "module" equipped with 22 instructions, a bit register. BR, and a number of inputs is the basic unit (Fig. 10.1 on p. 352 in Ref. [233]) of a tessellation structure. Copies of the module are placed at a lattice point, in a planar configuration. An array of modules defines the tessellation. Directions within the tessellation are defined using the coordinates *(m, n)* for each module:

u = up,	defined by increasing *n;*
d = down,	defined by decreasing *n;*
r = right,	defined by increasing *m;*
1 = left,	defined by decreasing *m.*

For example, a group of four neighbor modules, or "cells", in such a tessellation structure will have the coordinates (m-1, n), (m, n), (m, n -1), and (m + 1, n). The tessellation is a "tissue-like" structure where a collection C of cells is *comoving* if $\alpha \varepsilon C \Rightarrow \beta \varepsilon C \Leftrightarrow W(\alpha, \beta)$, with *W* being the relation of "welding" on cells: $W(\alpha, \beta) \Leftrightarrow \alpha, \beta$ are welded neighbor cells, that is, $\exists \gamma$ such that $W(\alpha, \gamma)$ and $W(\gamma, \beta)$. The neighboring cells are said to be welded if either of the cells is joined to its neighbor, so that they "change" their position together in the array; "moving a cell" in direction *x* means in fact moving the contents of the cell registers in the *x* direction. The commoving set, therefore, is really a pattern rather than a set of cells in the tessellation. The basic idea is that if any cell moves in one direction all the cells in the commoving set move in the same direction. Arbib [233] showed that any finite automaton or Turing machine can be simulated by a tessellation structure. Tessellation structures can be used not only to generate new cell configurations but also to construct automata by acting on neighboring squares. A tessellation structure capable of constructing and simulating a Turing machine was called a "CT-machine"[233]. Arbib proved the existence of a CT which is also self-reproducing (p. 370 in Ref. [233]). Figure 40 shows such a CT -machine, with *p* being its program and _____ being its tape configuration; the reproduced machine (_____), which is the "off- spring", also includes the program to reproduce itself.

Such abstract constructions were designed to address the question: how can a complex, multicellular organism grow from a single cell assumming that each cell behaved like an automaton and executed a finite program?

1561

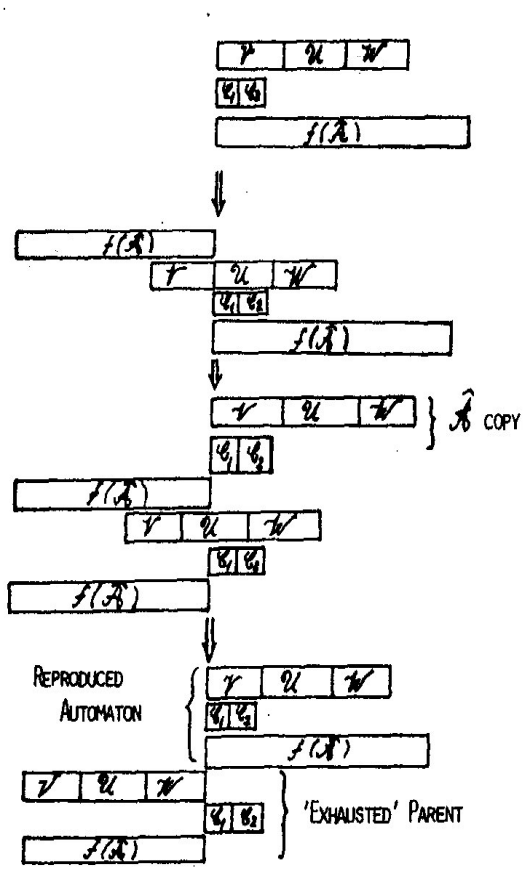


Fig. 40. Model of self-reproduction in the form of a CT-machine (according to Arbib[233]).

The process represented in Fig. 40 shows how a multicellular organism can reproduce itself but it does not indicate how a multicellular organism can grow and develop from a single cell.
The tessellation model presents a number of differences from the behavior of a real biological organism: the program of the machine is contained in a string of automata rather than in a single cell, the specification of the machine is complete within each module but it is incomplete in a biological cell, and the construction assummes that the modules are passive, in the sense that any subassembly will remain fixed and inactive until the whole tessellation structure is complete. Biological development, however, is an active process, generated through the multiple interactions and "induction" between the subassemblies. In the tessellation model, growth is simulated by an increase in the number of cells in a comoving set; cell division is mimiced by modular activity at the construction site in the tessellation. When a module "divides" the original module is preserved at the initial site, while its replica has been produced at a neighboring location in the lattice. In such a model, a module $\hat{c}(A)$ produces initially a copy of itself which is rendered "dormant", that is nondividing. Then, $\hat{c}(A)$ produces $c(A)$ which builds A, and before A is turned loose, it is attached to the copy of $c(A)$. Upon reaching "maturity", module A reproduces by releasing such a copy $\hat{c}(A)$ into the tessellation. The design of a differentiated organism would be, however, much more complex than such a tessellation process; one would have to use subroutines to build differentiated tissues and employ higher-level routines to assemble such differentiated tissues in the correct anatomical position.

Models employing molecular automata rather than cellular ones may be closer to the molecular biology approach to differentiation, growth and development. A category of such models comprises metabolic-'replication, or *(M, R),* systems and was designed by Rosen [150].

D. (M, *R)-Systems*

The simplest (M, R) system [151] is defined by a set of mappings H (A, B) of the form f: A → B, together with a set of mappings H (B, H (A, B)) of the form Φ: B → H (A, B), such that Φ_f (b) = f for f (a) = b. A is the set of inputs and B is the set of outputs of the *(M, R)* system. By defining the mappings *f* as states of the *(M, R)* system, one can associate with each *(M, R)* system an automaton [151]. A set of mappings β: *H (A, B)* → *H (B, H (A, B))* ensures the "self-reproduction" of the *(M, R)* system since it associates with a mapping *f* the corresponding *<I>.r* mapping that replicates *f.* In a more recent version[237], *(M, R)* systems have an associated dynamical realization so that their state space is a topological structure rather than a discrete set. An example of the simplest *(M*, *R)* system is given in Fig. 41, which also contains a proposed realization of such an *(M*, *R)* system in molecular terms. *(M, R)* systems were shown to form a category [151] which is a subcateogry of the category of all automata [153, 154]. The basic algebraic properties of this category of (*M, R)* systems were considered in relation to their possible biological realizations [153, 154], and a generalized form of *(M, R)* systems with variable algebraic and topological structure was proposed [238]. Such extensions of *(M, R)-systems* theory will be discussed in Sec. 11 together with other organismic structures and molecular set theory.

An interesting possibility is the computer simulation of dynamical realizations of specific (*M, R)* systems with structures selected to simulate a particular group of biological organisms. The approach would be similar to the computer simulation of Kauffman nets and one may expect to obtain from such a simulation biologically relevant results.

11. NATURAL TRANSFORMATIONS OF ORGANISMIC STRUCTURES

In a recent report, relational theories of organismic sets [239], metabolic-replication systems [237] and molecular "sets" [240] were shown to have a similar foundation, and thus can be studied within a unified theory which employs a categorical framework [241]. This is possible because all relational theories have a molecular basis. Complex structures such as genomes, living cells, organs and organisms were mathematically represented in terms of molecular properties and molecular entities such as "molecular sets". The definition of organismic sets, for example, requires that certain quantities are determined from experiments: these quantities are specified by a special set of values of general observables which is derived from physico-chemical data [241]. Such observables are directly represented by *natural transformations* in category theory, and are also encountered in theories of metabolic-replication [(M, R)] systems, as well as molecular set theory (Ref. [240]).

Since the problem of uniquely decomposing such organismic structures into simpler, functional (or active) components appears to be an unsolvable one in the general case, we have adopted a complementary procedure which begins by building up specific structures corresponding to specific biochemical, genetic or organismic systems. We have then examined their properties in

terms of their mathematical representations such as the generating formalisms [241]. It is interesting that energetic considerations ultimately lead to molecular models and natural transformations [241, 242] or natural equivalences in particular cases [243]. A detailed investigation of the natural transformations of organismic structures is therefore necessary in order to understand certain basic themes of relational theories in biology, such as observability of molecular processes, realizability conditions for relational models/theorems, unicity of representation, dynamics /kinetics of biomolecular reactions and molecular evolution.

The simplest METABOLIC-REPAIR (M, R)-System with REVERSE TRANSCRIPTION.

ENZYMES ----------->> ***DNA / GENOME***

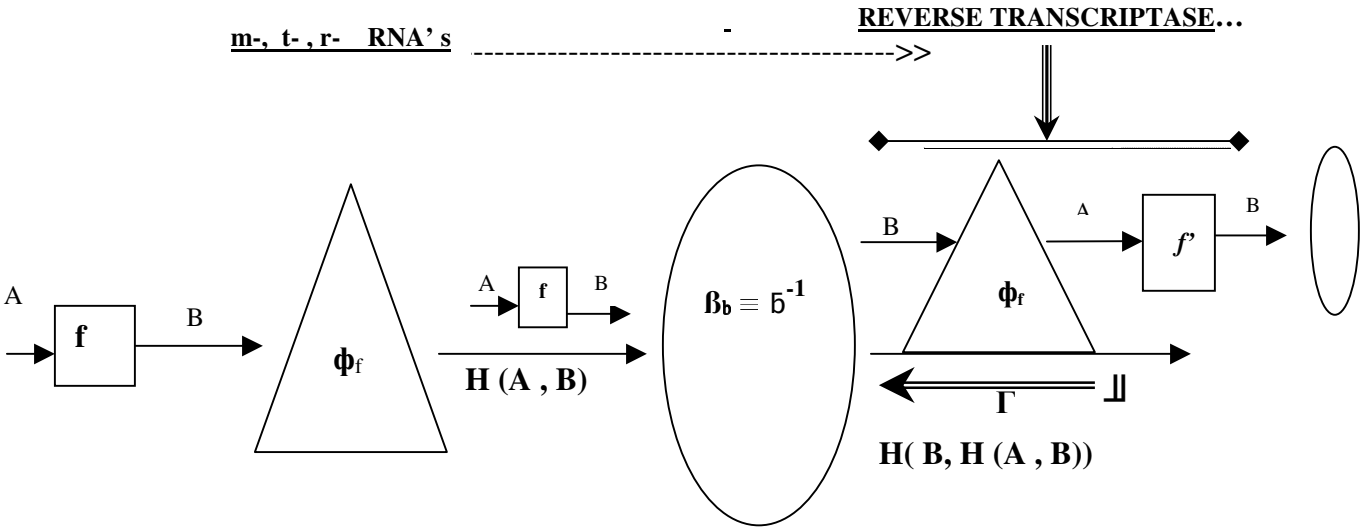


Fig 41. Diagram representation of the simplest (M, R) system (according to Ref. [238]) (Updated 06/11/2004).

When attempting to compare mathematical results of relational theories with recent advances and experimental observations in molecular biology one invariably finds that a theoretical-molecular model approach, such as the procedure adopted by us. is really needed in order to be able to realize the experimental implications of relational results.

Most theoretical studies of molecular models are, however, taking a very different course and are merely the result of formalizing a fixed set of experimental observations, without pursuing its wider implications for a synthesis of theoretical knowledge in biology. On the other hand, relational theories are still developing and may reach a stage when their predictions will make a strong impact on the molecular biologist's view, which so far has been "to divide and conquer". Our abstract molecular models of organismic and genetic structures in terms of natural transformations could help bridge the gap between relational and molecular biology.

A. *Molecular Automata and Molecular Models in Categories*

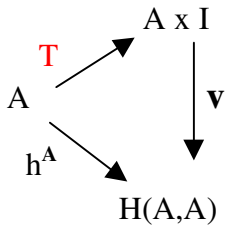
We showed in an early report [242] that energetic considerations of nerve excitation and conduction lead to a physical, molecular model of these processes, and allow one to analyze in physical terms the behaviors of neurons and neuronal networks. From quite a different standpoint we later considered n-state genetic networks [226] using n-valued logic in categories, and derived general theorems concerning the decomposition and dynamic properties of such networks in a general fashion. Some of these properties were found to be similar to those of neuronal networks, but the analogy was far from being complete. In these two partially analogous biological networks we found that a full, general treatment required the use of the mathematical theory of categories and functors [226]. Recently, it became possible to derive a synthesis of the three different approaches to relational biology-the organismic set theory [239], the theory of *(M, R)* systems [237] and the molecular set theory [240]. In our synthesis [241], natural transformations of organismic structures play a central role. The unifying concept of natural transformation is provided by the standard theory of categories and functors, initially developed as a general theory of natural equivalences [244] by Eilenberg and MacLane in 1945. A simple introduction of such a synthesis is based on set-theoretical models of chemical transformations [240]. Consider the simple case of *unimolecular chemical transformations* [240].

$$\boldsymbol{T:}\ A \times \boldsymbol{I} \rightarrow B \times \boldsymbol{I}, \qquad \textbf{(50)}$$

where A is the original sample set of molecules, $I = [0, t]$ is a finite segment of the real time axis and $A \times I$ denotes the indexing of each *A-type* molecule by the instant of time at which each molecule $a \in A$ is actually transforming into a *B-type* molecule [see also Eq. (3) in Ref. [240]). $B \times I$ denotes the set of the newly formed *B-type* molecules which are indexed by their corresponding instants of birth. As a flexible means of formalization, any chemical component-molecular set A of a biochemical subsystem in a living organism may be regarded as a variable quantity, or as a molecular set variable (msv), which spans certain allowable molecular sets (that is, those which are actually observed or realized if a molecular model was considered). The functional dependence of these msvs on time (in the mathematical sense) was then regarded as a kind of <<"relation" from the time axis to the class of molecular subsets as "range">>. This was termed the "wide-sense" kinetics of molecular set theory [240]. The transitions from one possible value (or state) of a msv to the next allowable value*(s)* will occur with some definite probabilities-in a statistical sense. While the concentration, or cardinality, of a molecular set component is constant, the set itself can change continuously its composition; this is the biologically significant factor in the operation or functioning of an organism.

At this point, one can easily categorize the transformations of a molecular set A by simply using the endomorphisms $f: A \rightarrow A$ and by considering the new set of all possible transformations of A, which will be denoted by $H(A, A)$. The molecular sets and their transformations can now be organized into a *category M* and certain functors h^x between such categories of molecular sets can now be defined. (For definitions of the concepts of category and functor please see Ref. [245].) Specifically, the functor $h^A: M \rightarrow$ Set, with Set being the category of sets, is defined by
$h^A(X) \equiv H(A, X)$ for any X in M as object, and $h^A(t) = m: H(A, A) \rightarrow H(A, B)$ for any morphism $t: A \rightarrow B$ in M, and B a molecular set of reaction products of type "*B*".

The flexible notion of molecular set variable (msv) is exactly represented by the morphisms *v* **in** the diagram



(51)

where morphisms *v* are induced by the inclusion mappings $A \xrightarrow{T} A \times I$ and the commutativity conditions $h^A = v \circ T$. The naturality of this diagram simply means that such conditions hold for any functor h^A defined as above.

The unimolecular chemical reaction is thus represented by the natural transformations $h^A \xrightarrow{\eta} h^B$, as one can readily check in the commutative diagram

h: *M-* ***---[M,*** **Set]**

$$\begin{array}{ccc} h^A(A) = H(A,A) & \xrightarrow{\eta_A} & h^B(A) = H(B,A) \\ \downarrow h^A(t) & & \downarrow h^B(t) \\ h^A(B) = H(A,B) & \xrightarrow{\eta_B} & h^B(B) = H(B,B) \end{array} \qquad (52)$$

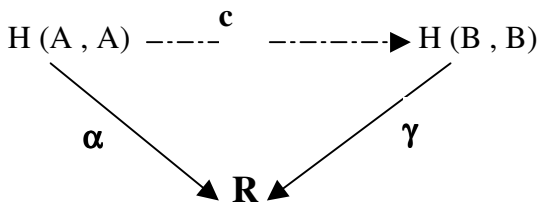
if the states of the molecular sets $A_u = a_1, \ldots, a_n$ and $B_u = b_1, \ldots b_n$ are represented by certain endomorphisms in H(A,A) and H(B,B), respectively. In the case of *multi- molecular reactions,* the canonical functor of category theory

$$h : \underline{M} \dashrightarrow \underline{[M, Set]} \qquad (53)$$

assigns to each molecular set $\underline{A}$ the functor $h^{A,}$ and to each chemical transformation

t: $A \longrightarrow B$, the natural transformation $h^A \xrightarrow{\eta} h^B$. As an example, let us consider the "replication maps" of *(M, R)-system* theory. These "maps" were shown to be representable by natural transformations (details are given in Ref. [238]). The machinery associated with metabolic and genetic activities of the "simplest", or primordial organism, could be simply visualized as in Fig. 41. This is, of course, Robert Rosen's simplest *(M, R)* system. Molecular candidates for the components of this metabolic-replication model are indicated at the top of the diagrams in Fig. 41. The corresponding molecular model is well established in molecular biology on an experimental basis. One quantity often omitted from this model is energy. Energy could be

readily introduced into the model as an input/output element for each component of the (M, R) system. A more physical approach would, however, introduce energy by defining an observation process on a molecular set and then convert it into the molecular set formalism by defining the appropriate natural transformations. A few details of this formalization are given below and full details will be presented elsewhere. An observation process on a molecular set will involve the preparation of a certain msv, *A,* (with preparation in the quantum-mechanical sense) into a selected state, or field of states, A^*_u: .The process is described by a morphism α: *H (A, A)* → *R.* For the chemical product "*B* " of a reaction, γ : *H (B, B)* → *R* is an observable of the msv *B,* which is measured in some specially prepared state (or field of states) B^*_u. The preparation itself can be subject to an uncertainty 8 in the set of real numbers *R.* α and γ are connected together in the commutative diagram



with *c* being "uniquely" defined as a morphism *c:* A^*_u → B^*_u:, within the uncertainty range δ. Among such observables of an msv, energy is an essential one since it places limits on the possible reactions and products, although it changes such restrictions to probabilistic statements. In our formalism, therefore, energy is represented as a morphism, and has quite general properties. In specific cases it will be necessary to define more precisely the energy and the statistical conditions related to the uncertainty range δ. Our definition of energy as a morphism is natural both formally and conceptually. Formally, the diagram (54) is natural both in a and 'Y, as well as certain "reactions" *c.* Conceptually, energy is one of the general observables of any organismic structure (for details please see pp. 433-434 in Ref. [241] and literature cited therein). As shown previously, the general observables of any organismic structure are natural in the categorical sense [244]. Natural transformations thus establish a formal and energetic link between organismic structures, (M, R) systems and molecular sets. By employing the canonical functors *f* and the natural transformations 11 as a translation device between (M, R) systems, molecular and organismic sets, once a result is obtained in one of these theories it can be readily translated into the others without losing generality. The translation in terms of molecular set theory is particularly important for physicochemical representations of abstract molecular models, such as the (M, R) systems, because molecular-set models have relatively direct interpretations in experimental / physicochemical terms. In practice, one is still left with the question of deciding which abstract model is actually realized in molecular biology, that is, defined by experimental data.

B. *Natural Transformations in Protein Biosynthesis and Embryogenesis*

As a particular example of protein biosynthesis let us consider the synthesis of approximately 50 different ribosomal proteins (or r-proteins) in the Gram-negative bacterium *E. coli.* The r-protein genes are arranged in many different operons, apparently placed at separate locations on the *E. coli* chromosome [246]. The r-protein biosynthesis is very well coordinated in *E. coli* and the

question of how this is achieved is an important one. From a theoretical standpoint, this system is a good example of a genetic network. The experiments indicate that the overall regulation of r-protein biosynthesis responds to changes in growth conditions and these may be primarily mediated by changes in the rate of transcription of the r-protein genes. By comparing, however, the transcription rates of r-protein mRNA in haploid and merodiploid strains, it was found that the expression of r-protein genes in the str-spc region is controlled by a post -transcriptional mechanism [246]. Such a mechanism could involve the inactivation and degradation of the excess of r-protein mRNA. Nevertheless, it is equally possible that the translation of intrinsically active *r-* protein mRNA. is blocked by other means; the "overproduced" mRNA could be more labile and this may impair its participation in r-protein biosynthesis. This example leads to an abstract molecular model similar to the one shown in Fig. 41. Thus, if the set of *r-* proteins is denoted by *H (A, B)* then the set of r-protein mRNAs will be represented by some subset of *H(B, H(A, B»;* the genome region which is transcribed into r-protein mRNAs will be then represented by a subset of *H(H(A, B), H(B, H(A, B)),* as in the standard *(M, R)-system* theory. As shown by Rosen [150], the "metabolic" components, such as the r-proteins, can be reorganized into a finite category *M.* Let any two sets of M be X and Y, and let t: X → Y be a mapping of *M.* If Set is the category of all sets and mappings of sets, then one can define a special functor h^x*: M* → Set as

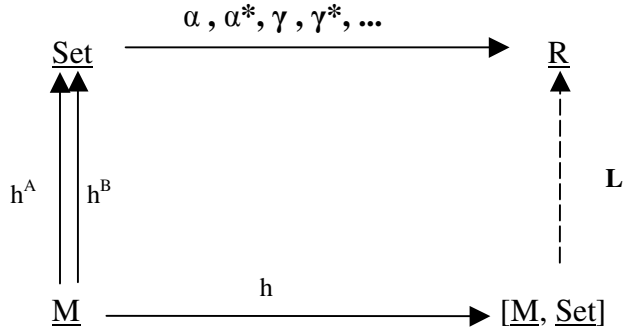
$$\begin{cases} h^X(Y) = H(X, Y) & \text{for any set } Y \text{ in } M, \\ h^X(t) = m\colon H(X, X) \to H(X, Y) & \text{for any } t\colon X \to Y, \\ h^X(g)(t) = gt\colon H(X, X) \to H(X, Y') & \text{for any } g\colon Y \to Y' \text{ in } M, \end{cases} \tag{55}$$

where X is a certain fixed object in M. The functor h^x carries Y into H (X, Y) without acting on the elements of Y. A family of functors of the type h^x , which is obtained by varying X in , will produce all sets of the form H (X, Y). The set of r-proteins H (A, B) can thus be generated by consiteering h^A (B) for A and B sets in M. IN order to construct the sets H (B, H (A, B)) which represent the r-protein mRNAs, one can use the canonical functor h: M→ [M, Set]. The functor h is defined by the assignments

$$S \rightsquigarrow h^X \qquad \text{and} \qquad t \rightsquigarrow h^X \to h^Y, \tag{56}$$

where t: X → Y and [M, Set] is a category of functors from *M* to Set. An *embedding* I: *M* → Set, carries any *X* of *M* into the same set *X* of Set, and any morphism *t: x* → *Y of M,* into the corresponding mapping of Set. The *natural transformations* Φ: I→ h^x with *X* varying in *M* now define the "genetic" maps Φ_f, that is, the elements of *H (X, H(X,* Y)). In the particular case of certain natural transformations Φ_r*:* I→h^A*,* we obtain a representation of the *r-protein* mRNAs; kinetic calculations derived from such a model and the "wide-sense" kinetics of molecular set theory are in broad agreement with the experimental observations of r-protein biosynthesis in *E. coli.* A categorical representation of r-protein biosynthesis thus consists of a functor hA-generating models of the *r-pro*teins-and certain natural transformations Φ_r: I→h^A*,* -generating the models of r-protein mRNAs. The generalization of this formalism to any protein *biosynthetic* process is im*mediate* by considering *the* appropriate h^P functors and Φ_p*:* I→h^P natural transformations. During protein *biosynthesis the* composition of *the* biological system will vary as a result of certain *multi* molecular reactions taking place; such processes induce certain natural

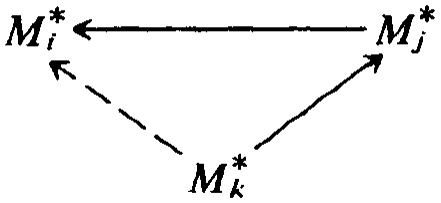
transformations *v:* α –—> α* and ω : γ –—>γ*, *with* α, α* : Set –—> *R* and γ , γ*': *Set* –—> *R* being certain special *functors.* From *the* definitions of natural transformations and multi-molecular *reactions* [relations (52) and (53), respectively] one obtains the commutative diagram



with ***L*** playing the role of a ***generalized observable***. In this diagram, the *canonical* functor h assigns to each *molecular set A* the functor h^A and to each *chemical transformation* t: A→B, the natural transformation η : h^A → h^B. The advantages of employing natural transformations over the use of the standard maps *t: A* → *B* of molecular set theory are:

(1) the natural transformations η are not restricted to mappings; and
(2) *A* and *B* can be truly varying structures rather than simple discrete sets.

This generalization of molecular set theory, (M, R)-systems theory and organismic set theory is important for kinetic modelling of protein biosynthesis since the states of molecular structures (previously discrete sets) can be defined in quantum-mechanical terms, for example. Calculation of transition probabilities between such states may be thus possible starting from physical first principles, in favorable cases; such calculations would employ standard quantum-mechanical approaches, such as the Dirac or the Heisenberg representation. Even without making use of such detailed formalisms it is possible to derive some of the basic properties of protein biosynthetic processes. For example, the msv associated with the final form of a synthesized polypeptide, can be shown to be representable as the direct limit of the intermediate forms in its synthesis (for a definition of direct limit and an example see p. 482 in Ref. [224]). On the other hand, a cell at the end of its synthesis stage will be represented by the projective limit of certain metabolic-replication systems associated with selected intermediate stages. This projective limit is constructed as a Cartesian product of sets of states / inputs / outputs and transition /output functions of the corresponding *(M, R)* automata or sequential machines (for details of the construction and the relevant definitions see Ref. [153]). The cell dynamics, including protein biosynthetic processes, are thus subject to the following natural restriction:



(58)

is commutative for any *i ,j, k* belonging to an ordered set *I,* and such that $i \leq j \leq k$. The ordered set in this case corresponds to the set of cellular events, and the M_j^* components are some selected *(M, R)* systems which represent certain cell stages at which different mRNAs are being synthesized. In this model, a complete cell, or organism, can be built from certain intermediate stages as a Cartesian product of the sets defining the selected stages. The problem is, of course, one of defining experimentally such stages. In the case of developing embryos, tests for nuclear equivalence showed that somatic nuclei from the blastula stage to early stages of organogenesis become progressively restricted in their ability to promote "normal" development when transplanted into enucleated eggs [247]. Up to a certain stage, however, the nuclei were equivalent, in the sense of being capable of inducing normal development. From a dynamic viewpoint such nuclei are adjoint, that previous section and in Ref. [169]. The special case of *centered L-algebras,* that is, those algebras which have *(n* -2) elements called centers, encompasses the previous model of Boolean algebras in medical diagnosis. An efficient diagnostic procedure would impose a number of restrictions on the structure of the subcategory D_n of $\mathrm{Luk_n}$ which represents the diagnostic decisions. For example, D_n should perhaps be loop-free in the sense that a chain of morphisms with the same orientation must not have the same beginning and end L-algebra. Such loops were previously called cycles [226]. This restriction on D_n corresponds to unalterable decisions, that is, decisions which cannot be reversed. From a practical viewpoint, this restriction may appear too severe; parallel reasoning and intuitive jumps are, however, permitted, and in this respect, too, our present model is an extension of the previous Boolean approach.

Categorical structures such as $\mathrm{Luk_n}$ could be hardwired on a digital computer as an aggregate of interconnected n-state black boxes, or else coded in numerical form using appropriate software. Such developments will be considered elsewhere.

We conclude that an algebraic model of medical diagnosis is now possible by taking into account the rather subtle aspects of diagnosis, such as parallel reasoning and contingent thinking. Our model is effectively a subcategory D_n of $\mathrm{Luk_n}$, the category of all L-algebras. Further developments will be needed to render some of our algebraic structures computable on a digital computer. Our approach thus opens the possibility of detailed analyses of medical diagnosis using n-valued logic and L-algebras.

CONCLUDING REMARKS

Computer simulations in biology and medicine are of increasing variety, subtlety and importance; the number of computer simulations in these fields is increasing almost exponentially. Selected examples of such simulations, ranging from branching studies of arteries to biochemical oscillators, neural networks, genetic nets and molecular automata were discussed with a view to the underlying, or unifying trends.

Automata theory and other algebraic models of biological systems provide sophisticated means for simulating the behavior of real biological systems. Dynamical systems and analogs of biological organisms are now considered which have complex state-space structures and exhibit novel behavior when compared with the simpler networks that have become traditional in theoretical biology.

A unifying view of theoretical models in biology was discussed and new applications of natural transformations were presented. Conjectures were made indicating new possible developments of computer applications.

Limitations of present digital computers for mimicing biological dynamics were considered in relation to "chaos" and n-ary networks based on n-valued logic.

The need for n-valued logical models in medical diagnosis was discussed and the role of previous Boolean models in mathematical medicine was briefly reviewed.

The underlying theme of all examples considered is a search for more general algorithms, better suited for modelling complex biological systems, and in which computer simulation and automata-theoretic approaches playa key role.

REFERENCES

1.Arbib, M. 1966. Categories of (M,R)-Systems. **Bull. Math. Biophys**., **28**: 511-517

2. Ashby, W. R. 1960. *Design for a brain,* 2nd ed., New York: J. Wiley & Sons, Inc.

3. Ashby, W. R 1956. .*An Introduction to Oybernetics,* New York: J. Wiley & Sons, Inc.

4. Baianu, I.C. and Marinescu, M. 1968. Organismic Supercategories:I. Proposals for a General Unitary Theory of Systems. **Bull. Math. Biophys**., **30**: 625-635.

5. Baianu, I. 1970. Organismic Supercategories: II. On Multistable Systems. **Bull. Math. Biophys**., **32**: 539-561.

6. Baianu, I. 1971. Organismic Supercategories and Qualitative Dynamics of Systems. **Bull. Math. Biophys**., **33**: 339-354.

7. Baianu, I. 1971. Categories, Functors and Automata Theory. ***, Proceed. 4th Intl. Congress LMPS***, August-Sept. 1971.

8. Baianu, I. and Scripcariu, D. 1973. On Adjoint Dynamical Systems. Bull. Math. Biology., **35**: 475-486.

9. Baianu, I. 1973. Some Algebraic Properties of (M,R)-Systems in Categories. Bull. Math. Biophys, **35**: 213-218.

10. Baianu, I. and Marinescu, M. 1974. A Functorial Construction of (M,R)-Systems. **Rev. Roum. Math. Pures et Appl**., **19**: 389-392.

11. Baianu, I.C. 1977. A Logical Model of Genetic Activities in Lukasiewicz Algebras: The Non-Linear Theory., Bull. Math. Biology, **39**:249-258.

12. Baianu, I.C. 1980. Natural Transformations of Organismic Structures. **Bull. Math. Biology**, **42**: 431-446.

13. Baianu, I.C. 1980. Structural Order and Partial Disorder in Biological Systems. Bull. Math. Biology, **42**: 464-468

14. Baianu, I.C.1983. Natural Transformations Models in Molecular Biology. SIAM Natl. Meeting, Denver, CO, USA.

15. Baianu, I.C. 1984. A Molecular-Set-Variable Model of Structural and Regulatory Activities in Metabolic and Genetic Systems., *Fed. Proc. Amer. Soc. Experim. Biol*. **43**: 917.

16. Baianu, I.C. 1987. Computer Models and Automata Theory in Biology and Medicine. In: "*Mathematical Models in Medicine*.",vol.7., M. Witten, Ed., Pergamon Press: New York, pp.1513-1577.

17. Bourbaki, N. 1958. *Elements de Mathematique,* Paris: Hermann & Cle, Editeurs.

18. Carnap. R. 1938. "'The Logical Syntax of Language" New York: Harcourt, Brace and Co.

19. Cazanescu, D. 1967. On the Category of Abstract Sequential Machines. **Ann. Univ. Buch**., Maths & Mech. series, 16 (1):31-37.

20. Comorozan, S. and Baianu, I.C. 1969. Abstract Representations of Biological Systems in Supercategories. Bull. Math. Biophys., 31: 59-71.

21. Ehresmann, Ch. 1966. "Trends Toward Unity in Mathematics." *Cahiers de Topologie Et Geometrie Differentielle,* 8,1-7.

22. Eilenberg, S. and S. MacLane. 1945. "General Theory of Natural Equivalences."*Trans. .Am. Math. Soo.,* 58, 231-294.

23. Eilenberg, S. and J. Wright. 1967. "Automata. in General Algebras." *Seventy-Second Meeting American Math. Soc.,* 1-17.

24. Georgescu, G. and D. Popescu. 1968. "On Algebraic Categories." *Rev. Roum. Math. Pures et Appl.,*

13, 337-342.

25. Georgescu, G. and C. Vraciu 1970. "On the Characterization of Lukasiewicz Algebras." J Algebra, 16 (4), 486-495.

26. Hilbert, D. and W. Ackerman. 1927. Grunduge.der Theoretischen Logik, Berlin: Springer.

27. Lawvere, F. W. 1963. "Functional Semantics of Algebraic Theories." *Proc. Natl.Acad. Sci.,* **50**, 869-872.

28. Lawvere, F. W. 1966. "The Category of Categories as a Foundation for Mathematics." In the *Proc. Conf. Categorical .Algebra-La Jolla* 1965, Eilenberg, S., *et al.* eds., Berlin, Heidelberg and New York: Springer-Verlag, pp. 1-20.

29. Lawvere, F. W. 1969. "Closed Cartesian Categories." (Lecture held as a guest of the Romanian Academy of Sciences).

30. Lofgren; L. 1968. "An Axiomatic Explanation of Complete Self-Reproduction." *Bull.Math. Biophysics,* 30, 317-348

.

31. M. Conrad and O. Rössler. 1982. Example of a system which is computation universal but not effectively programmable. *Bull. Math. Biol.* 44: 443-448.

32. McCulloch, W and W. Pitts. 1943. "A logical Calculus of Ideas Immanent in Nervous Activity" *Ibid*., 5, 115-133.

33. Mitchell, B. 1965. *The Theory of Categories.* New York and London: Academic Press.

34. O. Rössler. 1979. Chaos and strange at tractors in chemical kinetics, in Synthetics-Far from Equilibrium. Edited by A. Pacault and C. Vidal. Springer-Verlag, Heidelberg. 107-113.

35. Rashevsky, N. 1954. "Topology and Life: In Search of General Mathematical Principles in Biology and Sociology." *Bull. Math. Biophysics,* 16, 317-348.

36. Pitts, W. 1943. "The Linear Theory of Neuron Networks" *Bull. Math. Biophys*., 5, 23-31.

37. Rosen, R.1958.a."A relational Theory of Biological Systems" *Bull. Math. Biophys.,* 20, 245-260.

38. Rosen, R. 1958b. "The Representation of Biological Systems from the Standpoint of the Theory of Categories" *Bull. Math. Biophys.,* 20, 317-341.

39. Rosen, Robert. 1968. On Analogous Systems. Bull. Math. Biophys., **30**: 481-492.

40. Rosen, Robert. 1973. On the Dynamical realization of (M,R)-Systems. Bull. Math. Biology., **35**:1-10.

41. Russel, Bertrand and A.N. Whitehead, 1925. *Principia Mathematica,* Cambridge: Cambridge Univ. Press.

42. Warner, M. 1982. Representations of (M,R)-Systems by Categories of Automata., Bull. Math. Biol., **44**:661-668.