Nonlinear Analysis, Proc. of Second World Cong. of Nonlinear Analysts (WCNA96), 30(3):1815-1824 (1997)

THE GENE EXPRESSION MATRIX: TOWARDS THE EXTRACTION OF GENETIC NETWORK ARCHITECTURES

ROLAND SOMOGYI[†1], STEFANIE FUHRMAN[†], MANOR ASKENAZI[§], ANDY WUENSCHE[§]

[†]Laboratory of Neurophysiology, NINDS, Bldg.36, Rm.2C02, NIH, Bethesda , MD 20892, U.S.A.; and
[§]Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, U.S.A

Key words and phrases: Parallel distributed processing networks, genetic networks, gene expression, PCR, molecular biology, functional genomics, differentiation, boolean networks, states, trajectories, attractors, cluster analysis, reverse engineering

1. INTRODUCTION

Understanding of complex biological processes requires knowledge of the component molecular elements, as well as the principles that govern the interactions between them in forming higher ordered structures. We are founding our laboratory studies of CNS development and cell differentiation on the integrative concept of a genetic network, based on the tenets of genetic information flow. But first it is important to establish the intellectually challenging principles by which complex networks of functionally cross-linked elements lead to predictable, higher-ordered behaviors. To this end we are studying Boolean network models, which exhibit dynamic properties similar to those of living systems, such as self-organization and cycling. In this model, genes are conceptualized as binary (on/off) elements interacting within a freely cross-wired network. The on/off pattern, or state, of the entire network of genes updates itself as the genes interact, until the system reaches a final state, the attractor. This process of updating represents the pattern, or trajectory, of gene expression which results in the mature organism or differentiated cell type, representing analogies of the attractor.

Since trajectories and attractors are specific expressions of the architecture of a particular system, any experimental strategy must gain access to the states of the biological network. In that context, PCR (polymerase chain reaction) is being used to measure the expression of a large variety genes at different time points in a tissue or experimental cell system in order to gain access to data on trajectories. While many alternative trajectories may be obtained experimentally during cell and tissue differentiation or responses to perturbation, it is equally important to development the computational tools to infer genetic network architectures from such data sets. Here we discuss a heuristic approach to this problem using examples from Boolean networks as illustrations. Finally, analysis of experimental data is expected to provide testable hypotheses concerning further interconnections, some of which might not otherwise be predicted by strict molecular/mechanistic approaches. Especially within light of the massive genetic tool set generated by the genome projects, one may anticipate that a strategy of large scale gene expression mapping and genetic signaling network inference may become essential to the study of complex medical problems such as cancer or tissue regeneration.

[1] Internet contacts: for reprints and related literature, R.S., rolands@helix.nih.gov and S.F., sfuhrman@codon.nih.gov; for details on GeneTool, M.A., manor@santafe.edu; for information on DDLAB (a Boolean network exploration software package), A.W., wuensche@santafe.edu

2. PRINCIPLES: GENE FUNCTION IN DISTRIBUTED PARALLEL PROCESSING NETWORKS

Advances in molecular biological research and biological signal transduction have given us insight into the fundamental molecular processes of living organisms [1]. DNA has been identified as the primary information carrying molecule of all life forms. The central biomolecular tenets revolve around the direction of information flow in biomolecular systems (Fig. 1). For the information coded in the DNA to be expanded into biological function, individual genes must be activated and transcribed into a complementary RNA molecule (messenger RNA). While the RNA macromolecule can fold into conformations that show catalytic activity, or specific binding to other macromolecules, it appears to primarily serve as a template for the translation into a chain of amino acids, forming proteins. Proteins can explore a larger region of structural space because of the chemical diversity of the amino acid building blocks. Finally, many proteins catalyze biochemical reactions resulting in a large variety of organic molecules, many of which in turn carry information as intra- or intercellular signaling mediators. All of these macromolecular structures contain information that defines their specific affinities to one another. These affinities are exploited in biological signaling and control mechanisms, which span across any of the four categories of biomolecules shown in Fig. 1. Considering the number of members in each category, almost endless biomolecular combinations are possible. Exploring biomolecular function through particular examples is fascinating and has provided us with much basic knowledge on biological information processing. However, we may hesitate to expect to uncover all important molecular interaction because there simply are too many possibilities. While the schematic in Fig. 1 may appear simple, it implies the mind-boggling complexity of the hardware that underlies biological computation.

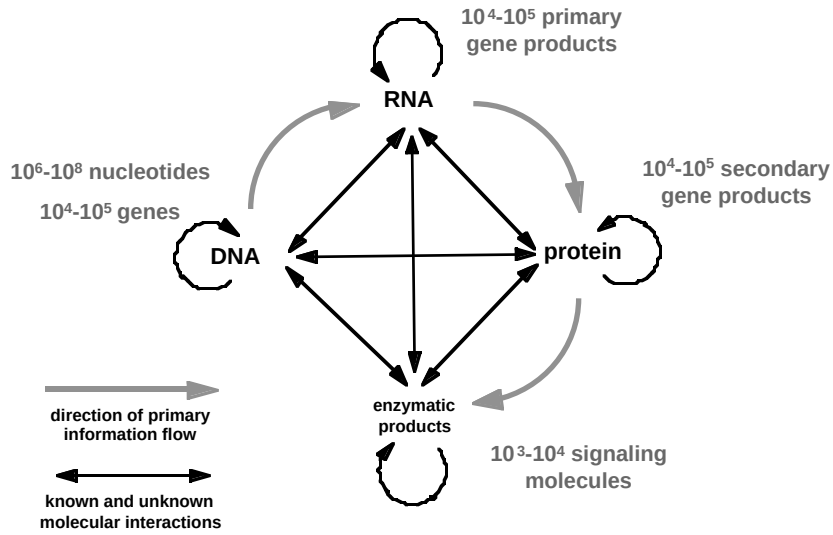

Fig. 1. Biomolecular information flow and feedback regulation[2]

But we would still like to understand how these parts can form higher-ordered functional wholes, first in principle, then in biological detail. This means that we have to begin integrating the insights we have gained so far. Let us start by examining the processes responsible for gene regulation, the origin of biomolecular information flow (Fig. 1). In a simple idealization (Fig. 2), an input activates a gene (implicitly leading to the formation of a protein), which forms the output. This output could activate a further gene, which is in turn linked to a another gene, and so forth, forming a signaling chain. While such signaling chains can already exhibit a significant degree of

[2] Perhaps some readers may perceive the abundant cross-wiring of Fig. 1 as exaggerated. But consider the following practical examples: Proteins aggregate with one another (multi-protein regulatory complexes) and with their enzymatic products (signaling molecule receptors, GTP-binding proteins) in regulating the activity of other proteins. Enzymatic products (e.g. steroids) form complexes with proteins (steroid receptors) that directly bind to DNA and activate genes. RNA structures directly play an important regulatory role in RNA stability, and can bind to complementary RNA sequences (antisense RNA) and lead to its elimination. Amino acids, lipids, small sugars and other enzymatic products act as central signal mediators in inter- and intracellular signaling. We can safely assume that many unforeseen molecular interactions will be discovered in the future.

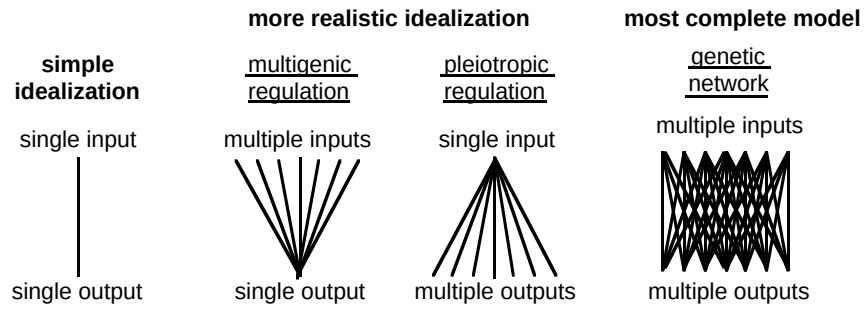

Fig. 2. Multigenic & pleiotropic regulation: the basis of genetic networks

complexity (as we already know from metabolic reaction series) they do not capture what we know about the combinatorial regulation of gene expression. Indeed, the activation of a gene is a multigenic process, involving the interactions of several different gene products with regulatory structures on the DNA (see [2] for a more in-depth discussion). Furthermore, the product of a single gene may act pleiotropically, i.e. may participate in several cellular functions because it can interact with a variety of molecular structures dependent on the context (Fig. 2). Therefore, we must accept that these molecular processes are tied into complex networks which form the real basis of the output that we associate with organizational structures such as cells and multicellular organisms.

How can such complex behavior be sensibly coordinated? One approach to this question is hardware oriented, meaning that by the careful study of all the molecular structures and all their possible interactions according to lock-and-key principles, we can take data on these reduced components and calculate up the organism in a high-powered computer (Fig. 3). Perhaps such thinking is motivated by the oversimplified idealization of one-to-one signaling chains (Fig. 2). However, the scope of data acquisition may lie beyond our experimental means (even taking into account the most generous estimates), and then may be non-computable even assuming that we have a precise representation of the input.

Premise:

"A Gene for Every Function and a Function for Every Gene"

- complete reduction of organism into genes
- determination of protein structures and activities
- mapping of molecular gene product interactions
- database assembly of molecular-mechanistic data?
- synthesis in sum-of-its-parts computer model?

Fig. 3. A reductionist-mechanistic approach to gene function

Alternatively, we may approach this problem from a computational perspective. To this end, the paradigm of deterministic genetic networks may prove most useful, particularly in development. Simply stated, the decompression of genetic sequence information in development (and beyond) may be understood in terms of patterns of coordinate gene expression governing proliferation and differentiation. Since sequence determines structure, and structure determines function (Fig. 1), the molecular interactions following gene expression do not account for additional information. Once the rules of interaction among genes are known, knowledge of all the intervening mechanistic steps is not absolutely necessary to determine the flow of genetic information.

A simple modeling language that accounts for the fundamental features of the global behavior of genetic networks can be found in Boolean networks (Fig. 4; studied by Kauffman, for overview see [3]). Boolean networks are based on the premise that biological molecular interactions exhibit a high degree of cooperativity, reflected in sigmoid interaction curves, which can be modeled by discrete on/off behavior in the limiting case. Essentially, the network elements, or genes, are either on or off, several of which act together through combinatorial or Boolean functions in the regulation of a particular element (Fig. 4). Such a simplified model will enable us to study the behavioral principles of distributed objects in biological systems. This is

Genetic Network	**<-> Boolean Network**
genotype / DNA	**<-> wiring and rules**
gene	**<-> element**
expression pattern	**<-> state**
development	**<-> trajectory**
mature cell	**<-> attractor**

Fig. 4. Genetic and Boolean network terminology

not a trivial question. One may easily be overwhelmed by the details of "mechanistic" molecular interactions and wonder how anything so complex as a living organism may exhibit stability, reproducibility, perhaps harmony!

Beyond network models, how can we gain experimental data on biological networks that would provide a foundation for detailed examination? In fact, the basic technology for the massively parallel analysis of genetic networks exists *here* and *now*. Analytical molecular biological techniques such as PCR (polymerase chain reaction) allow us to detect and precisely measure the concentration of any nucleic acid in solution [4]. The primary product of a gene is a nucleic acid, mRNA (messenger RNA). An mRNA isolated from a sample of living tissue (standard laboratory procedure) can be quantified using PCR, or perhaps in the future by alternative, equally sensitive nucleic acid hybridization-based approaches (such as DNA chips [5]). Efforts towards the automation of PCR analysis are being actively pursued [6]. Application of this technology provides us with a **Gene Expression Matrix** of a particular state or series of states of the organism's genetic network [7]. Perfecting experimental analytical techniques is only one aspect of genetic network exploration. Equally important will be the interpretation of the volumes of acquired data. How much of the genetic network architecture could possibly be extracted from a Gene Expression Matrix? Boolean networks allow us to examine this question.

3. STATES, TRAJECTORIES AND ATTRACTORS IN BOOLEAN NETWORKS

The elementary structure of Boolean networks is based on an analogy to biomolecular interactions, genes turning each other on and off through complex combinatorial functions in a richly cross-wired network. Assuming the model is valid, we should expect that computation within a Boolean network produces structures analogous to higher ordered distributed biological processes and objects, such as cells (characterized by a "final", stable state of the network) , the program of development, even transdifferentiation processes leading to e.g. cancer.

Following an example of a simple Boolean network, "Genet", will let us examine this question. Genet consists of 15 elements or "genes" labeled alphabetically. The links between the genes are shown in the wiring diagram (Fig. 5). According to these connections and the set of Boolean rules (Fig. 6), the state of the network at Time=t is transformed to the follow-up state at Time=t+1. The incremental calculation from state to state generates a *trajectory*. A set of 4 trajectories is shown in Fig. 7. Trajectory I begins with all elements off, except for "O", which is on. Therefore, according to the rule in Fig. 6, M must be turned on in the next iteration (time 2) because it responds to N or O. Each of the following states of the network is computed analogously (one can confirm this by

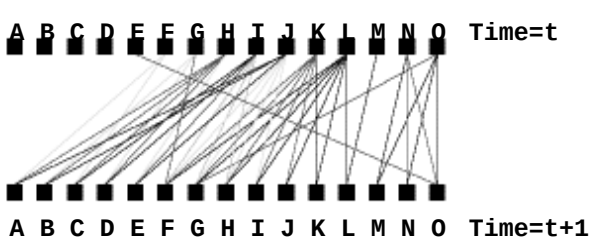

Fig. 5. Wiring diagram of a Boolean network: "Genet"

gene	Boolean rule
A	F and H and J
B	G and H and J
C	F and H and I
D	G and H and I
E	H and I and J
F	I and J and K and L and (not G)
G	I and J and K and L and (not O)
H	I and J and K and L
I	J and K and L
J	K and L
K	K or L
L	L or M
M	N or O
N	N and O
O	N and O and (not E)

Fig. 6. Boolean rules of Genet

		gene name														
trajectory	time	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
I	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
	5	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0
	6	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0
	7	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0
	8	1	1	1	1	1	0	1	1	1	1	1	1	0	0	0
	9	0	1	0	1	1	0	1	1	1	1	1	1	0	0	0
	10	0	1	0	1	1	0	1	1	1	1	1	1	0	0	0
II	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0
	2	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
III	1	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0
	2	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
IV	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	0
	2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig. 7. Four selected trajectories of Genet

calculating each step manually following the rules in Fig. 6). Interestingly, at time=10, the same state as at time=9 is computed; Genet has reached a "point *attractor*". Analogous behavior is also observed in trajectories, II, III and IV (Fig. 7). Although II, III and IV begin at different starting states, each trajectory falls into an attractor after 3 iterations. In fact, they share the same attractor, the all "off" state, which means that they are all part of the same "basin of attraction"((for detailed exploration of attractor basins see [8] and Boolean network structures see [9]).

The complete basin of attraction is depicted in Fig. 8 (graphics generated using the DDLAB software [10]). Each end point of a radius and each node represent a particular state, while each set of connected lines leading to the center, the attractor, represents a trajectory (trajectories II, III and IV are labeled specifically). In essence, the graph depicts a set of centripetal trajectories, covering 1024 of the total of 32768 states of Genet. A simple argument explains why each state in a Boolean network must lead to an attractor. Each network has a finite number of states, precisely 2^N, N being the number of elements in the network. Since each state leads exactly to one follow-up state, after a maximal number of 2^N steps (usually much less), a state must be reached that the network has occupied before. Once that occurs, the attractor has been reached and the network will oscillate within this cycle or point *ad infinitum*.

Genet illustrates how the state space that the network occupies is reduced as time progresses. Note this is exactly the behavior we see in living organisms. Of all the possible states a genetic network could produce, only a small subset is realized. For instance, in metazoan development the genetic network follows a determined program leading to e.g. several hundred cell types in higher vertebrates. This program can be likened to a trajectory, while the end-points represent attractors. While a myriad of gene

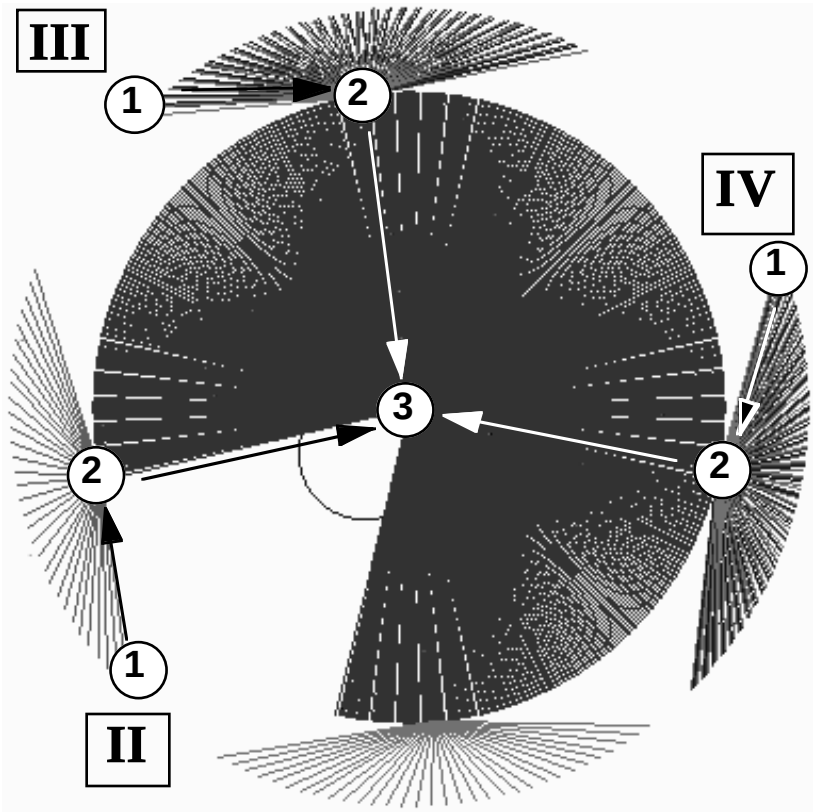

Fig. 8. Basin of the "all off" attractor capturing 1024 states of the network

activation states are possible, only very few are realized, i.e. the number of cell types actually observed is only a small fraction of the number that could in principle be created by a network of such size [11]. Of course, these analogies go beyond genetic networks and also apply to even more complex neural networks, where trajectories and attractors find analogies to pattern recognition, memories etc. [9]; after all, we often and appropriately speak of "mental states". Furthermore, nodes in trajectories into which a high number of states flow, or attractors with reasonably large basins of attraction, offer stability (discussed in [2]). Many changes in the value of an element, as could be introduced by noise or a slight perturbation, will likely generate a state which is part of a trajectory leading back to the attractor. Therefore, while one may have intuitively found it perplexing that the actions of so many elements can be coordinated into organized behavior, the intertwined nature of the network demonstrates that stability and finality are actually inherent features, and probably the great advantage, of such complex distributed structures as modeled by random Boolean networks.

4. FIRST-ORDER HEURISTIC NETWORK ANALYSIS: IDENTIFICATION OF PRIMARY FUNCTIONAL CONNECTIONS

We have discussed Boolean nets as a tool for understanding the computational principles of genetic networks, and how higher-ordered structures emerge and are maintained in richly interlinked systems. But how can we begin to understand the structure of any particular living genetic network? First of all, we must map the states and trajectories of a biological genetic network, which we are measuring today via the Gene Expression Matrix (discussed above). The next challenge is the extraction of the network architecture from the Gene Expression Matrix. This can be explored at several levels of analysis.

As a first step, one may look for similarities between trajectories of individual genes. We expect that genes which a) operate together (e.g. proteins that are part of a metabolic pathway or signaling network) or b) are members of the same gene sequence family (most genes are members of larger evolutionary families), will be regulated in a largely parallel fashion and therefore should exhibit overlapping trajectories. This type of relationship was anticipated in the construction of the Genet example; note the overlap in wiring between many of its elements (Figs. 6 and 7). Essentially, the elements of Genet fall into 4 or 5 major clusters (encircled by dotted lines) according to analysis of their wiring as shown in Fig. 9a (the tree was computed with FITCH[3] [12] from the 15 gene × 15 gene distance matrix of relative shared wiring). Can we gain access to this information without *a priori* knowledge of the wiring diagram? To this end, we compared a set of trajectories of Genet, covering a total of 73 states, using the Euclidean distance measure: each gene is a point in 73-dimensional parameter space; therefore the Euclidean distance between a pair of points (genes) is simply the square root of the sum of the squared distances in each dimension (time). The pair-wise Euclidean distances were entered into a 15 gene × 15 gene distance matrix, which served as the input for the FITCH[3] clustering algorithm,

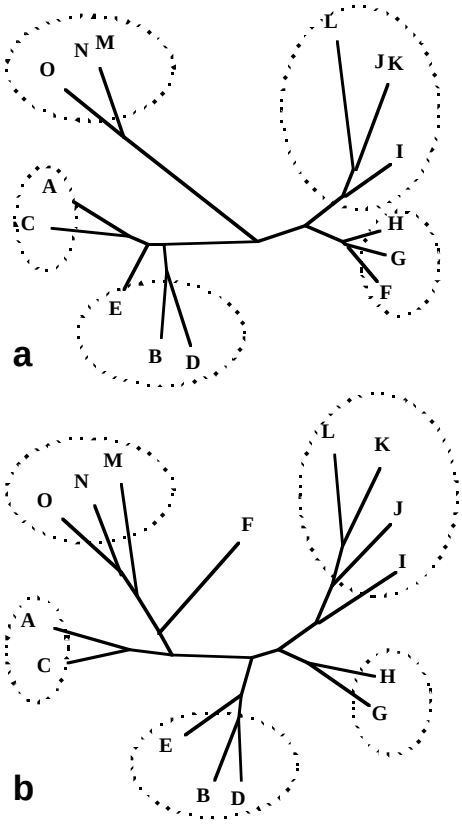

Fig. 9. Gene clusters determined from a) wiring and b) trajectories

[3] FITCH is a clustering algorithm from the PHYLIP package of Joe Felsenstein and can be obtained through the internet: http://evolution.genetics.washington.edu/phylip.html. We used the default parameters and a power of 0 for error weighting.

producing the tree shown in Fig. 9b. The clusters of the tree of trajectories bear close resemblance to those obtained from the direct analysis of wiring shown in Fig. 9a. The major outlier in this group is gene F, which is to be expected since F is inhibited by G, even though they otherwise share the same wiring (see rules, Fig. 6). Therefore whatever activates F and G together will cause an immediately following inhibition of F, leading to a bifurcation of F's and G's trajectories. However, alternative distance measures such as mutual information may be able to capture the relationship between F and G. We conclude that cluster analysis (Euclidean distance measure) of trajectories provides us with a good first approximation of which genes share major inputs.

5. HIGHER-LEVEL HEURISTIC NETWORK ANALYSIS: COMPLETE REVERSE ENGINEERING OF BOOLEAN NETWORKS

We have introduced a straightforward, first-order genetic network analysis which identifies genes having common inputs. More sophisticated techniques are required to go further and identify input source, multiple inputs and their combinatorial rules, culminating in the extraction of the complete network architecture. Again, Boolean networks offer a testing ground for such a strategy.

A Boolean network is determined by simply defined wiring and rules. Therefore the task of reverse engineering a network from a set of trajectories is straightforward: find a subset of wiring and Boolean rules that produces the required trajectory. But there may be many subsets, many networks that produce a small set of trajectories. Here it is important to introduce an optimization criterion to select a "minimal" network that will satisfy the input trajectories.

To this end, we are developing the GeneTool algorithm. GeneTool attempts to reverse-engineer the target genetic network based on a very small set of observed trajectories and some heuristic knowledge about the nature of the network. We also require criteria of minimality to reduce the number, and consequently increase the plausibility of the candidate solution(s). To date we have concentrated on minimizing the wiring per gene (i.e. reducing the set of genes responsible for the regulation of every gene in the network). Other minimization criteria are possible (e.g. logic gate minimization, as performed by computer chip designers, promises to be another exciting alternative [13]).

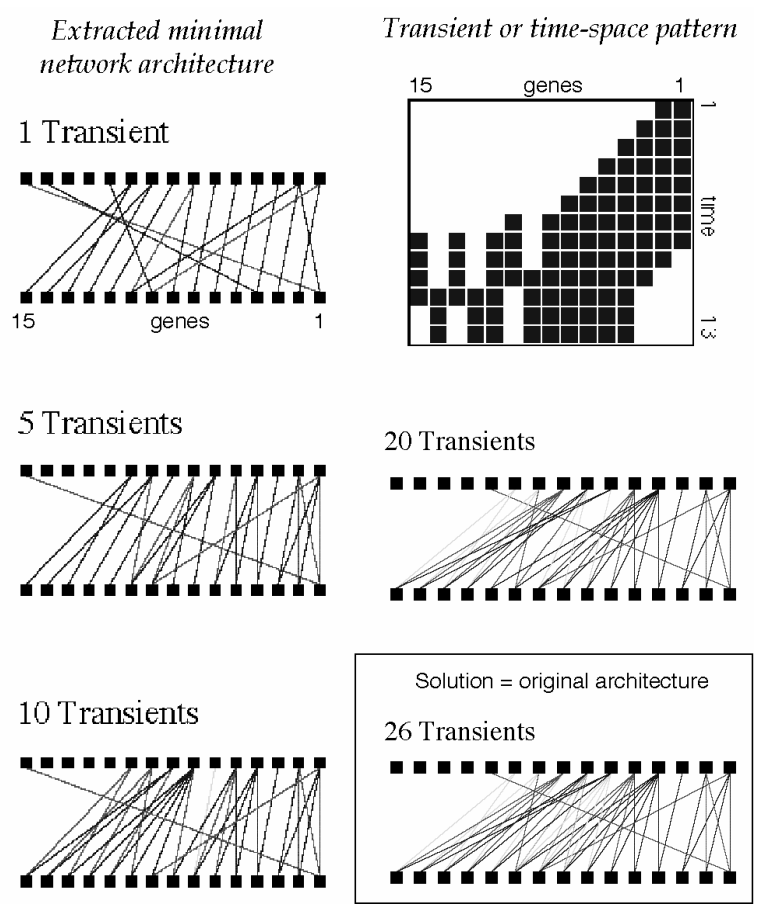

Fig. 10. Reverse engineering of Genet using the GeneTool algorithm[4]

[4] The figures shown here were generated using an earlier version of the GeneTool . The algorithm described in this paper produces even *smaller* networks.

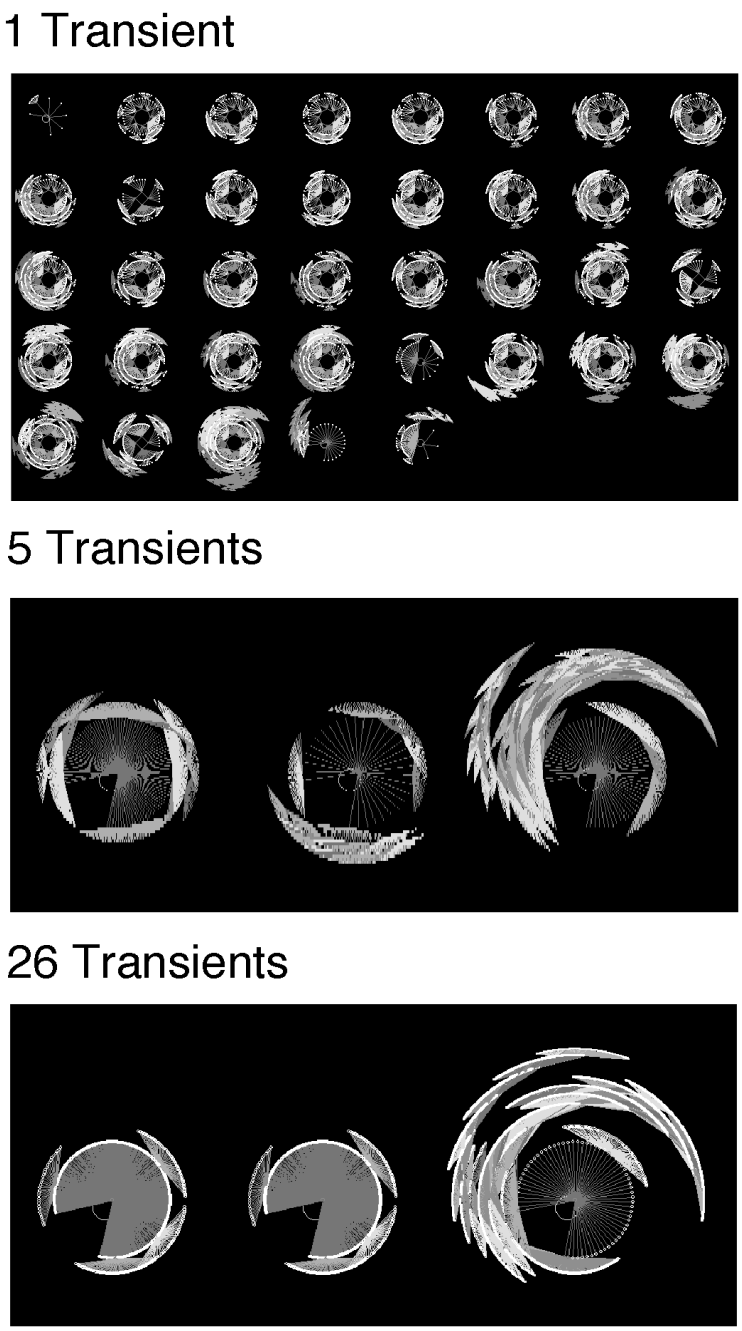

Fig. 11. Global dynamics of reverse engineered networks

We are currently investigating the following general approach: for every gene G, we find a minimal subset of genes S{G}, for which we can write a rule table taking the values of S{G} at time t and producing the observed state of G at time t+1. In practice, this means that for every candidate set S{G}, we fill in the rule table based on the observed state transitions until either we find an inconsistency, or we have managed to explain all the observed data. If, upon termination, we have incomplete rule tables, we complete them according to some biologically-motivated heuristic (one such heuristic is to maximize the number of canalizing inputs [14]).

We have tested the performance of GeneTool on our Genet model network, and found the results in some ways surprising. As expected, GeneTool was able to generate a minimal Boolean network for all the trajectories provided, and furthermore, the accuracy of the match with the original network increased with the number of state-transitions analyzed. However, while the wiring diagram became progressively more dense with increasing number of test trajectories (Fig. 10), the global dynamics of Genet were generally outlined following only 5 sets of trajectories as shown in (Fig. 11). In detail, following a single transient (shown in upper right of Fig. 10), a minimal network was extracted which accounted for the trajectory and produced a total of 37 different basins of attraction (top panel, Fig. 11). After a set of 5 transients, the wiring density increased marginally (Fig. 10), but the basins of attraction were reduced to three. In fact, the attractors corresponded to the exact solution of the network, and even their size and basic structure closely matched the original network's (Fig. 11). Examining additional trajectories led to a closer approximation of the exact network wiring and rules (not shown) until the perfect match was found after 26 sets of transients[5] (Fig. 10). But only a small subset of trajectories was necessary to capture the essential features of the global dynamics of the network (Fig. 11). As discussed above, Genet was designed with the redundancy in mind that we expect of biological

[5] Once we had established a convergence towards the original network's basin of attraction field, we began to feed the transients non-randomly, thus shortening the number of iterations required.

networks based on sequence and functional gene families. Reverse engineering of Genet using GeneTool may be reason for optimism about practical applications to biological networks, for which we can only expect partial data sets as provided by the Gene Expression Matrix. Even forgoing a "perfect match", reverse engineered network approximations may exhibit useful predictive powers.

6. CONCLUSION

The success of the molecular reductionistic approach in the life sciences is providing us with an ever increasing toolbox of genes and insights into individual molecular processes that underlie biological function. Scientific explanations of a particular phenomenon amount to algorithmic compression, trying to find a simple unifying rule to explain a variety of behaviors. But this should not result in the expectation of finding "point causes" for important biological processes; one cannot today ignore the network nature of biological function. But how can we seek simplifying explanations in the face of such complexity? Using the analogy of Boolean networks, we find that higher order processes can be explained in terms of trajectories and attractors generated by the interactions of the system's fundamental elements. We have introduced genetic networks as a paradigm in which today's technology will allow us to explore living networks. In principle, measurement and analysis of the Gene Expression Matrix should enable us to find the major causal links between genes (Fig. 12). Methods such as cluster analysis and reverse network engineering, as validated in principle on Boolean network models, are a beginning for a systematic program of network architecture extraction. In one way or another, the strategy outlined in Fig. 12 will likely play an important role in "Functional Genomics", the new frontier following the Human Genome Project.

Premise:

"Gene function is distributed across a parallel processing network."

1. identify organism's genes, genetic network elements
2. determine network states (gene expression patterns)
3. mapping of alternative trajectories and attractors
4. parallel trajectories suggest shared inputs
5. confined perturbation waves determine temporal links
6. computational reverse engineering of network

Fig. 12. A heuristic, integrative systems strategy for deciphering gene function

Of course, many important problems must still be solved. While the idealizations of Boolean networks help us provide a crisp modeling scheme with which we can study the principles of genetic networks and explore methods of data analysis, the biological reality is more diffuse. An important step will be to move away from the binary idealization in order to analyze experimental data which is measured on a continuous scale. Artificial neural network systems tailored to genetic network analysis could make important contributions here. Furthermore, the precise details of the genetic network may elude us for some time, since it is not possible to measure system states in single cells; we are restricted to measuring the aggregate Gene Expression Matrix averaged over several thousands or millions of cells. Nevertheless, important regulatory pathways should be identifiable even at such a course level of measurement. Furthermore, since we cannot hope to identify all important molecular interactions, a mechanism-

independent approach of Gene Expression Mapping followed by advanced network analysis should allow the identification of significant functional links otherwise not accessible.

Perhaps it is time to clearly formulate the general goal of the endeavors schematized in Fig. 12. While a complete understanding of an organism in molecular network terms may be precluded, the construction of approximate models based on analysis of large data sets could permit the formulation of profound predictions accessible by no other means. Model predictions are testable using today's technologies. Inadequate models can be improved by incorporating new data previously beyond their grasp. While this task may appear overwhelming at first glance, advances in automation of experimental data acquisition, analytical network software, and a concomitant increase in computing power will make this approach feasible, perhaps unavoidable.

REFERENCES

1. ALBERTS, B., BRAY, D., LEWIS, J., RAFF, M., ROBERTS, K. & WATSON, J.D. (1994) Molecular Biology of the Cell. Garland Publishing
2. SOMOGYI, R. & SNIEGOSKI, C.A. (1996) Modeling the complexity of genetic networks: understanding multigenic and pleiotropic regulation. *Complexity* 1(6):45-63
3. KAUFFMAN, S.A. (1993) The Origins of Order, Self-Organization and Selection in Evolution. Oxford University Press
4. SOMOGYI, R., WEN, X., MA,W. & BARKER, J.L. (1995) Developmental kinetics of GAD family mRNAs parallel neurogenesis in the rat spinal cord. *J. Neurosci.* 15:2575-2591
5. NOWAK, R. (1995) Entering the Postgenome Era. *Science* 270: 368-369
6. SOMOGYI, R. (1995) National Institutes of Neurological Disorders and Stroke: Opportunity for a Cooperative Research and Development Agreement (CRADA) for the development of a high performance gene expression mapping assay system. *Federal Register* 60:34544-34545
7. WEN, X., FUHRMAN, S., SMITH, S., ASKENAZI, M., BARKER, J.L. & SOMOGYI, R. (1996) From Genes to Dynamic States: Mapping Spinal Cord Development with the Gene Expression Matrix Reveals Functional Gene Clusters. *Soc. Neurosci. Abstr.*, p. 519
8. WUENSCHE, A. & LESSER, M.J. (1992) The Global Dynamics of Cellular Automata. SFI Studies in the Sciences of Complexity, volume 1. Addison Wesley
9. WUENSCHE, A. (1993) The Ghost in the Machine: Basins of Attraction in Random Boolean Networks. In Artificial Life III, ed. Langton, C.G., 465-501.
10. WUENSCHE, A. (1995) Discrete Dynamics Lab (DDLAB), software and documentation at MIT Press Artificial Life Online (alife.santafe.edu / alife / software / ddlab.html).
11. KAUFFMAN, S.A. (1974) *J. Theoretical Biol.* 44:167-190
12. FELSENSTEIN, J. (1993) PHYLIP (Phylogeny Inference Package), version 3.5c, distributed by the author, Department of Genetics, University of Washington, Seattle, WA
13. BRAYTON, R., HACHTEL, G., MCMULLEN, C. & SANGIOVANNI- VINCENTELLI, A. (1984) Logic Minimization Algorithms for VLSI Syn- thesis. Kluwer Academic Publishers
14. KAUFFMAN, S. A. (1971) Gene regulation networks: A theory for their global structure and behavior. *Current Topics in Dev. Biol.* 6:145