On Evolutionary Design, Embodiment, and Artificial Regulatory Networks

Wolfgang Banzhaf

Department of Computer Science, Memorial University of Newfoundland
St. John's, NL, A1B 3X5, CANADA
banzhaf@cs.mun.ca

Abstract. In this contribution we consider the idea that successful evolutionary design is best achieved in a networked system. We exemplify this thought by a discussion of artificial regulatory networks, a recently devised method to model natural genome-protein interactions. It is argued that emergent phenomena in nature require the existence of networks in order to become permanent.

1 Introduction

Michael Conrad [1] is often cited with the following: *„In conventional design the vast majority of interactions that could possibly contribute to the problem are deliberately excluded"*. As designers of system we often lean toward the easiest solution: Divide and conquer. I.e., we design a system using components proven to function as specified, with each of these components in turn being designed by the same process, but for a particular sub-task. Whereas there is nothing wrong with such a design methodology, the question is whether it will scale up. By scaling-up I mean whether it would be possible, using such a method, to design a system with, say, human-like complexity and sophistication. Essentially we are asking whether a complexity which rivals that of Life's creatures can be designed and constructed in this way. It might be conjectured, that this will not be possible [2].

Now that life has already entered the scene, we can put forward a different thesis: Life-like performance and complexity in the human (artificially designed) world will only be possible if we take inspiration from Biology. Alternatively, human-designed systems will unintentionally develop into life-like systems. The essence of this idea of bio-inspiration is emergence (of functionality) through (possibly unforeseen) interactions among components. Thus, instead of isolating the sub-parts of our systems in order to get "clean" functionality, we should rather count on the interactions for securing the functionality.

In order to stabilize emergent phenomena Nature uses networks. Networks are able to capture the interactions (links) of components (nodes), and through multiple connections from each component, become less prone to failures in components. This way networks allow the emergent phenomenon to embody itself (as the network). Examples from the natural world (including human activity) are

F. Iida et al. (Eds.): Embodied Artificial Intelligence, LNAI 3139, pp. 284–292, 2004.

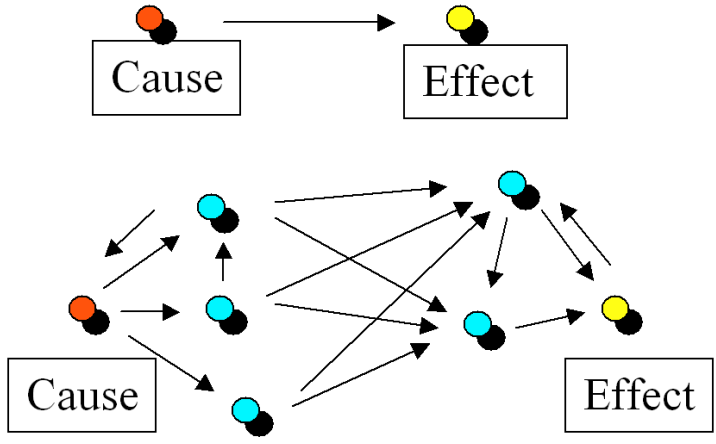

Fig. 1. Networks distribute the connection between cause and effect, and produce fault-tolerant mechanism.

- Elementary particles,
- Chemical reactions,
- Regulatory networks,
- Social interactions,

just to name a few. Elementary particles could be considered nodes in a network of elementary particle interactions, molecules could be considered nodes of a chemical reaction network, genes could be considered in the same way, interacting via their proteins and regulatory sites. Organisms could be considered nodes of a social network with communication links providing the edges of social interaction. What makes networks so fascinating and, at the same time so difficult to analyze, is their effect on simple cause-effect relations: They basically dissolve simple relations between causes and effects in favor of highly distributed networks of partial causes and partial effects. Figure 1 tries to sketch the situation: Assuming outgoing edges as causes and incoming edges as effects, one can see, that a simple relation of cause and effect could be substituted by a network. Nodes collect effects from incoming nodes and distribute there causes over outgoing edges. As a result, if analyzing form the point of view of the original nodes, it is difficult to understand how cause and effect nodes are actually connected. Natural evolution is not they only mechanism finding it useful to apply these systems. A single link (or more than one) can be broken without interrupting the cause-effect chain. The signals will simply flow via other edges. Thus networks provide a highly fault-tolerant environment for signal transfer.

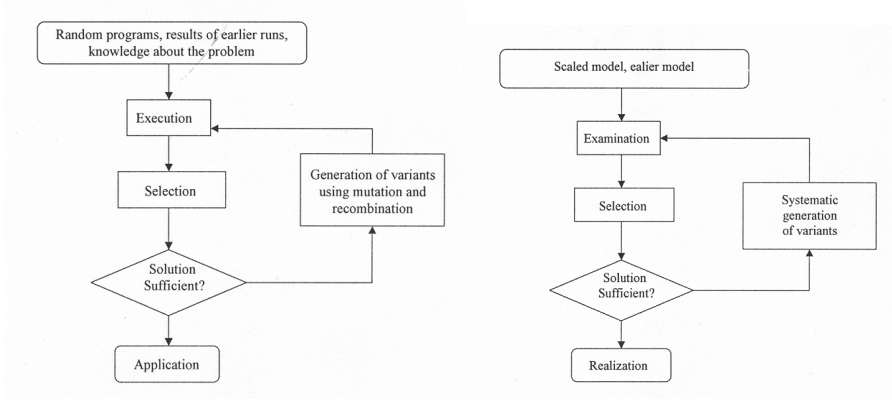

Fig. 2. Comparison of human and natural (evolutionary) design process

2 The Human Design Process Versus the Evolutionary Design Process

In this contribution, we are concerned with the design of systems, i.e. the intentional production of effects or function. Because networks are acting in a highly non-linear fashion and are difficult to analyze, evolutionary approaches to the design of those systems are considered. Here it might help to compare the human and the natural design process in more detail (see Fig. 2).

Traditional design consists of the application of complex principles and rules. It is usually a TOP-DOWN approach which begins with a high-level specification of the problem and moves down through a hierarchy of refinements until realization is reached. Only the best design is realized and closely examined for weaknesses. These weaknesses are then addressed in a separate step and weeded out.

Evolutionary design, on the other hand, consists of an often random combination of a large number of structural elements. If not random, the combination follows simple principles. It is a BOTTOM-UP approach which often passes through a more or less complex developmental process. In order to work properly, a multitude of designs must be examined by Nature. Examination takes the form of tests under "real" conditions.

The difference between the two design methods boils down mostly to cost considerations. If humans had the same cost structure as nature, we'd probably embark on an evolutionary design path towards new products and systems.

Natural design processes are said to be non-intentional, because no overarching plan to achieve certain functions can be identified. If something new is achieved, it is often by way of exaptation [3], a discovery process that works by exploiting side-effects to other functionally important (and selected for) features. In order for this process to be effective, functions should not be isolated from each other. In other words, a single element should have a potential for multiple functions and the partition of functions between components should be more "fluid" than we usually would

tolerate. There should also be no boundaries between subsystems, at least no tight boundaries, because communication between subsystems is essential in order to benefit from discovered side-effects. All of this naturally connects to what has been said about networks.

It is also interesting to note, that living organisms are usually generated through a developmental process, i.e. a process of building while working. Again, networks are able to accommodate such a requirement, by providing functions while new nodes and connections are added in.

3 Approaches to Construction

Putting together an entity with of the order of

- 1000 parts — is possible, but difficult
- 1,000,000 parts — is possible in principle, but very difficult
- 1,000,000,000 parts — is impossible with conventional methods.

Humans construct machines by producing parts, putting them together and turning them on. Each of the parts has its independent existence and can be manufactured in isolation from other parts. An overall plan will make sure that the correct order of construction is obeyed, leading to an ultimately functional device.

This is fundamentally different from how Nature constructs systems. No fixed order of events in the construction of a system can be obeyed, due to the stochastic and distributed nature of interactions. Also, parts cannot be produced in isolation; neither can they be produced from a master plan. Finally, living organisms cannot be "turned on" once enough components are assembled. Rather, even a rudimentary system must live from the very beginning in order to be able to continue to live.

Besides problems of controllability of spatial and temporal flows, the sheer number of elements needed to build a living organism is substantial. This is one of the most daunting problems Nature has faced when designing and constructing organisms. Other methods of construction than those we apply in machine construction are needed if a working entity should result from the construction process. These new methods involve growing an entity from a single plan. But instead of having this single plan be always accessed from multiple sites, the plan itself becomes part of what is being built, by integrating it into the parts and subsystems being constructed.

The information dilemma of Nature's evolutionary design is substantial, and can be summarized with the following questions [4]:

- How to instruct bodies with so few genes?
- How to program brains for so many situations?

Nature's answer to the first question posed was to invent a developmental process of construction. Nature's answer to the second, more detailed question was to use an adaptive process which extends development from the ontogenetic level.

In order to appreciate the challenge Nature has been facing, I'll give a few numbers:

Take a single-cell organism like *E.Coli*. It has
- 300 x 10^6 molecules (excluding water)
- 3250 different varieties (proteins, mRNA, tRNA, DNA, lipids,)
- 4.6 x 10^6 base pair genome = 6 Mbits of information
- 4,300 protein coding genes (88 % of genome)
- 11 % of genome contains regulatory information.

If we want to go further and have a look at a multi-cellular organism like *H.Sapiens*:
- 50 x 10^12 cells
- Each cell of about the complexity of an *E.Coli*
- 3 x 10^9 base pairs = 4 Gbit of information
- 40,000 – 100,000 genes

The human brain consists of
- 10^9 neurons
- 1,000 – 10,000 synaptic connections per neuron
- 300 – 1,000 vesicles per synapse

As mentioned, the response of Nature was to invent development, a process, by which
- the required complexity is grown
- environmental complexity is channeled into the developing phenotype, i.e. the genotype only directs the assembly
- exploitation of side effects (through evolution) is possible
- open-ended evolution is brought about through adding layers of complexity
- the generation of modular structure comes for free due to its recursive nature
- coordination of cells is achieved via a chemical cell dialogue
- a switch into a mode of self-maintenance can be made after maturity has been reached
- fitness tests are punctual
- time and dynamics play an essential role
- and organism grows from a 1-cell stage which allows for sexuality.

Many of these features can be explored in computer models (see for example [5]), and a whole new area, computational development is presently forming itself. Before moving any further, we should check with biologists about what development is in their mind. I'd like to adopt the following definition of development for the purposes of this paper.
- Development is a differential transcription (and translation) process of genes in different cells and tissues at different times and with different rates.
- Each step in this sequence is ultimately initiated by the transcription and translation of the previous step.
- Diversity of body plans in all organisms is caused by
  - Interaction between gene products
  - Shifts of timing of gene expression: Heterochrony
  - Shifts in location of gene expression: Differentiation
- Implementation of development is realized using regulatory networks.

This last point is most notable, as it underlines the earlier reference to networks as the means for anchoring emergent phenomena. A closer look at the mechanisms of development and especially the role of genes in the process reveals:

- Genotypes are different from phenotypes
- Genes orchestrate the interaction of molecules (by regulation)
- Most molecules are available from the environment
- Some molecules are produced by genes
- Gene products are most often used as "amplifiers" injected at crucial branching points

The interface between genomes and bodies, and thus the implementation method for embodiment is provided by regulatory networks. They are the means of Nature to stabilize emergent phenomena of construction. How does it work? In order to explore this question, we have suggested a model for artificial regulatory networks that builds on an earlier model proposed by T. Reil [6].

4 An Artificial Regulatory Network

The model (see [7, 8] for details) consists of a genome of bits generated by a random process. This process comes in two alternative implementations, one being a simple seeding of the bits by a process randomly determining the value in each bit position. The other process is more sophisticated and consists of 2 different phases that are iteratively applied, until the prescribed length of the genome is reached: Starting from a small genome kernel, again seeded by randomly choosing values, it loops through successive stages of duplication and divergent mutation until final length is reached.

In a second process, meaning is ascribed to bits of the genome. Figure 3 describes the situation. By scanning the genome, certain bit patterns are isolated which consist of sub-strings of small size. These sub-strings are called promoters, and they determine the open reading frame of genes. To make things simple, genes are of fixed length, and thus a fixed number of bits subsequent to the promoter are expressed by applying a genotype-phenotype mapping. After mapping, which results in another bit pattern of fixed length, a protein has been produced. The main feature of this protein is that it is mobile. Upon it wandering around it might encounter other proteins, or it might hit the genome at any place of chance. Depending on the pattern match between the protein and the genome at the position of encounter, an interaction occurs in the following way: The protein will attach at the genome, and will detach again after some time. The time of attachment will be all the longer, the better the (complementary) match between protein and genome is.

A third feature needs to be mentioned to understand what is happening in such a system: Upstream from the promoter site of a gene (we envisage only very few genes relative to the number of bits on a bit string genome), there are special sites called enhancer and inhibitor sites. Occupation of these sites with proteins will have a profound effect on the efficiency of expression of the corresponding gene. The effect will depend in a nonlinear way on the matching between genome and proteins trying

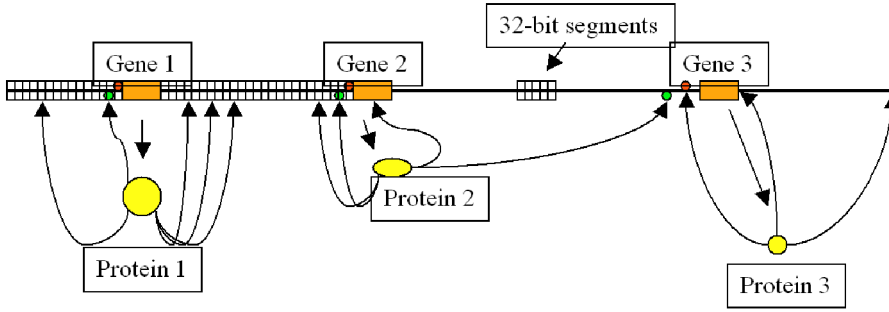

Fig. 3. The artificial regulatory network. A strand of bits, the genome contains short sequences signaling the beginning of a gene. Genes are expressed into proteins which subsequently can wander around and attach to the genome, specifically at regulatory sites upstream from genes, where they influence the rate of expression of other genes. Attachment is controlled in turn by the degree of complementarity between pattern on the genome and pattern on the protein.

to attach. The result of this process, in connection with a time-scale for production of new proteins via genotype-phenotype mapping is that particular inhibitor / enhancer sites are occupied more often and thus have more influence on the expression of the corresponding gene than others. Readers interested in details and quantitative considerations need to compare recent publications [7, 8].

It can be said, that the bit genome discussed and its gene products constitute an artificial regulatory network: The amount of each protein is determined by the matching between proteins and its regulatory site on the genome. We can imagine genes to be nodes in a network, with proteins building the links between those nodes, and the weight on links being the interaction strength determined by the degree of matching between a protein and the regulatory site of a gene.

Figure 4-6 show images of networks resulting from the second generation process (duplication and divergence) mentioned above at various stages of resolution. Figures have been drawn among network nodes that exceed a certain interaction threshold only. The lower the threshold, the more nodes (genes) come into the play, and the more intricate the connection pattern becomes. It is interesting – though not surprising - to see that above a certain threshold the network decays into unconnected components of smaller and smaller size.

At this point, my earlier statement regarding adaptation using side-effects (exaptation) probably becomes clearer: The strongest interactions basically show a network of unconnected components, mostly two genes interacting with each other. Suppose for the moment this would be all that is there. Thus, a modular structure is present in which each pair of genes can be put to independent, yet functionally similar use by evolution.

Lowering the threshold only a bit shows a different picture: New nodes come into play that did not have a role at the former threshold level. And there is a second effect: Independent modules become connected into larger units, thus there is crosstalk between modules. Both effects can be molded by evolution in an arbitrary way. If

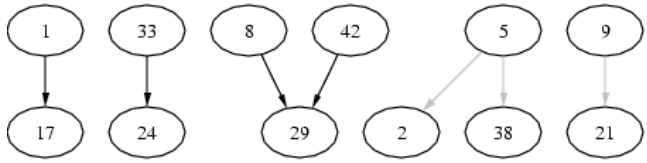

Fig. 4. ARN, generated by a process of duplication and divergence events. High threshold of matching between nodes required. Network decays into very simple modules of 2 to 3 nodes.

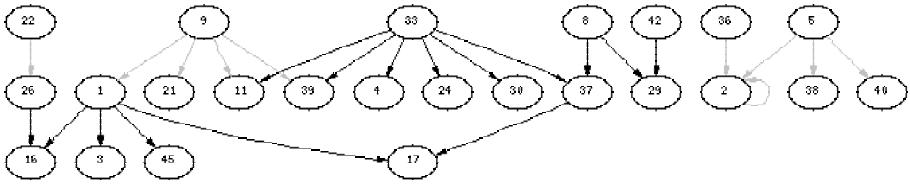

Fig. 5. ARN, generated by a process of duplication and divergence events. Lower threshold of matching between nodes required. Network contains both connected and non-connected components.

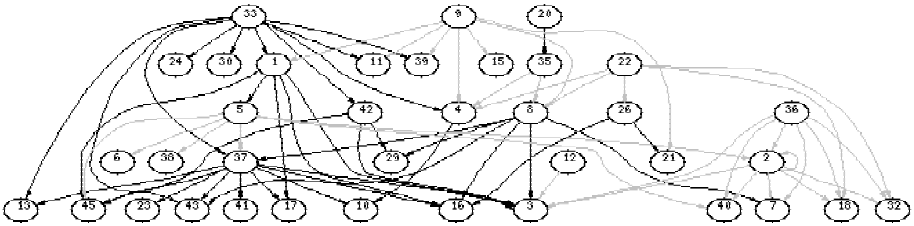

Fig. 6. ARN, generated by a process of duplication and divergence events. Low threshold of matching between nodes required. Network is fully connected and shows complex organization.

there is a need for new function one of the new genes in the game could be assigned to such use. If, on the other hand, it should turn out that crosstalk between "modules" has a beneficial effect on the overall system, this crosstalk could be elevated by increasing the interaction strength between the participating modules. This is the stuff evolution likes the most: Rich behavior yet smooth transitions between alternatives.

How would "molding" by evolution actually work? Simply by mutating the regulatory sites upstream from a gene, the interaction with particular proteins (and thus other genes) can be strengthened. A single bit flip would already elevate (or decrease) an interaction one step, allowing this interaction to become visible in a picture drawn again after the mutation. Possible side-effects of the mutation notwithstanding, very smooth transitions between network configurations are realized.

What has been numerously stated in regard to the evolution of modularity has probably become clearer with our example of an artificial regulatory network: Mod-

ules in nature are isolated from each other only to a certain degree. There is always a weak interaction between modules which effectively blurs the distinction between them and provides the rich material evolution likes to work with.

5 Embodiment

There are various notions of embodiment. In this contribution I have tried to argue that emergent phenomena can be "embodied" in networks which in turn are subject to evolutionary forces of variation and selection. If we step back a bit and look at bodies in the literal sense, we might adopt the same perspective: Bodies are so important for active entities, for adaptation, learning and intelligence, because bodies allow the environment to network with the system. I.e. bodies at least partially remove the isolation of an otherwise (machine-like) entity. Trying to achieve intelligent functions without this "crosstalk" between bodies and the environment is a typical human enterprise bound to fail.

Acknowledgments. I gratefully acknowledge discussions and joint work with Julian Miller on aspects of development and evolution. Discussions and joint work with my student P.Dwight Kuo at Memorial University have also contributed to posing questions raised here.

References

1. M. Conrad, Adaptability: The significance of variability from molecule to ecosystem. Plenum Press, New York, 1983.
2. R. Rosen, *Life Itself: A comprehensive inquiry into the nature, origin and fabrication of life*. Columbia University Press, New York,1991.
3. S.J. Gould, *The Structure of Evolutionary Thought.* Harvard-Belknap Press, Cambridge, 2003.
4. W. Banzhaf and J. Miller, *The challenge of complexity*, in A. Menon (Ed), Hilbert Challenges to Evolutionary Computing, Kluwer, Norwell, 2004.
5. J. Miller and W. Banzhaf , *Evolving the program for a cell: From French Flags to Boolean Circuits*, in S. Kumar and P. Bentley (Eds), On Growth, Form and Computers, Academic Press, New York, 2003.
6. T. Reil, *Dynamics of Gene Expression in an Artificial Genome*, in D. Floreano, J.-D. Nicoud, F. Mondada (Eds), Advances in Artificial Life, 5th European Conference ECAL-1999, Springer, Berlin, 1999.
7. W. Banzhaf, Artificial *Regulatory Networks and Genetic Programming,* in R. Riolo, B. Worzel (Eds), Genetic Programming in Theory and Application, Kluwer, Norwell, 2003.
8. W. Banzhaf, *On the dynamics of an artificial regulatory networks*, in W. Banzhaf, T. Christaller, P. Dittrich, J. Kim, J. Ziegler (Eds), Advances in Artificial Life, 7th European Conference ECAL-2003, Springer, Berlin, 2003.