

7. Modeling Evolutionary Systems

“How does evolution produce increasingly fit organisms in environments which are highly uncertain for individual organisms? How does an organism use its experience to modify its behavior in beneficial ways (i.e. how does it learn or ‘adapt under sensory guidance’)? How can computers be programmed so that problem-solving capabilities are built up by specifying ‘*what* is to be done’ rather than ‘*how* to do it’?” [Holland, 1975, page 1]

These were some of the questions concerning John Holland when he thought of Genetic Algorithms (GA's) in the 1960's. All these questions were shown to be reducible to a problem of optimizing multi-parameter functions. Nature's “problem” is to create organisms that reproduce more (are more fit) in a particular environment: the environment-organism coupling dictates the selective pressures, and the solutions to these pressures are organisms themselves. In the language of optimization, the solutions to a particular problem (say, an engineering problem), will be selected according to how well they solve that problem. GA's are inspired by natural selection as the solutions to our problem are not algebraically calculated, but rather found by a population of solution alternatives which is altered in each time step of the algorithm in order to increase the probability of having better solutions in the population. In other words, GA's, or other Evolutionary Strategies (ES) such as Evolutionary Programming (EP), explore the multi-parameter space of solution alternatives for a particular problem, by means of a population of encoded strings (standing for alternatives) which undergo variation (crossover and mutation) and are reproduced in a way as to lead the population to ever more promising regions of this search space (selection) [Goldeberg, 1989; Mitchell, 1999; De Jong, 2006].

7.1 Evolutionary Strategies and Self-Organization

The underlying idea of computational ES is the *separation* of solutions for a particular problem (e.g. a machine) from *descriptions* of those solutions (memory). GA's work on the descriptions and not on the solutions themselves, that is, variation is applied to descriptions, while the respective solutions are evaluated, and the whole (description-solution) selected according to this evaluation. Such machine/description separation follows von Neumann's self-reproducing scheme (see chapter 6) which is able to increase the complexity of the (organization of) machines described. Therefore, the form of organization evolved by GA's is not self-organizing in the sense of a boolean network or cellular automata (see chapter 4). Even though the solutions are obtained from the interaction of a population of elements, and in this sense following the general rules usually observed by computationally emergent systems (e.g. Langton [1988], Mitchell [1992]), they do not *self-organize* since they rely on the *selective* pressures of some environment (in ES, defined by an explicit or implicit fitness function). The order so attained is not a result of the internal dynamics of a collection of interacting elements, but is instead dictated by the *external* selection criteria. In this sense, ES follow an organizing scheme that is driven by external selection of encoded symbolic descriptions (a “Turing tape”). It is perhaps useful to think that ES are modeling the most fundamental design principle of biological systems: natural selection. While self-organizing systems model the dynamical characteristics of matter, ES model the existence of external, selective pressures on populations of symbolic descriptions of some system. While self-organization models material dynamics, ES models the selection of information (encoded in stable, effectively inert, memory structures [Rocha and Hordijk, 2005]) about dynamics.

7.2 Development and morphogenesis: self-organization and selection come together

Since the original introduction of GA's, many subsequent developments had to do with the inclusion of a developmental stage, or intermediate layers between genotype and phenotype; in other words, the creation of some artificial morphogenesis or regulation. The idea has been to encode rules that will themselves self-organize to produce a phenotype, rather than the direct encoding of the phenotype itself, or the introduction of gene regulation. As discussed in class, these rules often use L-System grammars which dictate production system programs [Wilson, 1988] leading to some phenotype. The most important advantage of this intermediate stage, as initially explored by Kitano [1990], Gruau [1993], Belew [1992] and others, is the ability to code for much larger structures than a direct encoding allows. In practical terms, they have solved some of the scalability problems of encoding (e.g.) neural networks in GA's, by reducing the search space dramatically.

L-system grammars are higher-level descriptions of self-organizing developmental processes. However, these first approaches used solely context-free, state-determined, L-System grammars, compromising epistasis (or mutual, non-linear, influence of genetic descriptions amongst each other) in the simulation of self-organizing development. Dellaert and Beer [1994] and Kitano [1994], for instance, used Boolean networks to simulate genetic epistasis and self-organization. In other words, the GA encodes rules which construct Boolean networks whose nodes stand for aspects of the phenotypes we wish to evolve on some physical simulation. In Dellaert and Beer's model, the nodes stand for cell mitosis and other characteristics. This way, the solutions of the GA are self-organizing systems whose attractor behavior dictates pre-defined phenotypic traits.

These approaches in effect offer an emergent morphology, that is, they encode rules which will themselves self-organize into some phenotype (instead of strict programming of morphology). The indirect encoding further allows the search to occur in a reduced space, amplified through development. An interesting side effect has to do with the appearance of modularity traits on the evolved phenotypes [Wagner, 1995]. Subsequent developments paid even more attention to the contextual regulation that indirect encodings afford to the search [Rocha 1995, 1997]. More recently, given our expanded view of genomics, other intermediate layers between genotype and phenotype have been explored, such as transcription regulation [Reil, 1999; Hallinan & Wiles, 2004] and RNA Editing [Rocha et al, 2006]. The inclusion of more sophisticated regulation of genetic information prior to translation, while not necessarily including a self-organizing component, allows us to model a much more realistic genotype/phenotype/environment interaction. Instead of genotypes used exclusively for Mendelian inheritance (see chapter 5) of (directly encoded) phenotypic traits, ES with genotype regulation allow us to model the contextual, plastic development of phenotypes we have come to understand via modern Genomics—thus also learning additional design principles for bio-inspired computation [Huang et al, 2007].

The most important aspect of GA's with emergent morphologies is the utilization in the same model of an external selection engine (the GA) coupled to a particular self-organizing dynamics (e.g. Boolean networks) standing for some materiality. Such schemes bring together, computationally, the two most important aspects of evolutionary systems: self-organization and selection. These models belong to a category of self-organization referred to as *Selected Self-Organization* which is based on symbolic memory [Rocha, 1996, 1997, 1998]. Selected Self-Organization with distributed memory is also possible in autocatalytic structures, though its evolutionary potential is much smaller than the local memory kind [Rocha, 2001][Vasas, 2010]. The reason lies in Von Neumann's notion of self-reproduction (see chapter 6). The introduction of symbolic descriptions allows a much more sophisticated form of communication: structures are constructed from static

descriptions and do not have to reproduce through some complicated, and limited process of self-inspection. In other words, separate descriptions (stored in effectively inert memory structures like DNA) can be used to reliably construct any kind of structure in an open-ended manner, while self-inspection relies on only those structures that happen to be able to make copies of themselves. As an example, a non-genetic protein-based life form, would have to rely only on those proteins that could make direct copies of themselves [Rocha, 2001].

7.3 Multi-level complexity of evolutionary systems

Throughout this course emphasis was put on identifying the most important tools utilized in the field of Artificial Life. We started with *self-organizing systems*, exemplified with the logistics equation, random boolean networks, cellular automata (e.g. Conway's game of Life), and all characterized in terms of *dynamical systems theory*. Later, with the *von Neumann self-reproduction scheme*, I argued that state-determined (purely dynamic) systems are not able to offer *open-ended evolution*, that is, to increase their complexity with genuine emergence of new functionalities. Dynamic systems are restricted to the complexity of their attractor landscape.

For this purpose, systems inspired by von Neumann's scheme, which demand the separation between the description of a machine from the machine itself, and therefore introduce the concept of *memory* and external selection, were introduced. Such systems offer a model of the mechanisms utilized by natural selection, and are accordingly known as *evolutionary systems* (or evolutionary strategies) — e.g. genetic algorithms and evolutionary programming. We can also refer to the mechanisms utilized to model the kind of evolution that natural selection offers as *memory-based selective strategies*: selection acting on memory elements in order to change the dynamic structure they encode.

I further emphasized hybrid systems which try to model both the self-organizing and selective mechanisms of biological systems, and can therefore offer a more complete understanding of evolutionary systems. These memory-based, selected self-organizing systems, follow a view of evolutionary systems based on the *semiotic closure* of genotype-phenotype maps [Pattee & Raczaszek-Leonardi, 2012], which are rather open to control [Rocha, 2000] and prone to be organized in multiple scales of interaction (multi-level complex networks) [Pattee, 1973]. For practical applications, I showed approaches aiming at the introduction of non-deterministic, self-organizing, developmental steps between genotype and phenotype such as the evolution of boolean/neural networks encoded through L-System rules in a genetic algorithm. Also discussed were models capable of emergent computation by coupling genetic algorithms to cellular automata in order to have the latter solve non-trivial tasks, and the dynamics of adaptative immunity that harnesses a "private" natural selection of lymphocytes in vertebrates.

The understanding of the relative importance the two basic categories of organization in artificial systems introduces a very powerful way to study the relative importance of self-organization and natural selection in biological systems themselves. In other words, by creating different forms of *life-as-it-could-be* with different degrees of both these categories, we may shed some light on the credit assignment problem of biology: how much of evolution is a result of natural selection and how much is a result of the self-organizing characteristics of its specific materiality.

I was able to introduce many of the usual applications of Artificial Life and Bio-Inspired Computing, from bugs and boids, to evolutionary robots, social evolution, and the multi-level complexity of adaptive immunity. Each of these applications can be a universe of investigation in itself, so emphasis was instead put

on the basic categories of organization and their respective simulation tools referred above. In one way or another, all of these applications utilize in different degrees such tools described throughout the course. For instance, evolutionary robots may use a genetic algorithm to evolve a Boolean network for its control system allowing it to solve some maze. To the extent that its control system was evolved and uses self-organizing mechanisms, we can say that such control system was subjected to a memory based selective type of self-organization. Naturally, the robot itself (its moving parts and sensors) were not evolved but engineered; the complete evolution of a robot through self-organization and selection represents probably the most ambitious long-term goal of Artificial Life, showing us how far behind we still are from getting there.

Further Readings and References:

- Belew, R.K. [1993]. "Interposing an Ontogenic Model Between Genetic Algorithms and Neural Networks." In: *Advances in neural information processing (NIPS5)*. J. Cowan (Ed.). Morgan Kaufmann.
- De Jong, K.A. [2006]. *Evolutionary Computation: A Unified Approach*. MIT Press.
- Dellaert, F. and R.D. Beer [1994]. "Toward an evolvable model of development for autonomous agent synthesis." In: *Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*. R. Brooks and P. Maes (Eds.). MIT Press.
- Goldberg, D. E. [1989]. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley.
- Gruau, Frédéric [1993]. "Genetic Synthesis of Modular Neural Networks." In: *Proceedings of the fifth international conference on Genetic Algorithms*. S. Forrest. Morgan Kaufman, pp 318-325.
- Hallinan, J. And J Wiles [2004]. "Asynchronous Dynamics of an Artificial Genetic Regulatory Network". *9th International Conference on Artificial Life*. MIT Press.
- Holland, John H. [1975]. *Adaptation in Natural and Artificial Systems*. University of Michigan Press.
- C. Huang, J. Kaur, A. Maguitman, L.M. Rocha [2007]. "Agent-Based Model of Genotype Editing". *Evolutionary Computation*, 15(3).
- Kitano, H. [1990]. "Designing Networks using Genetic Algorithms with Graph Generation System." In: *Complex Systems* Vol. 4, pp 461-476.
- Kitano, Hiroaki [1994]. "Evolution of Metabolism for Morphogenesis." In: *Artificial Life IV: proceedings of the fourth international workshop on the synthesis and simulation of living systems*. R. Brooks and P. Maes (Eds.). MIT Press.
- Mitchell, M. [1992]. "Genetic algorithms". In *Lectures in Complex Systems*. L. Nadel and D. Stein (Eds.). SFI Studies in the Science of Complexity Vol. V, Addison-Wesley. pp 3-87.
- Mitchell, M. [1999]. *An Introduction to Genetic Algorithms*. MIT Press.
- Pattee, H. H. (1973). *Hierarchy theory: The Challenge of Complex Systems*. Braziller.
- Pattee, H. H., & Raczaszek-Leonardi, J. [2012]. *Laws, language and life: Howard Pattee's classic papers on the physics of symbols with contemporary commentary* (Vol. 7). Springer Science & Business Media.
- Reil, T. 1999. "Dynamics of Gene Expression in an Artificial Genome - Implications for Biological and Artificial Ontogeny". In *Proceedings of the 5th European Conference on Advances in Artificial Life* (September 13 - 17, 1999). D. Floreano, J. Nicoud, and F. Mondada, Eds. Lecture Notes In Computer Science, vol. 1674. Springer-Verlag, London, 457-466
- Rocha, Luis M. [1995]. "Contextual Genetic Algorithms: Evolving Developmental Rules." In: *Advances in Artificial Life*. F. Moran, A. Moreno, J.J. Merelo, and P. Chacon (Eds.). Series: Lecture Notes in Artificial Intelligence, Springer-Verlag. pp. 368-382.
- Rocha, Luis M. [1996]. "Eigenbehavior and symbols." In: *Systems Research* Vol. 12, No. 3, pp. 371-384.
- Rocha, Luis M. [1997]. *Evidence Sets and Contextual Genetic Algorithms: Exploring Uncertainty, Context, and Embodiment in Cognitive and Biological Systems*. PhD. Dissertation. SUNY Binghamton.
- Rocha, Luis M. [1998]. "Selected Self-Organization and the Semiotics of Evolutionary Systems." In: *Evolutionary Systems: The Biological and Epistemological Perspectives on Selection and Self-Organization*. S. Salthe, G. Van de Vijver, and M. Delpo (eds.). Kluwer Academic Publishers, pp. 341-358.
- Rocha, Luis M. [2000]. "Syntactic autonomy, cellular automata, and RNA editing: or why self-organization needs symbols to evolve and how it might evolve them". In: *Closure: Emergent Organizations and Their Dynamics*.

- Chandler J.L.R. and G, Van de Vijver (Eds.) *Annals of the New York Academy of Sciences*. Vol. 901, pp 207-223.
- Rocha, Luis M. [2001]. "Evolution with Material Symbol Systems". *Biosystems*. Vol. 60, pp. 95-121.
- Rocha, Luis M. and W. Hordijk [2005]. "Material Representations: From the Genetic Code to the Evolution of Cellular Automata". *Artificial Life*. **11** (1-2), pp. 189 - 214
- Rocha, L.M., A. Maguitman, C. Huang, J. Kaur, and S. Narayanan. [2006]. "An Evolutionary Model of Genotype Editing". In: *Artificial Life 10: Tenth International Conference on the Simulation and Synthesis of Living Systems*. L.M.Rocha, L. Yaeger, M. Bedau, D. Floreano, R. Goldstone, and A. Vespignani (Eds.). MIT Press, pp. 105-111.
- Vasas, Vera, Eors Szathmary, and Mauro Santos [2010]. "Lack of evolvability in self-sustaining autocatalytic networks: A constraint on the metabolism-first path to the origin of life. ". *Proceedings of the National Academy of Sciences of the United States of America*: 0912628107.
- Wagner, Gunter [1995] "Adaptation and the modular design of organisms". In: *Advances in Artificial Life*. F. Moran, A. Moreno, J.J. Merelo, and P. Chacon (Eds.). Series: Lecture Notes in Artificial Intelligence, Springer-Verlag. pp. 317-328.

