A Regulation Network based Framework for Evolving Biological Models

Uri Yerushalmi, Mina Teicher
The Leslie and Susan Gonda Interdisciplinary Brain Research Center, Bar-Ilan University, Ramat-Gan, Israel
uri.yerushalmi@gmail.com

Abstract. Over the years many attractive evolutionary development models have been suggested, but none of them had the adequate amount of plausible biological details while at the same time requiring reasonable computing power to enable intensive biological study. This paper reports about a comprehensive framework for the development, design and analysis of simulative evolving biological systems. The framework is based on common biological principles like genomic & proteomic regulation, mitosis & meiosis, cellular development, neural development and behavioral based fitness. Several experiments are presented, all based on the presented framework, with different development and fitness policies.

Introduction & Related Work

In recent years intensive effort has been put into simulating evolutionary biological processes as well as small time scale biological processes. However, most of the simulation systems were either focused on a specific time scale, based on a restricted set of properties or oversimplified the biological processes in order to ease the simulation process.

The simplest principle on which simulations of evolutionary processes are based is genotype based selection, where the fitness is a direct function of the genotype. However, in nature, the fitness is based on the organisms' phenotype, that despite being encoded as a genotype, is a separated entity.

The most common encoding used to evolve phenotypes from genotypes is known as direct encoding, where the phenotype details are directly encoded in the genome. In most of these cases, the architecture of the phenotype is decided ahead of time [17].

One way to avoid direct encoding is to parameterize encoding. When evolving neural networks, this may include encoding in the chromosome number of layers, size of layers, and other general parameters. This approach has been tried in [11] and is restricted to predefined architectures, and non-modularity. In [20] the author evolved developmental parameters which lead to the creation of edge-detecting retina. The same simulator was used later in [21] to grow single neurons within developmental environment rich with gradients of chemical attractants.

In the grammar encoding method the genome encodes a set of grammar rules that are used to build the phenotype. The main example for such an encoding is presented
in [14]. In [23, 24] the author described a grammar system for the evolution of artificial creatures that compete in a physically realistic simulation of a three-dimensional artificial world. Grammar encoding approaches have the advantage of not being difficult to use as the size of the desired network increases. In addition, repeated or nested structures can be represented efficiently.

Based on the advantages presented by grammar encoding, Cellular encoding was introduced in [10]. In this model, the object on which the grammatical rules are applied is a cell. Each cell has a copy of the genome, which encodes directly a grammar tree. Each cell reads the grammar tree at a different position. Depending on what the cell has read, it may divide and change its internal parameters.

A similar grammar-based model for generating neural phenotype is presented in [2]. The genotype contains a set of developmental instructions, some of which become "relevant" to the particular context in which each developing cell finds itself. The development starts with a germ cell that is represented by the start symbol of the grammar. And a series of production rules, encoded in the chromosome, specify the mitosis of this cell into two other cells represented by two grammar symbols. All cells develop according to the same set of genetically determined rules until all cells are in a mature, terminal state.

However, the grammar-based and cellular-based encodings presented above lack the dealing with high interactions among the different developmental phenomena, what has lead to emergence of more biologically plausible models.

In [5] the authors show that a simulation of complex biologically inspired development is possible and can be successful, by evolving an organism in many cycles of cell division, differentiation, and axonal growth. However, hand written genetic instructions are used to control the organisms' development.

A computational model of neurogenesis based on metabolic processes is tested and proved to be capable of evolving large neural networks in [15]. A different system for evolving 3D organisms using gene expression mechanisms is presented in [7]. Models of development that can be used to evolve functional autonomous agents, are presented also in [6] and [1]. However, despite the promising direction, a deep examination of evolution that is based on biologically plausible behavior selection has not been presented using such models.

It is expected however [19], that success in finding an efficient indirect biological encoding model should provide us with simulative tools that will probably teach us a lot about the organization and functioning of biological systems.

The Modeling Framework

The presented framework was designed to enable an easy development of complex simulations of evolutionary biological processes. The framework includes software components that, together, are useful in simulating, designing and analyzing such processes. It is designed to force developers to implement reusable, consistent and reliable code, and to enable researchers to design and analyze simulative experiments in a communicative and consistent methodology.
The framework's components enable the simulation of a population of organisms in a 2D environment, where each organism's phenotype is encoded as a genotype and may include a neural network that controls the organism's behavior in the environment.

Genotype structure

Each organism in the model has a single chromosome that encodes its phenotype, and includes a sequence of genes, whose content, structure and length may vary during evolution. Each gene includes a promoter sequence followed by a messenger RNA sequence. Each promoter sequence includes several cis-regulatory elements, and an element that includes the gene parameters. Each mRNA sequence includes a cis-regulatory element, followed by a parameters sequence, followed by a trans-acting element, all represent a translated protein.

All elements in the chromosome are represented in the computer simulation as real numbers, the cis & trans elements are sequences of 16 real numbers, and the parameter blocks are sequences of 12 real numbers.
Decoding the genotype to a regulatory network

The chromosome presented above can be translated to a regulatory network where the nodes represent genes, proteins or other regulated elements, and the connections represent regulation associations. The network connections’ strengths \( w_{ij} \) are assigned according to the Hamming distance \( d_{ij} \) between the round values of the cis-regulatory elements and trans-acting elements (cis & trans sequences are real values between 0 and 1). Each gene and each protein transcripted holds several parameters that are read from the chromosome and control its dynamics: activation threshold \( \theta \), activation slope \( \beta \), activity \( x \), diffusion coefficient \( k \), time coefficient \( r \), and a Boolean parameter \( b \), that govern the translated protein’s anchoring type on the membrane (in cell, out of cell, entering receptor, exiting receptor). The regulatory network controls protein’s concentration \( v_j^{con} \) and activity level \( v_j^{act} \) in a cell, and it’s concentration out of the cell \( v_j^{con} \) as follows:

\[
\frac{\partial v_j^{act}}{\partial t} = f_{x,b} \left( \sum_j \alpha_j w_{ij} v_j^{con} \psi_j^{act} \right) + \delta_{x,con} k_i \nabla^2 v_i^{act} - v_i^{act}
\]

(1)

where \( f_{x,b}(h) = \frac{1}{2} \left( 1 + \tanh \left( b(h - \theta) \right) \right) \),

\( v_i^{act} \) can be one of \( v_i^{con}, v_i^{act}, v_i^{con} \).

\( v_j^{con} \) is either the internal concentration \( v_j^{con} \) or external one \( v_j^{con} \), according to the values of \( x \) and \( b \), which makes the model capable of evolving receptor-ligand relationships, based on the parameter \( b \).

\( \delta_{x,con} \) is 1 in case of external concentration and 0 otherwise, \( k \) is the diffusion coefficient of \( i \), and \( \nabla^2 = \sum_{x,y,z} \frac{\partial^2}{\partial x^2} \), so the expression \( \delta_{x,act} k_i \nabla^2 v_i^{act} \) represents the contribution of diffusion to the change in external concentration, according to the diffusion equation \( \frac{\partial v_i^{act}}{\partial t} = k \nabla^2 v_i^{act} \).
The process of successor genotype production

Each real value in \( r_i \) of the chromosome is surrounded by several other values: mutation strategic parameter \( \sigma_i', \) crossover probability parameter \( c_i, \) and crossover strategic parameter \( \sigma_i \). The values of \( r_i, c_i, \sigma_i', \sigma_i \) are mutated self-adaptively [25]:

\[
\hat{\sigma}_i' = \sigma_i' \exp(\tau' N(0,1) + \tau N_i(0,1)) \\
\hat{x}_i = x_i + \hat{\sigma}_i' N_i(0,1) \\
\tau = \left( \frac{\sqrt{2/n}}{\sqrt{2/n}} \right)^i, \tau' = \left( \frac{\sqrt{2/n}}{\sqrt{2/n}} \right)^i
\]

Where \( x \in \{ r_i, c_i \}, i = \{1..n \} \), \( N(0,1) \) is a standard normal random number, \( N_i(0,1) \) represents a new random number generated for each component, and \( \sigma_i', \hat{x}_i \) are the new values for \( \sigma_i, x_i \). \( \sigma_i' \) & \( x_i \) may have lower and upper bounds, for example: \( c_i \in (10^{-5}, 0.5) \).

Before mutation takes place, the parent chromosomes are aligned using a dynamic programming algorithm [9] and recombined according to the average \( c_i \) of the aligned chromosomes. There is also a chance for chromosome structural mutations, like translocation and inversion; the cut & insertion locations are chosen proportionally to \( c_i \) also.

Cell development functions mechanism

The regulation network controls the organism's cells through functions such as cell death, mitosis, cell migration, and differentiation. To achieve this, each function is represented by a random-generated bit string \( m \). The nodes \( j \) in the regulation network that are close enough to the string \( m \) are marked as triggers for the specified function. The internal value of the function \( m \) is defined as:

\[
u_m = \sum_j \alpha_j \omega_j \gamma_j \gamma_j' \quad (\theta_{\alpha} = 0.5, \beta_{\alpha} = 1)
\]

For functions that trigger an event (e.g., occurrence of a mitosis, cell death, migration, differentiation event), the function is triggered when the value \( \nu_m \) passes a pre-defined threshold (0.5).
Organism development & neurogenesis

Each organism is initially made up of a single cell and during its lifetime it is composed of many cells, each one of which is controlled by the same chromosome (i.e., has the same controlling regulatory network), but may have different internal & external concentrations.

The organism has a certain period of time in which it has to stop mitosis, only then will the organism be an adult that may reproduce. In case the organism does not stop mitosis during the predefined period it is regarded as a cancerous tissue and removed from the environment without reproduction.

The development of the organism's nervous system [22] is based on the cell development function mechanism presented above. Using this mechanism, functions are defined for differentiation for motor, sensor and hidden cells, for neurite sprouting, axonal growth and guidance, for target selection, and also for synaptic transmission regulation that enables synaptic plasticity.

The framework supports two models of neuronal activity: a simple McCulloch & Pitts model [16], where the local field of a neuron is 0 or 1, and a more biological, though CPU consuming, model based on integrate and fire neurons [13].
Organism behavior in its environment

The sensory neurons supported in the framework can be treated as olfactory neurons. Each time a sensory neuron of an organism gets close enough to an odorant source, the sensory neuron is activated in a level which is proportional to its distance from the source. The framework supports coupling odors to different elements in the environment (organisms, food-objects, poison-objects, obstacles). Each sensory neuron can be sensitive to one odor only.

When fired, the motor neuron supported in the framework make the organism move in a direction that is proportional to its neurons' distance from the organism's centroid.

Each organism is provided initially with a certain amount of energy units. Every epoch in the environment decreases the organism's energy in an energy unit. Whenever number of organisms exceeds some limit, the organisms that have the least energy are removed from the arena.

Preliminary Results

The following experiments were based on the presented framework, where the cellular matrix of the organisms is limited to a 15x15 cell grid, and there are maximum 100 organisms in the environment simultaneously.

Evolution of organisms with wired neurons

An initial population of random chromosomes was evolved to have certain properties in a dynamic environment. Each individual received a fitness value that determined its life period. The selection was based on roulette wheel selection [8]. The fitness function was different in each one of two periods:

1. A period aiming for development of organisms with enough cells. The fitness value in this period is proportional to the number of cells of the organism, this period ended when the average organism cell count got to a certain threshold.

2. A period aiming for organisms with motor, sensory, hidden and undifferentiated neurons. Only individuals with more than a certain number of cells get a positive fitness value which is higher as the organism has more cell types.
3. A period aiming for development organisms with enough wired neurons. The fitness value in this period is proportional to the number of synapses of the organism. As presented in figure 4, we succeeded evolving an initial population of random chromosomes to a population of chromosomes that encode wired neural networks with about 300 synapses.

![Figure 3](image)

**Fig. 3.** An evolutionary session divided to 3 different fitness policies. A) Average cell number per organism during first period. B) Proportion of different cell type organisms around second period. C) The rate in the average synapse number in the last period. The arrows point to the epoch when the fitness policy changed.

**Evolving behaving organisms**

We have also used the chromosomes of the evolved organisms in figure 4 as an initial population of organisms with McCulloch & Pitts neurons for evolving organism behavior. Therefore, instead of using a roulette-wheel a fitness, and endogenous fitness was defined so that each time two organisms meet (i.e. an organism gets close enough to another) their genes are used to reproduce a child gene, and both parents and child organisms are relocated in the environment. As seen in figure 4, we could notice evolution of organisms that tend to attract each other.
Fig. 4. Increase in the number of cases in which two organisms met every time unit in the evolving behaving organisms experiment.

Discussion

The above results presented a software tool that can be useful in evolving and analyzing the evolution of detailed biologically inspired systems. Since the model embedded in the presented framework is relatively biologically detailed, we believe it would be helpful in future experiments examining relations between mechanisms in various scales that cannot be examined physically: microbiologic, neurophysiologic and behavioristic, all in the light of evolution.

References