Evolving gene regulatory networks for control of artificial cells: signal processing, chemotaxis, multicellular development

Michał Joachimczak[1] and Borys Wróbel[1,2]
[1]Systems Modeling Laboratory, Institute of Oceanology, Polish Academy of Sciences
[2]Evolutionary Systems Laboratory, Adam Mickiewicz University in Poznań, Poland
{mjoach,bwrobel}@iopan.gda.pl

Abstract

We provide an overview of our research on modeling the evolution of artificial gene regulatory networks. Networks in our model are encoded in linear genomes comprising regulatory elements and genes that encode transcription factors (TFs). The nodes of the network are regulatory units (one or more regulatory elements and one or more genes), the links correspond to regulatory interactions, and the weights depend on the affinity between the TFs and the regulatory elements. The expression of a particular regulatory unit determines the continuously changing concentrations of the TFs that the unit encodes. This expression depends on the concentration of the regulating TFs and their affinity to the unit's regulatory elements. We summarize the results of simulated evolution to obtain networks that perform various types of computation with the emphasis on processing of continuous external signals. We also review our results on using the evolving gene regulatory networks to control the behavior of unicellular organisms and multicellular development.

Introduction

Recent interest in gene regulatory network (GRN) models is on one side driven by the growing amount of data on biological GRNs, and on the other, by the hope to exploit the computational properties of GRNs and their robustness in various control domains. Also, there's a growing interest in the design of synthetic biological GRNs capable of performing desired tasks and to react adaptively to external stimuli.

In this work we present a brief review of our previous work on the evolvablity of our model of evolving artificial GRN in various types of problems.

The model

Genomes in our model are lists of genetic elements, each representing either genes (which encode products, transcriptional factors; TFs) or regulatory elements. Series of regulatory elements (promoters) followed by series of genes are grouped into regulatory units which form nodes in the artificial gene regulatory network (see Fig. 1). Connectivity between the nodes in the GRN is determined by the affinity between the TFs and the regulatory elements. Each genetic element (Fig. 1, right) is defined by a set of real values: one value determines the type of the element, one is an affinity

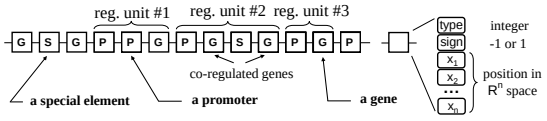

Figure 1: Structure of the genome. The genome (left) is a list of genetic elements with an internal structure shown on the right. Regulatory units consist of at least one promoter followed by at least one gene. Special elements code for inputs and outputs of the network.

modifier (sign) and the remaining values are the coordinates of a point in abstract R^N space (typically $N = 2$). The product-promoter affinity is calculated as a function of Euclidean distance between coordinates of the product and the promoter.

One of the properties of this type of GRN encoding is that there is no predefined limit on the number of types of transcription factors that can exist in the system. There is also no limit on the size of the genome and the number of nodes or connections between nodes in the regulatory network.

Concentrations of all products in the system are represented as real values restricted to the interval [0,1] and are updated continuously during the simulation of each cell, depending on the production rates calculated for each regulatory unit. Additionally, all TFs degrade exponentially over time. Special type of elements are used to mark inputs and outputs of the network. The outputs allow the cells to perform certain actions (such as division, apoptosis), the inputs provide them with external signals.

Results

We have used a genetic algorithm to evaluate the evolvability and computational capability of the presented model. The genetic algorithm included genetic operators that acted at the level of single genetic elements (by small changes to coordinates or changes to element type) or at the level of whole genomes, by duplicating or deleting whole series of genetic elements. Some experiments included recombination between genomes. The fitness function was specific for a particular problem.

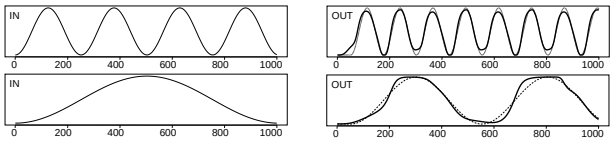

Figure 2: Training set of two I/O pairs representing change of concentration over time used to obtain networks doubling the input frequency, with the response of best evolved network overlaid (thick line).

Signal processing: Earlier works (e.g. Kuo et al., 2004; Knabe et al., 2006) have investigated regulatory network dynamics and their ability to generate desired behaviors such as externally driven oscillations. We have shown that GRNs that generate arbitrary oscillatory behaviors can be evolved also in our model and then attempted to evolve regulatory networks that perform more complex tasks, tasks that required some computation on the external signal. The signal was provided as externally driven concentrations of a selected TF. Similarly, a network's output was understood as a continuously changing concentration level of a specific TF.

To obtain networks with desired functionality, a set of training input/output pairs was prepared for each particular signal processing task. Each network from the population was simulated for a predefined number of time steps, with the concentration of input TF driven externally. The difference between obtained and desired output over time was used to calculate the error value. This process was repeated for each training pair and the fitness function was calculated as an average error over the training set. Fig. 2 demonstrates an example training set used to obtain networks that double the frequency of input oscillations. We have successfully obtained networks (Joachimczak and Wróbel, 2010b) that were required to filter oscillations by frequency, count pulses or memorize information. We observed a very good degree of generalization with networks properly responding to signals not present in the training sets.

To evaluate the possible benefits of continuous product accumulation/degradation, we have evolved networks with simplified node behavior. This behavior was equivalent to that of a perceptron-like neuron, i.e. the level of activation of each node would change instantly at each step. Higher fitness score of continuous GRN was observed for problem domain where gradual change of output was required.

Real time control: In another set of experiments (Joachimczak and Wróbel, 2010a), we have devised a 2D environment and a model of a unicellular animat equipped with two chemical sensors and two actuators, controlled directly by a GRN. We observed the evolution of search for beneficial substances and avoidance of poisonous substances in this environment. Our preliminary results show that control of chemotaxis can also be evolved in 3D.

Embryogenesis: We have created a model of 3D artificial morphogenesis in a continuous space (Joachimczak and Wróbel, 2009), with each cell driven by a copy of the same GRN and capable of division and movement. We have successfully evolved networks that can drive development to generate a desired 3D morphology and patterning (see Fig. 3) and demonstrated robustness to cellular damage of the developmental process.

Figure 3: Example obtained morphologies of multicellular embryos; left - fitness function rewarded stem-cap morphology, right - fitness function rewarded both ellipsoidal morphology and expression of correct color gene.

Summary

Our GRN model was found to be highly evolvable over various domains. The ability to automatically design GRNs that solve and generalize various control problems suggests that GRNs can be a useful computational model provided that they are applied to adequate control problems. Our results indicate that the introduction of the more realistic molecular dynamics in our or similar platform for the evolution of artificial GRNs might allow for the development of a useful tool for designing and optimizing synthetic biological regulatory networks.

Acknowledgments: Financial support: N519 384236. Computational resources: Tri-city Academic Computer Center (TASK) and ICM (University of Warsaw; G33-8).

References

Joachimczak, M. and Wróbel, B. (2009). Evolution of the morphology and patterning of artificial embryos: scaling the tricolour problem to the third dimension. In *Proceedings of the 10th European Conference on Artificial Life (ECAL2009)*, volume 5777 of *LNCS*, pages 33–41. Springer.

Joachimczak, M. and Wróbel, B. (2010a). Evolving gene regulatory networks for real time control of foraging behaviours. In *Artificial Life XII: Proceedings of the Twelfth International Conference on the Simulation and Synthesis of Living Systems*, pages 348–355. MIT Press, Cambridge, MA.

Joachimczak, M. and Wróbel, B. (2010b). Processing signals with evolving artificial gene regulatory networks. In *Artificial Life XII: Proceedings of the Twelfth International Conference on the Simulation and Synthesis of Living Systems*. MIT Press, Cambridge, MA.

Knabe, J. F., Nehaniv, C. L., Schilstra, M. J., and Quick, T. (2006). Evolving biological clocks using genetic regulatory networks. In *Artificial Life X: Proceedings of the Tenth International Conference on the Simulation and Synthesis of Living Systems*, pages 15–21. MIT Press/Bradford Books.

Kuo, D. P., Leier, A., and Banzhaf, W. (2004). Evolving dynamics in an artificial regulatory network model. In *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *LNCS*, pages 571–580. Springer Berlin / Heidelberg.