

# **Evolvability and robustness of in silico evolution of gene regulatory dynamics**

**Yaochu Jin, Yan Meng, Bernhard Sendhoff**

**2009**

**Preprint:**

This is an accepted article published in Foundations of Systems Biology in Engineering. The final authenticated version is available online at: [https://doi.org/\[DOI not available\]](https://doi.org/[DOI not available])

# EVOLVABILITY AND ROBUSTNESS OF *IN SILICO* EVOLUTION OF GENE REGULATORY DYNAMICS

Yaochu Jin <sup>1\*</sup>, Yan Meng <sup>2</sup> and Bernhard Sendhoff <sup>1</sup>

<sup>1</sup> Honda Research Institute Europe, 63073 Offenbach, Germany

<sup>2</sup> Stevens Institute of Technology, Hoboken, NJ 07030, USA

## Abstract

This paper investigates the influence of regulation logic, together with the way in which negative and positive feedback loops are coupled, on the easiness of evolving typical regulatory dynamics such as limit cycles in a computational evolutionary system. Our simulation results suggest that the evolvability of such regulatory motifs depends not only on the regulation logic that combines different regulatory factors, but also on the way how the feedback loops are coupled. Our results indicate that probabilistic ‘AND’ or summation logic facilitates the evolution of sustained oscillation for consistently regulated target genes. On the other hand, the probabilistic ‘OR’ and summation logic makes it easier to evolve sustained oscillation in case the target gene is regulated inconsistently. In addition, we find that when the target gene is consistently regulated, the sustainability of the oscillation is more vulnerable to perturbations in the initial state of the concentrations and the parameters of the regulatory motif.

## Keywords

Gene regulatory networks, Regulation logic, *in silico* evolution, Dynamics, Evolvability, Robustness

## Introduction

*In silico* evolution of typical dynamics of gene regulatory networks has attracted much research attention in systems biology. For instance, Francois and Hakim (2004) evolved both bi-stable and oscillatory dynamics using a number of predefined basic biochemical reactions. Follow-up research indicated, however, that it is a daunting task to evolve sustained oscillation. One possible reason is that the target function, also known as the *fitness function*, used in the evolution may not be effective for the computational evolutionary system to find the proper structure or parameters. In Francois and Hakim (2004), the difference between output of the regulatory model and the desired amplitude at the half-integer and integer periods has been adopted as the fitness function. To facilitate the evolution of sustained logic, Paladugu et al (2006) employed a fitness function based on the condition for generating Hopf bifurcation. Chu (2007), on the other hand, suggested a correlation based fitness function. Unfortunately, no definite conclusion can be drawn on

whether the suggested fitness functions have really facilitated the evolution of sustained oscillation.

Jin et al (2008) have investigated the easiness, evolvability in a loose sense, of evolving oscillatory dynamics from a different perspective. Instead of trying new fitness functions, they looked into the influence of the genetic encoding scheme as well as the activation function used in the gene regulatory model of a relaxation oscillation circuit. It has been shown that it is much easier to evolve sustained oscillation with a step function than with a Hill function. When a higher Hill is used, a larger co-efficient is preferred for the evolution of sustained oscillation. However, the evolutionary system is not able to find a large Hill co-efficient automatically.

Meanwhile, regulation control, particularly the regulation logic, is also attracting more and more research efforts. An experimental analysis of regulation control of the gene for development of the sea urchin has been conducted in Yuh et al (2000). A systematic investigation

---

\* To whom all correspondence should be addressed

of control logic in gene regulation has been performed in Schilstra and Nehaniv (2008), which concludes, among others, that networks consisting of competitively binding activators and repressors can be controlled more robustly.

This paper investigates *in silico* the role of regulation logic in evolving oscillatory dynamics on two regulatory motifs consisting of a negative feedback loop and a positive feedback loop. To this end, we employ an evolution strategy, one of the widely used artificial evolutionary algorithms (Bäck, 1996), to evolve the parameters of the given network motifs. Our results indicate that both regulation logic and the way how feedback loops are coupled play a substantial role in the evolvability of gene regulatory networks, and in the stability of the evolved dynamics as well.

### Regulatory Motifs and Regulation Logic

This work considers two small regulatory motifs, each consisting of a positive feedback loop and a negative feedback loop, as shown in Figure 1. The only difference between the two motifs lies in the way how the two feedback loops are coupled in the target gene,  $g_3$ . Hereafter, we term the motif on the left in Figure 1 *consistently regulated motif* (CRM), whilst the one on the right *inconsistently regulated motif* (IRM).

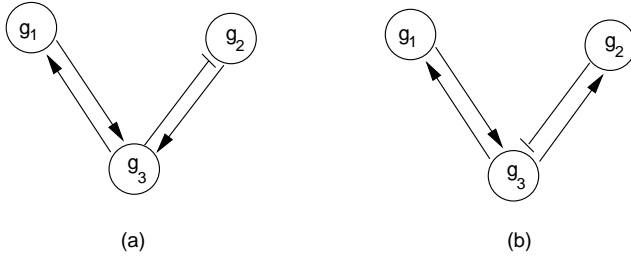


Figure 1. Two regulatory motifs. (a) Consistently regulated motif (CRM) (b) Inconsistently Regulated Motif (ICR)

The CRM can be described using the following ordinary differential equations (ODEs):

$$\dot{x}_1 = a_{13}H_{13}(x_3) - a_{11}x_1, \quad (1)$$

$$\dot{x}_2 = a_{23}H_{23}(x_3) - a_{22}x_2, \quad (2)$$

$$\dot{x}_3 = a_3L(H_{13}(x_1), H_{32}(x_2)) - a_{33}x_3, \quad (3)$$

where  $x_i$ 's are the concentration of the protein product of the genes,  $a_{11}$ ,  $a_{22}$  and  $a_{33}$  are the degradation rate of the proteins,  $a_{13}$ ,  $a_{23}$ , and  $a_3$  are the parameters representing the strength of gene-protein interactions. In the CRM, there are three activating regulations and one repressive regulation distinguished by different Hill

functions. An activating Hill function, e.g.,  $H_{13}$  can be described as follows:

$$H_{13}(x_3) = \frac{\beta x_3^n}{\theta_1^n + x_3^n}, \quad (4)$$

where  $n$  is called the Hill coefficient. By contrast, a repressive Hill function, e.g.,  $H_{23}$ , can be represented by

$$H_{23}(x_3) = \frac{\beta}{1 + (x_3 / \theta_2)^n}. \quad (5)$$

In Eq.(3),  $L(H_{13}(x_1), H_{32}(x_2))$  is a function describing the regulation logic combining the regulation of various transcription factors. In this work, we suppose that various transcription factors bind to the binding site of the target gene ( $g_3$ ) independently.

We assume that different transcription factors bind to the binding site of a gene independently. We consider two situations. First, both transcription factors are needed to activate the expression of the target gene. Second, either the product of gene 1 ( $g_1$ ) or that of gene 2 ( $g_2$ ) will be able to activate the expression of the target gene. For the former case, we use the fuzzy logic 'AND', which can be described using either the Zadeh 'AND' or the probabilistic 'AND' (Jin, 2003):

$$\text{Zadeh 'AND': } L(x, y) = \min(x, y), \quad (6)$$

$$\text{Prob. 'AND': } L(x, y) = xy. \quad (7)$$

Similarly, the following Zadeh 'OR' and probabilistic 'OR' logic forms are adopted in case that the product of either gene 1 or gene 2 is able to activate the expression of gene 3 independently:

$$\text{Zadeh 'OR': } L(x, y) = \max(x, y), \quad (8)$$

$$\text{Prob. 'OR': } L(x, y) = x + y - xy. \quad (9)$$

In addition, we also consider the case where a summation is used for the logic function, which can be seen as a combination of logic 'AND' and logic 'OR':

$$\text{summation: } L(x, y) = \frac{1}{2}(x + y), \quad (10)$$

### A Computational Evolutionary System

Since the late 1960s, a larger number of evolutionary algorithms have been developed in the field of artificial intelligence (Bäck, 1996). These algorithms are inspired by natural evolution, and have been widely used in solving scientific and engineering problems. An evolutionary

algorithm usually consists of a population of individuals, each coding a potential solution to the problem. Genetic operators such as gene duplication or mutation can be applied to the individuals in a parent population to generate an offspring population. Then, a subset of the offspring population will be selected based on individuals' *fitness* (quality) as the parent population of the next generation. This process repeats until a stop criterion is met. Figure 2 illustrates a generic evolutionary algorithm.

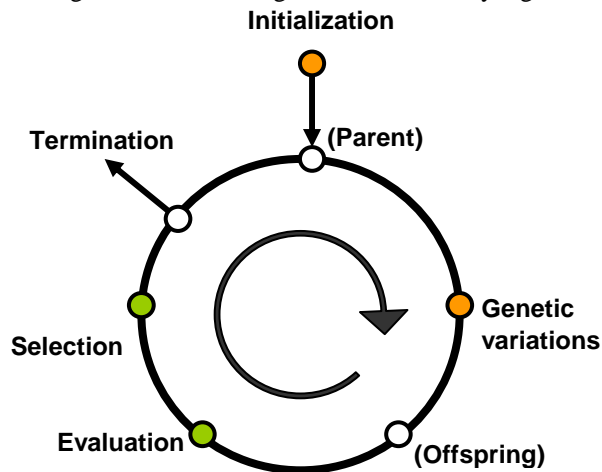


Figure 2. A generic evolutionary algorithm

In this work, we employ a canonical evolution strategy (ES),  $(\mu, \lambda)$ -ES, where  $\mu$  and  $\lambda$  are the parent and offspring population size, respectively, to evolve the parameters in the regulatory motifs. The goal of the evolution is to produce oscillatory dynamics for gene 3. In the evolution strategy, the parameters are mutated by adding a random number generated by a normal distribution, whose variance is also subject to evolution. The reader is referred to Bäck (1996) for a detailed description of the ES.

## Simulation Studies

### Experimental Setups

The differential equations are solved using the modified Euler's method with a step size of 0.1. The fitness function is as follows:

$$f = \sum_{t=0}^N (x_3(t) - x_3^d)^2, \quad (11)$$

where  $N$  is the number of time steps simulated in solving the dynamics,  $x_3^d$  is the desired output of the target gene defined by the following sinus function:

$$x_3^d = \sin(2\pi t / T) + 1.0, \quad (12)$$

where  $T$  is the desired period of the oscillation. In the simulations, all regulatory parameters are randomly initialized between 0 and 4, and the initial step-size of the ES is set to 1.0. The parent and offspring population sizes are set to 30 and 200, respectively. Each evolution is run for 500 generations.

### Easiness of Evolving Sustained Oscillations

To check the easiness of evolving sustained oscillation for the CRM and the IRM with different regulation logics, we attempt to evolve oscillatory dynamics for 12 different periods, that is,  $T=1,2,\dots,12$ . For each desired period, we performed 10 independent runs. Then we count the number of runs in which sustained oscillation is evolved successfully. The evolution is said to be successful if the amplitude change of the oscillation is less than 1% within 400 time steps. The results for the CRM and IRM are presented in Figure 3 and Figures 4, respectively.

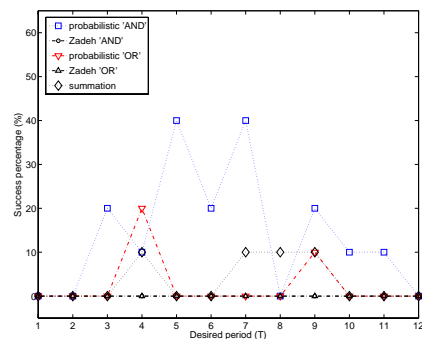


Figure 3. Success percentage of CRM

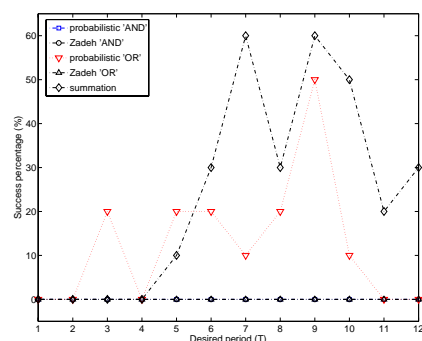


Figure 4. Success percentage of IRM

From Figures 3 and 4, we can see that the success percentage profiles for the CRM and IRM are quite different. CRMs with the probabilistic 'AND' logic and summation logic exhibit the highest and second highest success percentages on average. By contrast, IRMs with the summation and probabilistic 'OR' logic have the highest success percentage. As a whole, the success percentage for evolving sustained oscillation is quite low. Nevertheless, different to the relaxation motif studied in Jin et al (2008), a high Hill coefficient is not required for the CRM and IRM.

### Robustness of the Evolved Dynamics

In this section, we examine the stability of the evolved oscillatory dynamics when the initial concentrations vary or when the parameters of the regulatory motifs are perturbed.

We first change the initial concentrations of the three proteins after evolution is complete for testing the stability of the evolved dynamics. The initial states are set to (1,1,0) during the evolution. In the test, they are initialized randomly between 0 and 4. For the CRM with the probabilistic ‘AND’, sustained oscillation is evolved successfully in 16 runs out of 120 independent runs. When the initial states are re-initialized, some of the initial states converge to a point attractor instead of a limit cycle for all 16 evolved motifs. For CRM with the summation logic, sustained oscillation has evolved in four runs. In these cases, all random initial states converge to a limit cycle. From these results, we can conclude that the sustained oscillation of CRMs with the summation logic is less sensible to perturbations in the initial concentrations than the CRM with the probabilistic ‘AND’ logic. For the IRM, the results are quite different. For all the 29 success cases with the summation logic and the 15 cases with the probabilistic ‘OR’ logic, all randomly initialized states converge to the evolved limit cycle.

The sensitivity to perturbations in parameters shows also very different behaviors. It is found that for CRMs, with either the probabilistic logic or the summation logic, are vulnerable to noises in parameters. Figure 5 shows the 50 state-space trajectories of a CRM with the summation logic under 1% (left panel) and 5% (right panel) noise level. The 50 initial states are generated randomly in the interval of [0, 4]. For a level of 1% noise, the regulatory system is still able to produce sustained oscillation for all the initial states. However, when the noise level rises to 5%, the oscillation becomes damped, and finally converged to a point attractor.

The results from IRM are quite different. For IRMs with either the probabilistic ‘OR’ logic or the summation logic, the regulatory system is able to generate sustained oscillation under 1% or 5% noise level in the parameters. Fifty trajectories of an IRM with the summation logic are

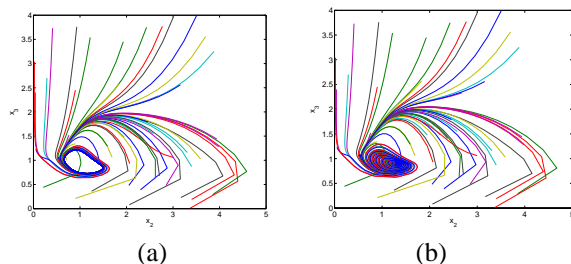


Figure 5. State-space trajectories of a CRM in presence of noise in the parameters, where the x-axis denotes  $x_2$  and the y-axis  $x_3$ . Left panel: with 1% noise. Right panel: 5%.

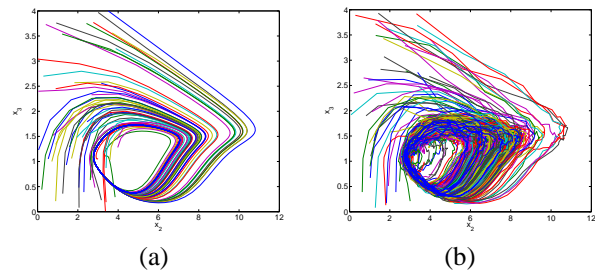


Figure 6. State-space trajectories of an IRM in presence of noise in the parameters, where the x-axis denotes  $x_2$  and the y-axis  $x_3$ . Left panel: with 1% noise. Right panel: 5%.

presented in Figure 6, where the left panel shows the results with 1% noise and the right panel the results with 5% noise. It can be seen that in both cases, the regulatory system is able to maintain a sustained oscillation.

### Conclusions

Our results suggest that both regulation logic and the way for coupling the feedback loops in regulatory network motifs play a central role in evolving sustained dynamics. We also find that the sustained oscillation of the evolved IRMs is more robust to perturbations in the parameters of the regulatory model than the CRM. Our future work will examine if the more evolvable and robust regulatory motifs will emerge in evolution where the regulation logic and the coupling structure are not predefined.

### References

- Bäck, T. (1996) *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press
- Chu, D. (2007) Evolving genetic regulatory networks for systems biology. *In Proceedings of IEEE Congress on Evolutionary Computation*, Singapore, 875-882
- Francois, F., Hakim V. (2004) Design of genetic networks with specified functions by evolution *in silico*. *Proc. Natl. Acad. Sci. USA*, 101, 2, 580-585.
- Jin, Y. (2003) *Advanced Fuzzy Systems Design and Applications*. Physica / Springer
- Jin, Y., Sendhoff, B. (2008) Evolving *in silico* bistable and oscillatory dynamics for gene regulatory network motifs. *In Proceedings of IEEE Congress on Evolutionary Computation*, Hong Kong, 386-391
- Paladugu, S., Chickarmane, V., Dekard, A., Frumkim, J., MaCormarck, M., Sauro, H. (2006). *In silico* evolution of functional modules in biochemical networks. *IEE Proceedings on Systems Biology*, 153, 4, 223-235
- Schilstra, M., Nihaniv, C. (2008) Bio-Logic: Gene expression and the laws of combinatorial logic. *Artificial Life*, 14, 121-133
- Yuh, C.-H., Bolouri, H., Bower, J., Davidson, E. (2000) A logical model of cis-regulatory control in a Eukaryotic system. *In Computational Modeling of Genetic and Biochemical Networks*, Bower, J., Bolouri, H. (Eds.). The MIT Press, 73-100