

Combining Genetic Oscillators and Switches using Evolutionary Algorithms

Spencer Thomas

Department of Computing, University of Surrey
Guildford, Surrey, GU2 7XH, United Kingdom
Email: s.thomas@surrey.ac.uk

Yaochu Jin

Department of Computing, University of Surrey
Guildford, Surrey, GU2 7XH, United Kingdom
Email: yaochu.jin@surrey.ac.uk

Abstract—It is hypothesised that complex biological gene regulatory networks can be evolved from simple networks through a process of modularisation, duplication and specialisation. However, the biological mechanisms of this process remain elusive and little work has been done to verify this hypothesis in a computational environment. This paper aims to couple two simple regulatory motifs, one toggle switch and one self-sustained oscillator using an evolutionary algorithm, which can be seen as a computational simulation of natural evolution. We have successfully evolved several complex dynamics for two different connections arrangements between the oscillator and toggle switch networks in a master/slave set up, which confirms the previously reported results achieved manually. Our results indicate that generating complex dynamics by coupling of simple motifs using simulated evolutionary mechanisms is methodologically feasible and more efficient, which can be seen as an indirect and partial verification of the above hypothesis.

I. INTRODUCTION

Biological systems can be modelled using biological networks and are often required to be robust to environmental perturbations [1]. In biological networks, a network motif is a repeating network pattern that appears in nature more often than in a random system [2], [3]. Gene regulatory networks (GRNs) are examples of biological networks, which are a group of genes that interact through their protein production. In this case the genes produce proteins which then effect the other genes' production of protein. These GRNs can be consistently regulated motifs (CRM) or inconsistently regulated motifs (IRM) depending on the interactions from the regulatory genes to the target gene. In a CRM, the incoming interactions are all activating (or all inhibitory), whereas in an IRM the incoming interactions can be either activating or inhibitory.

The auto- or self-regulation of a gene network allows self-activation or self-repression through its own protein production. Negative auto-regulation (NAR) results in a rapid response to an input signal, which is important for biological systems [2]. Of particular interest in biology are those network motifs that produce self-sustained oscillations and bistable toggle switch dynamics [4] such as signalling cascades, neural networks [5] and the irreversibly outcome of a cell, determined through transitions between stable points [2]. Analysing GRN motifs is an important research area in systems biology [6], [7], and understanding the relationship between the structure

and dynamics of GRNs are vital in the understanding of natural evolution [4].

It is hypothesised that GRNs evolve from simple structures to more complex networks via gene duplication [8], though the mechanisms for this process are still unclear [9]. Gene duplication leads to functional redundancies in the absence of mutation [8], as a stable genome is unlikely to maintain identical genes [10]. As a result, identical genes often diverge in natural evolution [8].

Generating motifs with multi-stability, such as in a toggle switch, requires positive feedback loops [5], [11], whereas a negative feedback circuit is required to produce oscillations in biochemical systems [5]. Biological systems also use negative feedback loops to improve robustness [12] and often consist of interconnected positive and negative feedback circuits [13]. Thus the combination of two simple motifs, particularly the toggle switch and the oscillator are of great interest in systems biology.

The method used in this work is based on the ideas presented in [5], where an oscillator and a toggle switch are coupled and the connection parameters between the two motifs are manually tuned. In this work, by using an evolutionary algorithm we attempt to more realistically simulate natural evolution of network motifs from simple to more complex networks. This method allows us to optimise multiple parameter values to best produce our desired state in a similar fashion to evolution in nature. Here we use a variant of evolution strategy as these have been shown to be successful in solving continuous parameter optimisation problems [14]. This also increases the flexibility of the calculation by avoid manually specifying the values of the control parameters, which has previously been noted as difficult [5].

This paper is organised as follows. In Section II, an introduction to the gene regulatory networks investigated here, the expressions used to model the gene interactions and the dynamics produced. The evolutionary method used here is described in Section III. How the network motifs are combined is described in Section IV. *In silico* results and discussions are in Section V and Section VI concludes this paper.

II. GENE REGULATORY NETWORKS

A. Self-sustained Oscillations

GRN can be modelled using Hill functions for the interactions between the genes in the network. Producing self-sustained oscillatory dynamics for a GRN in simulated evolution is a non-trivial task and is limited in the possible periods of oscillation [4]. In addition, oscillatory dynamics are sensitive to the initial conditions of the system and it is believed that the experimental setup can also effect the outcome [4], [15]. The GRN used to produce a self-sustained oscillation is given by

$$\dot{x}_1 = a_{12}H_{12}(x_2) - a_{11}x_1 , \quad (1)$$

$$\dot{x}_2 = a_{23}L(H_{23}(x_3), H_{21}(x_1)) - a_{22}x_2 , \quad (2)$$

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

where \dot{x}_n is the time derivative of x_n , x_2 is the target gene, a_{nm} are regulatory parameter, $H_{nm}(x_m)$ is the Hill function and $L(H_{23}(x_3), H_{21}(x_1))$ represents the logic function which combines the interactions of two regulatory genes to the target gene. Here we consider summation and probabilistic ‘OR’ logic functions given by,

$$L(x, y) = \begin{cases} (x + y)/2 & \text{Summation} \\ x + y - xy & \text{Probabilistic ‘OR’} . \end{cases} \quad (4)$$

The type of interaction defined by the Hill function can be activating or repressive and the logic functions are used when a gene has multiple interaction inputs from other genes. The target gene x_2 in Fig. 1(a) is consistently regulated by both genes x_1 and x_3 , whereas Fig. 1(b) x_2 is inconsistently regulated by x_1 and x_3 . The logic function can combine these interactions and can for example take the form of AND or OR gates [1].

A three-gene network is used to produce a self-sustained oscillation which results in the formation of a limit cycle as shown in Fig. 2(a). This limit cycle shown is produced using a CRM shown in Fig. 1(a), however IRM, Fig. 1(b), can also produce limit cycles in the phase plane. Both CRM and IRM are used in this work.

B. Toggle Switch

Compared to the sustained oscillation, a toggle switch is easy to evolve in a computational system. A two-gene network used to evolve a toggle switch is illustrated in Fig. 1(c), which leads to a bi-stability in the phase plane in Fig. 2(b). The dynamics for this GRN are given by

$$\dot{x}_4 = b_{12}H_{45}(x_4) + b_{11}x_4 , \quad (5)$$

$$\dot{x}_5 = b_{21}x_4 + b_{22}x_5 , \quad (6)$$

where x_5 is the target gene and b_{nm} are regulatory parameters.

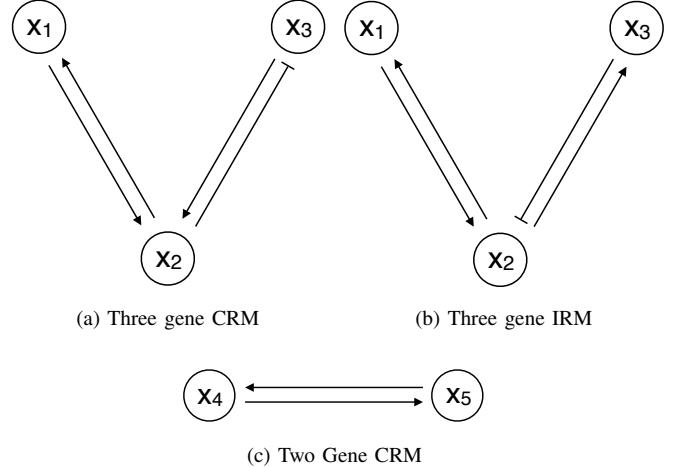


Fig. 1: Networks for (a) a consistently regulated motif and (b) an inconsistently regulated motif, both with three genes, and (c) a consistently regulated motif with two genes. Here arrows and lines with bar ends represent activating and repressor interactions between the genes respectively. All the genes here also have negative auto-regulations (not shown).

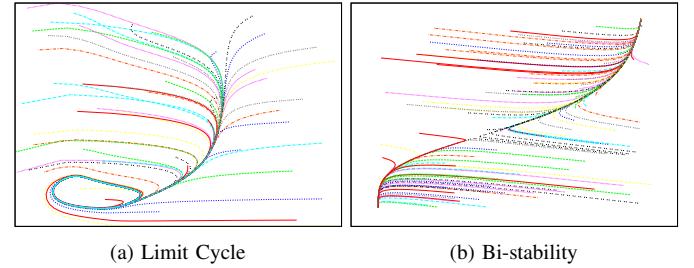


Fig. 2: Network dynamics showing the formation of (a) a limit cycle produced by Fig. 1(a) and (b) a bi-stability produced by Fig. 1(c).

III. EVOLUTION STRATEGY

The gene interactions are modelled by ordinary differential equations, (1-6), which are solved using the Euler method. The solutions are then optimised using an evolution strategy [14]. Evolution strategies belong to a class of evolutionary algorithms that have turned out to be very efficient for continuous optimisation. Since only the connecting parameters are evolved in this work, we adopt a canonical evolution strategy implemented in the EALib C++ library, which is part of the Shark library [16]. In a canonical evolution strategy (ES), the mutation of the objective parameters is performed by adding an $N(0, \sigma_i^2)$ distributed random number. The step-sizes σ_i are also encoded in the genotype and subject to mutations. The ES used in this work can be described as follows:

$$\sigma_i(t) = \sigma_i(t-1)\exp(\tau' z)\exp(\tau z_i), \quad (7)$$

$$\mathbf{x}(t) = \mathbf{x}(t-1) + \tilde{\mathbf{z}}; i = 1, \dots, n, \quad (8)$$

where \mathbf{x} is an n -dimensional parameter vector to be evolved, $\tilde{\mathbf{z}}$ is an n -dimensional random number vector with $\tilde{\mathbf{z}} \sim N(\mathbf{0}, \sigma(t)^2)$, z and z_i are normally distributed random numbers with $z, z_i \sim N(0, 1)$. Parameters τ , τ' and σ_i are called strategy parameters, where σ_i is mutated as in Equation (7) and τ , τ' are constants as follows:

$$\tau = \left(\sqrt{2\sqrt{n}} \right)^{-1}; \quad \tau' = \left(\sqrt{2n} \right)^{-1} \quad (9)$$

Two selection schemes have been proposed in evolution strategies, known as comma and plus strategies. Suppose there are μ and λ individuals in the parent and offspring population, usually $\mu \leq \lambda$. In the comma strategy, μ parent individuals are selected only from the λ offspring individuals, which is usually noted as (μ, λ) -ES. In the plus strategy, μ parent individuals are selected from a combination of μ parent individuals and λ offspring individuals, which is noted as $(\mu + \lambda)$ -ES.

$(\mu + \lambda)$ -ES is an elitism, which is often not recommended for continuous optimisation problems, as it can more easily lead to local minima rather than good global solutions [17]. However, we have found that elitism aides the convergence of the evolutionary search as it maintains the best solutions after each generation. This was also observed in [18].

We use a mean squared error between the desired concentration and the real concentration of the target gene as a measure of fitness, which is defined as,

$$f = \sum_{t=0}^N \left(x_{tg}^i(t) - x_{tg}^d(t) \right)^2, \quad (10)$$

where N is the number of time steps, $x_{tg}^i(t)$ is the state of the target gene (tg) for the i th generation, and $x_{tg}^d(t)$ is the desired protein concentration of the target gene.

All the parameter values in Equations (1-6) are randomly initialised and represented as a chromosome for an individual solution for the differential equations. There are a population of individuals representing many possible solutions for the equations. These parent populations are used to produce a population of offspring through random parent assignment for each individual offspring using recombination and mutation operators. The algorithm retains the parameter values that produce the better (smaller) fitness, f , value during each generation for each network. After each successive generation the solutions are ranked in order of best (lowest) fitness to worst (highest) fitness. The best solutions, i.e. the ones that produce solutions with the best fitness values, are retained and the others erased. This process is repeated until the number of generations has exhausted.

The desired state of x_2 , the target gene as in Fig. 1(a), that is used to produce a self-sustained oscillation has a sinusoidal structure,

$$x_2^d(t) = \sin(2\pi t/T) + 1.0, \quad (11)$$

where T is the desired period of the oscillator.

The desired state for evolving a toggle switch is dependant on the initial protein concentration of the target gene x_5 , see

Fig. 1(c), and is given by

$$x_5^d(t) = \begin{cases} 1 & \text{if } x_{5i} > 0.5 \\ 0 & \text{if } x_{5i} \leq 0.5 \end{cases} \quad (12)$$

IV. COMBINATION OF NETWORK MOTIFS

The connection between the network motifs can be set up in two different master/slave arrangements. Firstly by having the oscillator network, Fig. 1(a), controlling the toggle switch system, Fig. 1(c), as shown in Fig. 3(a). In the other arrangement, the toggle switch controls the oscillator, as shown in Fig. 3(b). The connection between the two motifs is a simple linear function, as the exact form of the coupling has been demonstrated not to be important, however the control parameters between motifs need to be appropriately tuned [5]. Also in [5] it was stated that it would be difficult to control these parameters, making the evolutionary method favourable in terms of its simplicity by allowing the algorithm to optimise these values based on a fitness function. The connection function is given by

$$M_{ij} = c_0 x_j + c_1, \quad (13)$$

where M_{ij} is the connection from gene x_j to gene x_i , and c_n represents the connection parameters. M_{ij} is then added to the equation representing \dot{x}_i .

For the case shown in Fig. 3(a) the equation for the time derivative becomes

$$\dot{x}_4 = b_{12} H_{45}(x_4) + b_{11} x_4 + M_{42}, \quad (14)$$

where $M_{42} = c_0 x_2 + c_1$.

To ensure we are combining a self-sustained oscillation and a bi-stability, the two network motifs are evolved separately. Once we have evolved both dynamics successfully we retain the parameter values for a_{nm} and b_{nm} that produce them and evolve only the connection parameters between the two motifs, i.e. c_0 and c_1 . Using this form of connection the master gene directly effects the slave gene, while the original dynamics of the master gene remain unchanged.

V. RESULTS

The parameters used for all evolutionary runs, including those for evolving the toggle switch and oscillator, as well as for coupling the two motifs, are given in Table I.

Here all motifs are optimised over 100 generations as this is sufficient to produce the oscillator and switch dynamics shown in Fig. 2. Our objective is to combine these two dynamics, therefore, once they are obtained further optimisation though successive generations is not required. We use population sizes given in Table I as there are sufficient to provide a diverse pool of candidate solutions without being computationally exhaustive. These population sizes have been shown to be successful in evolving oscillations and toggle switches when used in conjunction with an initial σ step size of 0.1 [4]. The parameter values are also used in the connection evolution to produce the more complex dynamics from the networks shown in Fig. 3.

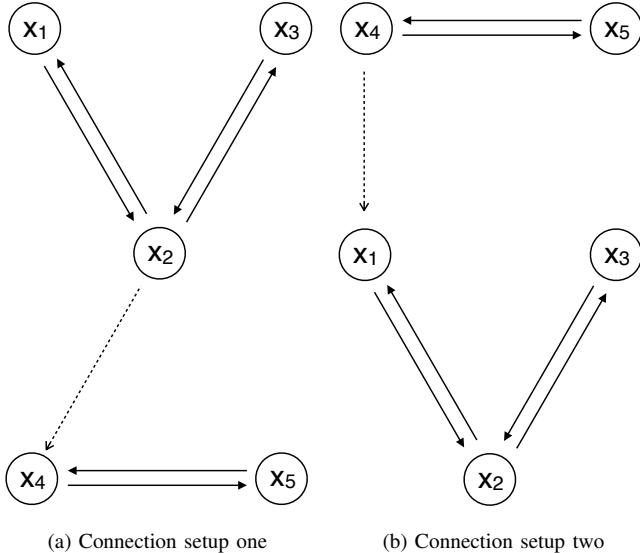


Fig. 3: Connection between the two network motifs; (a) oscillator controlling the toggle switch and (b) Toggle switch controlling the oscillator. Here the solid arrows represent with activate or repressive connections between genes and the dashed lines represent the connection between the network motifs. Note the simple linear connection is a one way interaction. As in Fig. 1 all genes have negative auto-regulation (not shown).

TABLE I: A list of parameters used in the experiments

Population size	
μ (Parent)	50
λ (Offspring)	300
Probability	
σ_{Cross} (Crossover)	0.9
σ_{Flip} (Bit flip mutation)	0.02
$\sigma_{Gaussian}$ (Gaussian mutation)	1
Other	
h_{Euler} (Step size in Euler solver)	0.1
Number of generations	100
Initial step size of ES (σ)	0.1

A. Oscillator Controlling the Toggle Switch

The coupling set up shown in Fig. 3(a) when using a CRM in conjunction with summation logic it is possible to produce a bi-rhythmicity. This is an observed phenomenon, in which the coupling to the oscillator results in an oscillation around each of the stable points of the toggle switch. When viewed in the phase plane, this results in limit cycles forming around the stable points of the bi-stability, which is shown in Fig. 4. These dynamics are produced by using the fitness function defined in Equation (10) but with a desired state of

$$x_d = 2 \sin(2\pi t/T) + 1 . \quad (15)$$

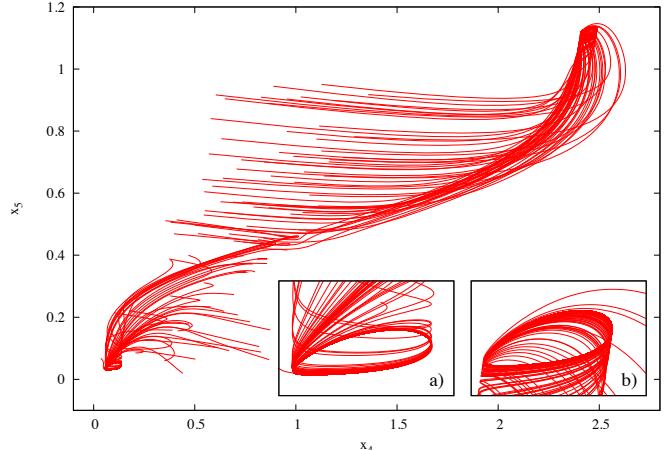


Fig. 4: A bi-rhythmicity produced using a CRM with summation logic. The insets here show the formation of limit cycles around a) the lower, and b) the upper stable points of the bi-stability.

In [19] multiple oscillatory domains, which are functions of the control parameters, are said to be closely related to the appearance of bi-rhythmicity. It is also stated that the control parameter can be perturbed, causing the system to go from one steady state to either of the bi-rhythmic stable point limit cycles, which differ in period and amplitude. A similar property characterises thalamic neurons, which are capable of having different oscillation periods based on the state of the membrane [20], [21], [22]. Phase plane analysis of these neurons reveal multiple excitability thresholds [19] giving rise to the bi-rhythmicity. The different levels of attraction to the stable limit cycles result from the different sensitivities to perturbations in the oscillatory regimes [19]. This gives rise to a larger attraction region for the upper limit cycle compared with the lower limit cycle shown in Fig. 4. It has been demonstrated that under the right conditions, the system can reversibly switch from one limit cycle to the other [19].

By using an IRM with summation logic it is possible to produce a limit cycle around the upper stable state of the toggle switch only. This is shown in Fig. 5 and produced using

$$x_d = 5 \sin(2\pi t/T) + 1 , \quad (16)$$

as the desired state of the connected gene in the slave network, see Fig. 3.

Using this set up with probabilistic ‘OR’ logic, the toggle switch stable states split leading to a doublet of stable states. When looking at the phase plane this leads to two bi-stability lines. This is shown in Fig. 6. These dynamics were produced using the same desired state and fitness function that produced the bi-rhythmicity. Here however, a different random number was used in running the code and therefore produced different dynamics.

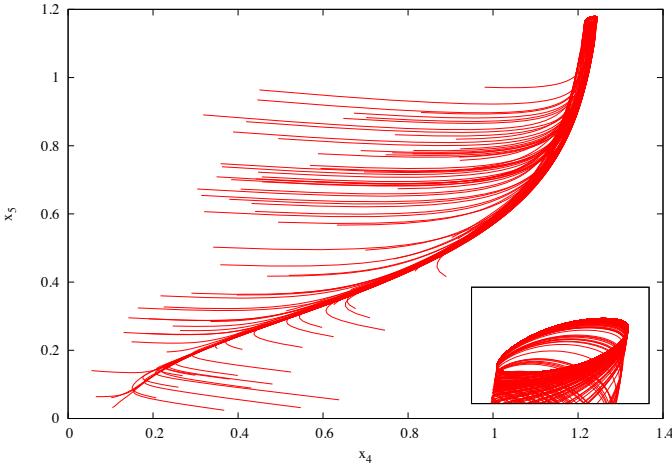


Fig. 5: Formation of a limit cycle around the upper stable point in the bi-stability as a result of the oscillator coupling to the toggle switch. The inset shows clearly the limit cycle around the upper stable point.

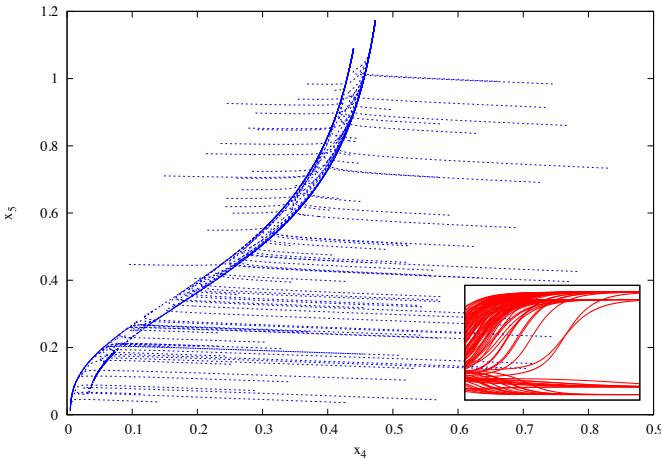


Fig. 6: The coupling of the oscillator and the toggle switch leads to a splitting of the stable points and the formation of a doublet bi-stability. The inset shows the splitting of the stable points in the toggle switch.

B. Toggle Switch Controlling The Oscillator

Coupling the toggle switch with the oscillator in the arrangement shown in Fig. 3(b) with summation logic and using a CRM for the three gene network enables the stable point to determine the outcome of the oscillation. The upper stable point leads to an equilibrium point as the result of a damped oscillation, whereas the lower stable point leads to a self-sustained oscillation, refer to Fig. 7. Here the coupling with the toggle switch enables the initial value of the toggle switch, and therefore the stable point from Equation (6), to determine whether the oscillation is self-sustained or damped. The desired state in the fitness function used to control whether

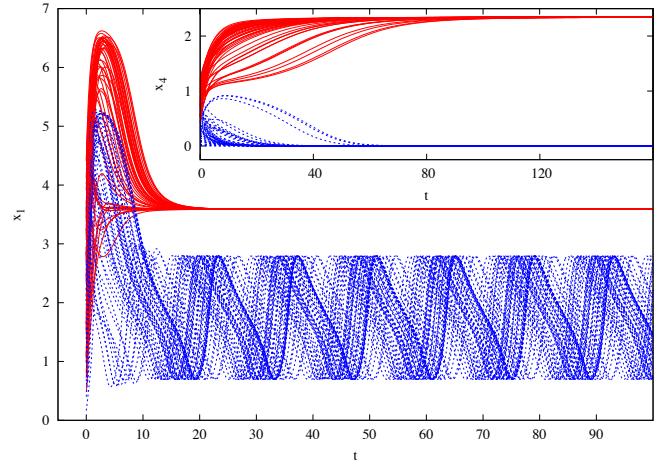


Fig. 7: Toggle switch controlling whether an oscillator evolved to be a self-sustained or a damped oscillation. The value of the stable point, and therefore the initial state of the toggle switch, determines whether or not a self-sustained oscillation is evolved from the coupled motifs.

the oscillation is self-sustained or damped is given in by,

$$x_d = 2 \sin(2\pi t/T) . \quad (17)$$

The same network setup that produces the dynamics shown in Fig. 7 can also produce a permanent change in the oscillation by a simple change to the desired state, given in Equation (18). Instead of determining whether or not a self-sustained oscillation is evolved, this configuration determines the amplitude and period of the oscillation shown in Fig. 8.

$$x_d = \begin{cases} 2 \sin(2\pi t/T) & \text{if } x_{4i} < 0.5 \\ \sin(2\pi t/T) & \text{if } x_{4i} \geq 0.5 \end{cases} \quad (18)$$

VI. CONCLUSION AND FUTURE WORK

Using a simple connection function we are able to successfully combine the dynamics of two gene regulatory networks to produce more complex dynamics. The use of an evolutionary algorithm provides solutions to the interaction equations by defining a fitness function and desired state. We were successful in producing many of the results found in previous works [5], including all observed dynamics for the toggle switch controlling the oscillator arrangement. We also observed the additional dynamics of a bi-stable doublet, as shown in Fig. 6. However, we have been unsuccessful so far in producing bi-stable switching and the formation of a limit cycle around the lower stable state in the oscillator controlling the toggle switch arrangement. Further investigation into different desired states could potentially lead to a periodic switch being observed with the setup demonstrated here. The evolution of a self-sustained oscillation has been noted in Section II-A to be difficult and small changes in the desired state have large affects on the dynamics demonstrated in Section V. Further

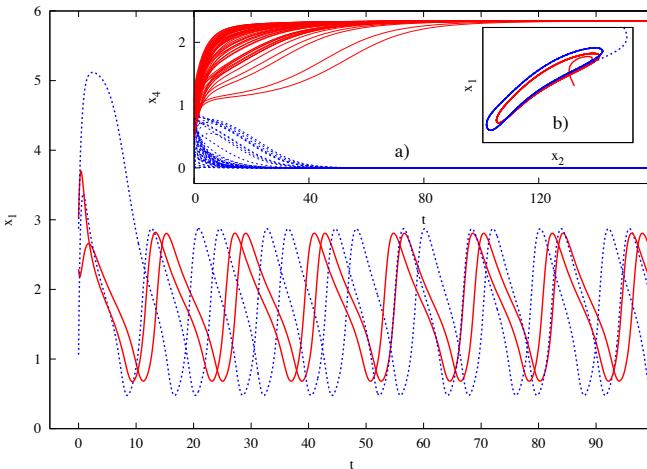


Fig. 8: The coupling to the toggle switch provides a control on the oscillations amplitude and period. Here only two oscillations coupled to each stable point are shown for the sake of clarity. It is clear that if the toggle switch evolves to the lower stable point (blue dashed line) the coupling to oscillator produces a significantly different self-sustained oscillation than if the toggle switch evolves to the upper stable point (red solid line). The oscillator lines correspond to the toggle switch stable points, solid red line and dashed blue line for the upper and lower stable points respectively. The toggle switch, inset a), for 100 test runs, showing the different line types for the upper and lower stable points. The phase plane of these different oscillations is shown in b), here only one of each oscillation is shown for clarity.

investigation into the fitness function, as well as the connection arrangement and network selection, CRM or IRM, may lead to other network dynamics including those previously observed but not reproduced here.

This method of combining the dynamics of small network motifs can be used to build larger networks which can produce complex dynamics. We have provided a simple yet flexible method that requires only a choice of desired state and definition of fitness function to produce the combined network dynamics that compare well to previous manual methods that require a large amount of user interaction. Our results also represent a preliminary step toward creating complex regulatory networks in artificial evolutionary systems using modularisation, duplication and specialisation.

ACKNOWLEDGMENT

The authors would like to thank the University of Surrey's Doctoral Training Centre and EPSRC for funding this work.

REFERENCES

- [1] U. Alon, "Biological networks: The tinkerer as an engineer," *Science*, vol. 301, no. 5641, pp. 1866–1867, 2003. [Online]. Available: <http://www.sciencemag.org/content/301/5641/1866.abstract>
- [2] C. Li, L. Chen, and K. Aihara, "A systems biology perspective on signal processing in genetic network motifs [life sciences]," *Signal Processing Magazine, IEEE*, vol. 24, no. 2, pp. 136 –147, march 2007.
- [3] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: Simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, 2002. [Online]. Available: <http://www.sciencemag.org/content/298/5594/824.abstract>
- [4] Y. Jin and B. Sendhoff, "Evolving in silico bistable and oscillatory dynamics for gene regulatory network motifs," in *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, June 2008, pp. 386 –391.
- [5] D. Gonze, "Coupling oscillations and switches in genetic networks," *Biosystems*, vol. 99, no. 1, pp. 60 – 69, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S030326470900149X>
- [6] U. Alon, *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman & Hall/CRC, 2006.
- [7] —, "Network motifs: theory and experimental approaches," *Nat. Rev. Genet.*, vol. 8, pp. 450–461, 2007.
- [8] J. Zhang, "Evolution by gene duplication: an update," *Trends in Ecology & Evolution*, vol. 18, no. 6, pp. 292 – 298, 2003. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0169534703000338>
- [9] R. V. Samonte and E. E. Eichler, "Segmental duplications and the evolution of the primate genome," *Nat Rev Genet*, vol. 3, no. 1, pp. 65–72, 2002.
- [10] M. A. Nowak, M. C. Boerlijst, J. Cooke, and J. M. Smith, "Evolution of genetic redundancy," *Nature*, vol. 388, pp. 167–171, 1997.
- [11] A. Crumire and M. Sablik, "Positive circuits and d-dimensional spatial differentiation: Application to the formation of sense organs in drosophila," *Biosystems*, vol. 94, no. 1-2, pp. 102 – 108, 2008. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0303264708001329>
- [12] Y. Jin, Y. Meng, and B. Sendhoff, "Influence of regulation logic on the easiness of evolving sustained oscillation for gene regulatory networks," in *Artificial Life, 2009. ALife '09. IEEE Symposium on*, april 2009, pp. 61 –68.
- [13] T. Y.-C. Tsai, Y. S. Choi, W. Ma, J. R. Pomerening, C. Tang, and J. E. Ferrell, "Robust, tunable biological oscillations from interlinked positive and negative feedback loops," *Science*, vol. 321, no. 5885, pp. 126–129, 2008. [Online]. Available: <http://www.sciencemag.org/content/321/5885/126.abstract>
- [14] H.-P. Schwefel, *Evolution and Optimum Seeking*. New York: Wiley, 1995.
- [15] D. Chu, "Evolving genetic regulatory networks for systems biology," in *Evolutionary Computation, IEEE Congress on*, sept. 2007, pp. 875 –882.
- [16] M. Kreutz, B. Sendhoff, and C. Igel, "EALib: A C++ class library for evolutionary algorithms," 6 2008-March 6 2008. [Online]. Available: <http://www.neuroinformatik.rhr-uni-bochum.de/in/PEOPLE/kreutz/top.html>
- [17] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies – a comprehensive introduction," *Natural Computing*, vol. 1, pp. 3–52, 2002, 10.1023/A:1015059928466. [Online]. Available: <http://dx.doi.org/10.1023/A:1015059928466>
- [18] Y. Jin and Y. Meng, "Emergence of robust regulatory motifs from in silico evolution of sustained oscillation," *Biosystems*, vol. 103, no. 1, pp. 38 – 44, 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S030326471001693>
- [19] A. Goldbeter, *Biochemical Oscillations and Cellular Rhythms*. Cambridge University Press, April 1997, vol. 1.
- [20] H. Jahnsen and R. R. Llins, "Electrophysiological properties of guinea-pig thalamic neurones: an in vitro study." *The Journal of Physiology*, vol. 349, no. 1, pp. 205–226, 1984. [Online]. Available: <http://jp.physoc.org/content/349/1/205.abstract>
- [21] —, "Ionic basis for the electro-responsiveness and oscillatory properties of guinea-pig thalamic neurones in vitro." *The Journal of Physiology*, vol. 349, no. 1, pp. 227–247, 1984. [Online]. Available: <http://jp.physoc.org/content/349/1/227.abstract>
- [22] R. R. Llins, "The intrinsic electrophysiological properties of mammalian neurons: insights into central nervous system function," *Science*, vol. 242, no. 4886, pp. 1654–1664, 1988. [Online]. Available: <http://www.sciencemag.org/content/242/4886/1654.abstract>