Learning Functions in Large Networks requires Modularity and produces Multi-Agent Dynamics

C.-H. Huck Yang*   CHAO-HAN.YANG@KAUST.EDU.SA

Living Systems Laboratory, BESE, CEMSE, King Abdullah University of Science and Technology, Thuwal, KSA*
Georgia Institute of Technology, North Ave NW, Atlanta, USA.

Rise Ooi*   RISE@EIS.HOKUDAI.AC.JP

Hokkaido University, Sapporo, Hokkaido, Japan.

Tom Hiscock   TWH27@CAM.AC.UK

University of Cambridge, UK

Victor Eguiluz   VICTOR@IFISC.UIB-CSIC.ES

IFISC (CSIC-UIB), Spain

Jesper Tegner   JESPER.TEGNER@KAUST.EDU.SA

Living Systems Laboratory, BESE, CEMSE, King Abdullah University of Science and Technology, Thuwal, KSA

Abstract

Networks are abundant in biological systems. Small sized over-represented network motifs have been discovered, and it has been suggested that these constitute functional building blocks. We ask whether larger dynamical network motifs exist in biological networks, thus contributing to the higher-order organization of a network. To end this, we introduce a gradient descent machine learning (ML) approach and genetic algorithms to learn larger functional motifs in contrast to an (unfeasible) exhaustive search. We use the French Flag (FF) and Switch functional motif as case studies motivated from biology. While our algorithm successfully learns large functional motifs, we identify a threshold size of approximately 20 nodes beyond which learning breaks down. Therefore we investigate the stability of the motifs. We find that the size of the real negative eigenvalues of the Jacobian decreases with increasing system size, thus conferring instability. Finally, without imposing learning an input-output for all the components of the network, we observe that unconstrained middle components of the network still learn the desired function, a form of homogeneous team learning. We conclude that the size limitation of learnability, most likely due to stability constraints, impose a definite requirement for modularity in networked systems while enabling team learning within unconstrained parts of the module. Thus, the observation that community structures and modularity are abundant in biological networks could be accounted for by a computational compositional network structure.

1. Introduction

The structure of networks has been investigated extensively using global metrics such as distributions of links and nodes, and local structures such as network motifs and graphlets. Learning algorithms, on the other hand, produce for example large deep layered structures with powerful pattern detection capabilities. In contrast, designed engineered systems are as a rule built from dynamical computational motifs or gates, producing a specific input-output function. However, due to combinatorial complexity only up to small three node functional motifs have been investigated. Network motifs (Milo et al., 2002) are the local features which are statistically over represented in networks. Investigators have designed computa-

tional searching methods to detect of over-represented motifs (Cussat-Blanc et al., 2011). However, statistical approaches include Monte Carlo methods and Bayesian models (Xing & Karp, 2004) are still computationally exhaustive and rely on existing quantitative data. GeneNet (Hiscock, 2017) is a developed neural network algorithm for gene circuit design without requiring prior data, which reinforce random initial networks to learn desired output functions by computing arbitrary functions (e.g., detect vibration of specified duration) and manipulating motifs behaviors (e.g., count pulses) in short computing time (e.g., 28.5 sec.) The previous results from GeneNet has shown its relevance on existing FF and Switch motifs studies (e.g., the classical FF motif from (Cotterell & Sharpe, 2010)) In this paper, we further investigate the connectivity and stability of GeneNet on high-ordered network components, as from 3×3 to 18×18 matrix. Through the regularization, genetic searching and stability mapping, we verified the reproducibility of this method as generative models on learnable networks as a functional multi-agent learning process. In contrast to exhaustive search, our approach is not limited to the identification of only small network motifs but more on the stability analysis with the high-ordered functional bio-motifs.

Functional gene circuit such as FF (Cotterell & Sharpe, 2010) and Switch (Gardner et al., 2000) have been further investigated on the modeling to analysis on *in vitro*. The previous (Joachimczak & Wróbel, 2009; Chavoya & Duthen, 2008) works on functional gene circuit design have been widely explored the FF model on three dimensional by physical simulation and genetic algorithm on a dynamic system. However, those approaches demand larger computing time and barely cover the feasibility of algorithm-based high-ordered gene circuit design.

Higher-ordered structures in networks (Lu et al., 2009) have become increasingly important since its applications and potential on understanding dynamically probing biological systems (Mutalik et al., 2013). Engineering of artificial gene networks (Ellis et al., 2009) has used control engineering principles to guide predictable gene network construction *in silico*. However, these engineered transcriptions (Khalil & Collins, 2010) factors have not yet been fully characterized, and if they are to be used as building blocks for complex gene networks. Instead of knowledge-based exclusive searching models, recent progress on ML-based techniques provides available software, etc. PyTorch and Tensorflow, tools to develop learning-based algorithms (Bengio et al., 2013) to describe and analyze the biosystem (Grover & Leskovec, 2016) in a high-ordered
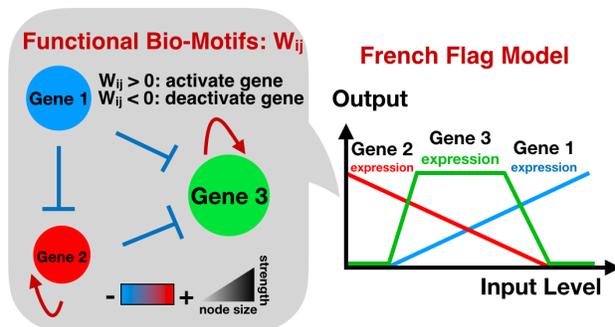
Figure 1. An example of a gene-circuit motifs which computes the French-Flag input-output function. The interactions between the units define the adjacency matrix.

network. learning-based method for high-ordered bionetwork would serve to investigate deeper on the *in vivo* kinetics and input-output transfer functions for synthetic biology (Wu et al., 2016).

Contributions.

- Based on existing ODE-based motifs learning method, we further expand the ML-based gene circuit design in a large network and propose a modified version of GeneNet, ES-GeneNet by evolutionary searching.

- We conduct adaptation behaviors on both homogeneous and heterogeneous team-learning among different sizes of a network.

- Finally, we find out the two specific network behaviors on size effects and evolving dynamics by the addressing stability analysis of multi-agent learning in bio-system.

2. Methods

2.1. Model of Gene Circuits

Dynamic gene network models (Reil, 1999) describe the time-dependent evolution of gene expression. Each node in the network receives input from the other nodes (De Jong, 2002). We follow the formulation of the gene network in (Hiscock, 2017; Cotterell & Sharpe, 2010):

$$\frac{dy_i}{dt} = \phi(\sum_{i,j} W_{ij} y_j) + I_i - y_i \quad (1)$$

where the y_i denotes the concentrations of ith gene, where $i = 1...N$ for an N-node network. W_{ij} is a matrix, $W \in R^{n \times n}$, correspond to network weight:
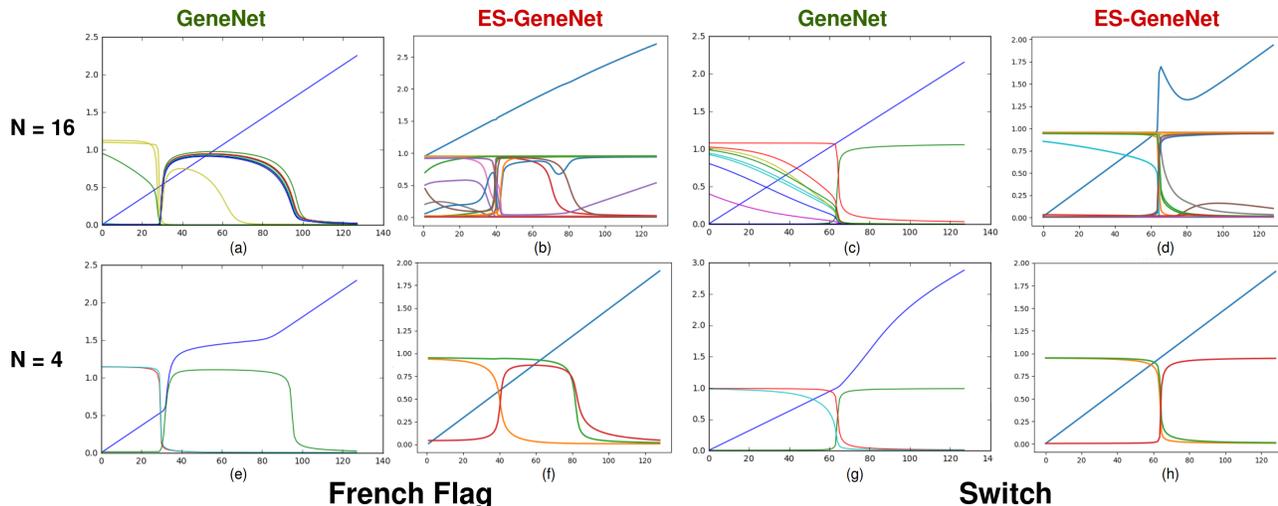
Figure 2. Learning larger network motifs. Both gradient descent (GeneNet) and genetic algorithms (ES-GeneNet) can learn functional network motifs larger than 3 nodes as discovered previously. Two examples with 4 and 16 nodes are illustrated for the cases of FF and the switch function. At a larger size of 16, the learnable gene circuits perform for FF on (a) and (b) and Switch on (c) and (d). For the smaller size of 4, the learnable gene circuits perform for FF on (e) and (f) and Switch on (g) and (h).

$W_{ij} > 0$ means that i-th activate the j-th gene, and $W_{ij} < 0$ means that i-th deactivate the j-th gene. For manipulating desired input and the output function, $y_i = F(x)$, where we define $y_1 \equiv x$ and $y_n \equiv y$ (for output at the final node.) I_i is the prescribed impulse, we give to the gene-network system. The $\phi(x)$ is a nonlinear function, ensuring that transcription rate to be activated. We choose the sigmoid function as $\phi(x)$ for activating neuron computing unit as

$$\phi(x) = \frac{1}{1 + e^{-x}} \qquad (2)$$

Genes expression and logic components could describe the initial condition (weight) of the first node and final output of the last node. Previous work (Hiscock, 2017) of GeneNet focused on 3-nodes system, which automatically learn the desired function to design a proper network computing the prescribed input-output function. We use this approach to train models for the French-Flag (FF) (Miller, 2004) and a switch function.

2.2. Learning parameters using Gradient Descent

The ordinary differential equation (ODE) model of GeneNet in the equation (1) shares similarities with the underlying neural network model on the matrix computing by considering I_i (prescribed impulse) as bias b_i. The basic form of the simplified classical neural network would be:

$$y_i = \phi(\sum_{i,j} W_{ij}x_j + b_i) \qquad (3)$$

By giving desired output function of the network at $\hat{y}$, such as FF model, we minimize cost between the training output and the desired function by Adam (Kingma & Ba, 2014), an effective gradient descent algorithm for this problem. In this paper, we use mean squared error as the cost function lessened by gradient descent in TensorFlow 1.7 module. The mean squared error (MSE) cost function has been defined as:

$$\arg\min_{x} ||(y(x) - \hat{y})||_2^2 \qquad (4)$$

We use the GeneNet framework with the modified learning rate for the higher-ordered network experiments in this paper. In contrast to the current BP gradient descent technique as implemented in GeneNet, our proposed ES-GeneNet, an evolutionary optimization for cost minimization would be discussed in *section* 3.3.

2.3. LASSO Regularization

To avoid over-fitting on the generated solutions, LASSO (least absolute shrinkage and selection operator, LASSO) has been used as a regression analysis method that performs both variable selection and regularization (Lim & Hastie, 2015) to enhance the prediction accuracy and interpretability of the statistical

Table 1. L1 Regularization with different coefficients

Coeff.	0	2×10^{-1}	2×10^{-2}	2×10^{-3}
Strength	-2.51	0.001	0.34	0.02

model it produces. We modify the existing MSE cost function as :

$$Cost(x) = \frac{1}{n}||y(x) - \hat{y}||_2^2 + \lambda \sum_{i,j} ||W_{ij}||_1 \quad (5)$$

With different L1 coefficients, we calculate the strength of genes by $\sum_{i,j} |W_{i,j}|$. Where λ represents the regularization coefficients, after testifying the the value among $2 \times 10^{-3} - 2 \times 10^{-1}$ and the under-fitting occur at 2×10^{-1} with an failure case on desired functional motifs. We use L1 coefficients 2×10^{-2} for the regularized GeneNet experiments in Figure 3, 4, 6, and 5.

2.4. Learning parameters using an Genetic Algorithm

Using gradient descent only a limited number of networks motifs is discovered that computed the FF function. We, therefore, explored whether an evolutionary approach finds other solutions compared to a gradient descent search, i.e., Equation (1).

An evolutionary algorithm (Fonseca et al., 1993) generate many diverse solutions among global searching domain. By relying on bio-inspired operators such as mutation, crossover, and selection, the modified weight from ES-GeneNet has more considerable diversity compared to the single pathway based on gradient descent from GeneNet. We implement the evolutionary algorithms to replace each iteration of the updating gradient to generations. We arrange 20 different updating gradients, like children in an evolutionary algorithm for each generation. The time of computation is 2.22 times faster than the result from GeneNet at three-nodes components searching.

2.4.1. CROSSOVER

First, we use a crossover rate of (60%) to enable the different weights matrices to share their learned features during the process of cost minimization. The crossover action ensures the cost function to be convex in a large search space where the search of the desired weights correspond to the network motifs.

2.4.2. MUTATION

Mutation is a genetic operator used to preserve genetic diversity from one generation of a population of evo-

---

Algorithm 1 Evolutionary Search

Input: $desired\_target\_function\ Y$
Initialize $population = randomized\_children()$.
for $g = 1$ **to** $max\_possible\_generations$ **do**
  $costs = calculate\_cost(population, Y)$
  $survived\_parents = select(population, costs)$
  $elite = survived\_parents[0]$
  if $is\_desired\_function(elite, Y)$ **then**
    Break loop
  else
    $population = repopulate(survived\_parents)$
    $population = mutate(population)$
  end if
end for
Output
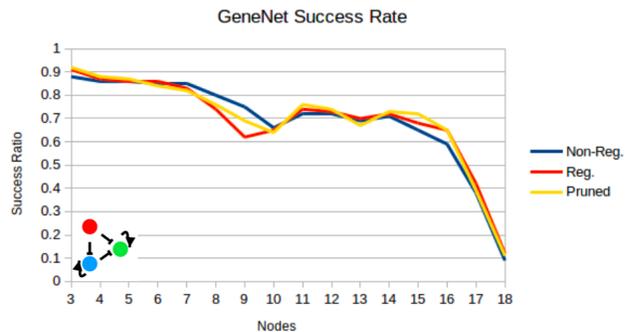
Figure 3. This figure illustrates the learnability of GeneNet on French-Flag function with increasing network size. The GeneNet algorithm is repeated 100 times and we calculate the ratio of successful learning of the FF function out of the 100 trials. Note the sharp drop around 16-18 nodes. This behaviour is consistent regardless of the degree of regularization (L1 or pruning, red and yellow respectively)

lutionary algorithm chromosomes to the next. We use mutation in two different ways in modified versions of GeneNet (1). The gradient computation is performed but with a mutation rate $\frac{0.5}{N \times N}$ (2) while doing without the backpropagation algorithm and secondly, to use random mutation and crossover to minimize the cost. Algorithm 1 shows an example.

2.5. Numerical Implementation

We run our code by Tensorflow 1.7.0 on a workstation with an Intel(R) Xeon(R) CPU E5-2680 v4 2.40GHz and a 32 GB Hynix DIMM Synchronous 2133 MHz. The code of ES-GeneNet is availble at *github.com/huckiyang/ES_GeneNet*.
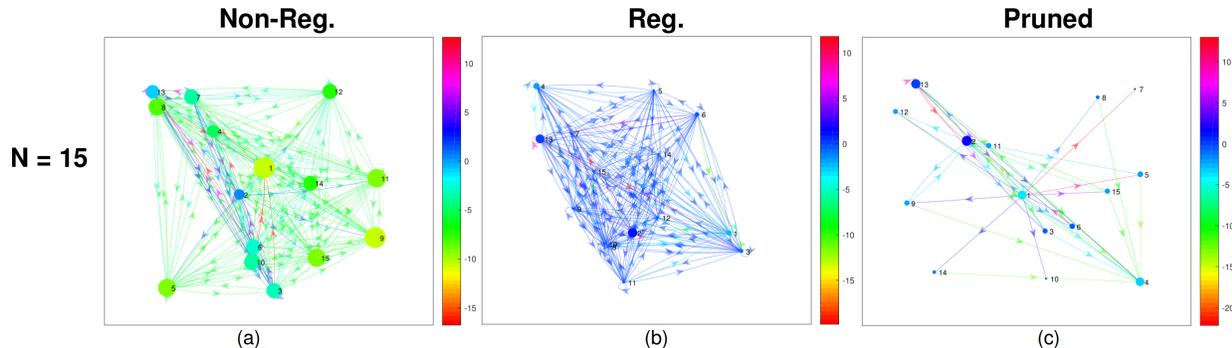
Figure 4. This figure visualizes the network interaction of learnable gene regulatory network in different gene strength on the LASSO (L1) regularization. At the network size of 7 (a) GeneNet without regularization, (b) GeneNet with L1 regularization, and (c) GeneNet with pruned L1 regularization.

3. Results and Discussion

3.1. Identification of larger ($N > 3$) Networks Motifs

First, we ask whether the gradient descent search and the genetic algorithm could find larger motifs to compute a given input-output function. The French-Flag (FF) model, is one of the most important and frequent models for gene-gene regulatory (Miller, 2004). By given desired function as $\hat{y}$ from the equation (4), we search for FF circuits larger than three nodes. For example, we use $y = 1$ for $x > 0.5$ $x < 1.5$ else $y = 0$ as a FF gene expression Model. We discover both 16-genes and 4-genes circuit motifs computing the FF functions, as illustrated in Figure 2 $(a), (b), (e)$, and (f).

We also design a gene-circuit with functional Switch model. As a crucial regulatory network model, the switch is constructed from any two repressible promoters arranged in a mutually inhibitory network. This functional gene circuit is flipped between thermal induction and exhibits a nearly ideal controlled threshold or stable states using a transient chemical. This function is parametrized as $y = 1$ for $x > 1$ else $y = 0$ and also here we find larger circuits, as illustrated by the 16-gene and 4-gene circuit motif of Figure 2 $(c), (d), (g)$, and (h).

Parametrizing the FF model, using $y = 1$ for $x > 1$ else $y = 0$ as a FF gene expression Model in 4-genes and 16-genes circuit as illustrated in Figure 2 (c), produced on average a speed-up of the computation by factors 2.73 and 7.73 respectively as compared to GeneNet. Also, we empower the Switch model, as $y = 1$ for $x > 1$ else $y = 0$ as a FF gene expression Model in 4-genes and 16-genes circuit of Figure 2 (d) with a average of 2.52 and 5.52 times faster than the computation of GeneNet.

In mathematics and computer science, connectivity

(Harris & Stephens, 1988) asks for the minimum number of elements (nodes or edges) that need to be removed to disconnect the remaining nodes from each other. We calculate the undirected edge-connectivity (Nagamochi & Ibaraki, 1992) from the generate matrix size from 3×3 up to the largest size we learn from the random generate 18×18 in Table 3. With identical desired training function, the result showed the more extensive network is represented by specific regularization motifs network to present the dynamic performance.

The nodes size of the network represents the strength in a gene regulatory circuit. Here we visualize nodes size by the value of $\sum_{i,j} |W_{i,j}|$. With stronger strength, the node would have a bigger size, which implies a dominant gene in the regulatory network. With fixed size of learned gene regulatory network from the Figure 4, the value of $\sum_{i,j} |W_{i,j}|$ would decreasing as a much neat network interaction without trivial nodes. As a input-output system, we also visualize the sum of system feedback, $\sum_{i,j} W_{i,j}$, with a color bar. For instance, as Figure 4 (a), the $1st$ node with a green and the $2nd$ node with a blue represent as a sum of feedback with positive and negative response in a gene regulatory network.

Moreover, we also feature the learned network as a tool to show the activate and deactivate relationship in an extensive system as gene regulatory network. As part of the result in Figure 4 (c), in a visualized pruned-network, we efficiently investigate the dynamics of both positive and negative impacts for each node as each gene.

3.2. There is a threshold of the size of the network motif computing an input-output function

We asked whether there is any limitation on the size of network motifs computing an input-output function. To quantify this, we use the ratio between the number of successful trials computing the function and the total number of attempts to compute the input-output function. This experiment is performed both for the FF and switch function. Surprisingly, we find a sharp drop at around 18 nodes as illustrated in Figure 3. Now, why did this drop occur? On the one hand, it is a consequence that the search space simply became too large at a certain point and that the algorithms were unable to identify appropriate solutions. Alternative the ratio of the proper solution decreases rapidly due to the exponential increase in search space. However, several observations suggest that this is not the reason. First, we observed the same threshold, regardless if we use a gradient descent search or a genetic algorithm. Secondly, the threshold is very similar for both the FF and the switch function. To further examine this we visualize the network at the transition from being able to compute the function over to the case when the network failed. Figure 6 illustrates that the sign of edges in the network gradually shift from being negative (a) to become positive (c). Hence, it appears that the negative feedback regulation in the network vanishes at the threshold. This suggests that the stability of the network is affected.

3.3. Stability analysis of the Network

To investigate the stability we estimate the size of the negative real eigenvalues as a function of the network size. In general, a fixed-point for a network is stable if the real part of the eigenvalue is negative. Using $Equation(1)$, the adjacency matrix from the GeneNet has been activation by a non-linear equation. We take a Jacobian transformation of a learned adjacency matrix on the fixed point mapping. From the Figure 6, as the orange arrow illustration, the following size effects gradually bring the stability from saddle points to stable point at the second quadrant, then at the size of 18 - the system closed to the zero-point as a threshold of stability. The regularized results among networks as $4'$, $5'$, $6'$ in a small network and $16'$, $17'$, $18'$ in a large network are echoed to the stability hypothesis on the fixed point theory.

3.4. Adaptation on Multi-Agent Learning

The previous study of GeneNet (Hiscock, 2017) on learning functional motifs has succeeded in initializ-

ing random 3-nodes genes circuits. However, higher-ordered network learning on functional gene-circuits has not yet been explored using the neural network generative model. When analyzing larger learned network motifs opens up the question if and how the other elements in the system contributes to the learned input output function. This has been referred to as the team learning problem (Panait & Luke, 2005). In brief, there is positive and negative feedback in such circumstances, which originates from the adaptation of the elements supporting team learning in order to compute the desired function. As our definition in Equation (1), we only supervise the last node, as $y_n \equiv y$, and minimize its MSE during the learning processing. Interestingly, with an increasing size effect, we observed that the agents nodes in our case tend to perform two types of Team Learning behaviors.

3.5. Homogeneous Team Learning

Team learning is an easy approach to multi-agent learning because its single learned can use standard single-agent machine learning techniques. However, a major problem with team learning is the large state space for the learning process with large dynamic with two opposite behavior homogeneous and heterogeneous team learning. With systematic feedback results, we can observe the homogeneous learning behaviors on desired FF functional circuit for different size of the networks. Homogeneous Team-Learning is an effect of a multi-agent system tend to learn the same as given desired function, even without supervised. This homogeneous learning is positive for supporting our higher-ordered gene-circuit to learn band-pass performance in FF model. The amplitude of the targeted red FF signal, $y_n \equiv y$, slightly increase 5.1% and 4.3% from (a) to (b) and from (c) to (d). Meanwhile, a series of un-supervised agents keep increasing their amplitude among the band-pass performing range.

3.6. Heterogeneous Team Learning

We find it to be different from previous observations as small as 9. Negative feedback after the size attains a large size at 16 as a Heterogeneous Team Learning effect. And, the un-supervised agents are pulled back away from the desired features of the supervised agent. Meanwhile, our multi-agent network is gradually losing its learnability and abruptly crashed down at the size of 18×18 with a corresponding low successful learning rate at 9% in 3. The amplitude of the targeted red FF signal, $y_n \equiv y$, dramatically drop 12.1% and 9.3% from (d) to (e) and from (e) to (f).

N = 16 **N = 17** **N = 18**

(a) (b) (c)
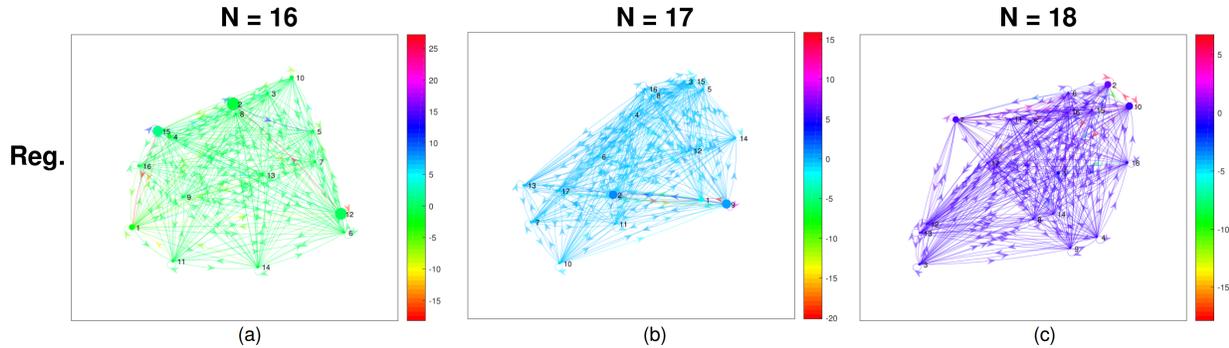
Figure 5. At the threshold of learnability there is a shift in the weight distribution from negative to positive weights. From a most negative weights at (a) nodes = 16, the shifting weights to near zero points at (b) nodes = 17, and finally most of the weights became positive at (c) nodes = 18. This shift in the weight distribution is in accordance with the stability map of eigenvalues as displayed in Figure 6
.
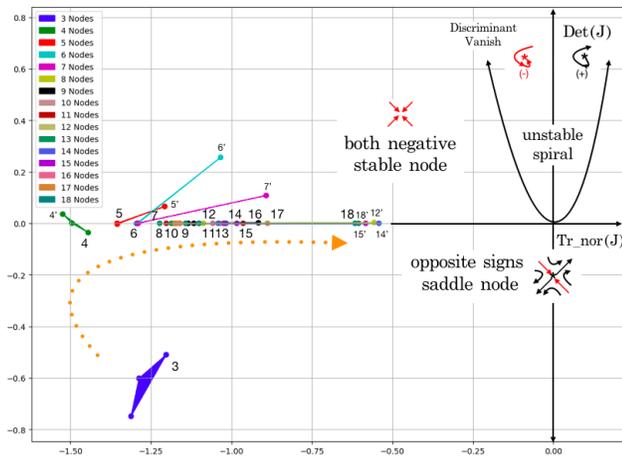
Figure 6. This figure show stability map of Jacobian matrices from bio-motifs. The process shows stability by routing different network size and evolutionary process on the map.

that the positive-feedback turn to opposite after the network size becomes too large. The un-supervised agent is drawn down the desired features from the supervised agent. Meanwhile, our multi-agent network is gradually losing its learnability and suddenly crash down at the size of 18×18 with a successful learning rate less than 9%, see Figure 3.

Original **Regulated** **Pruned**

(a) (b) (c)
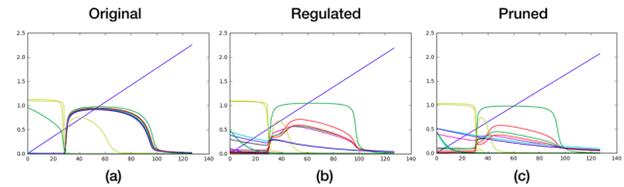
Figure 7. This figure shows an changing system feedback from a homogeneous team learning to a heterogeneous team-learning effect at the network size/agent numbers = 17 with (a) original (b) regulated (c) pruned functional French-Flag gene-circuit

3.7. Summary of the observed Team Learning

In summary, we found positive feedback from the un-supervised agents, i.e. the other nodes in the network motifs, echoes in their learned profiled the overall learned function in the network motif. Note, that this is unexpected as the supervised training is only conferred to the last node given the input to the first node, as the behaviour of all other nodes and interactions were left unconstrained. Note that the learned network motif input output function is only imposed on the system as a whole. This effect is functionally useful for learning in a complex system like the gene regulatory network. The homogeneous team learning turn to drag down the third agent from the band-pass performance and lead an unstable system. Also, note

4. Conclusion

In summary, we demonstrated the existence of larger functional input output network motifs in contrast of what have previously been described. To achieve this we had to adapt an ML approach to design and search for functional high-order gene regulatory network motifs. The rational is that an exhaustive search for larger network motifs does not scale due to the combinatorial complexity in the search space with increasing size of the network motifs. To this end we use both gradient descent and genetic algorithms. The observed threshold in learning that we observed is most likely due to loss of stability with increasing size. This effect requires further investigation as it would most likely have to with the specific structure of the Jacobian ma-

trix as the system size grows. Hence, the elements which are not directly constrained in the learning, i.e. the nodes in between the input and output have to organize themselves in such a manner that stability is preserved for as large system as possible. However, for some reason this does not apparently scale indefinitely. We conjecture that this phenomena has to do with the observed adaptive learning *in vivo* multi-agent behavior that we observed in the network motifs. Our proposed novel evolutionary algorithm framework to solve the ODE system without being trapped by local minimum with a single phenotype of FF-motifs hold the promise to be used for massive large-scale explorations. Both crossover mutation actions increase the diversity of the search space to find more diverse motifs-weight compared to the previous gradient descent method. This opens the possibility to not only identify multiple motifs of a given size for an input-output function. It could also explore higher-ordered motifs with adaptive team-learning in dynamic bio-system. Thus, our further work will focus on investigating biological properties and the analyze the high frequent motifs in the learned dynamic networks from GeneNet and ES-GeneNet.

Acknowledgement

References

Bengio, Yoshua, Courville, Aaron, and Vincent, Pascal. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Chavoya, Arturo and Duthen, Yves. A cell pattern generation model based on an extended artificial regulatory network. *Biosystems*, 94(1-2):95–101, 2008.

Cotterell, James and Sharpe, James. An atlas of gene regulatory networks reveals multiple three-gene mechanisms for interpreting morphogen gradients. *Molecular systems biology*, 6(1):425, 2010.

Cussat-Blanc, Sylvain, Bredeche, Nicolas, Luga, Hervé, Duthen, Yves, and Schoenauer, Marc. Artificial gene regulatory networks and spatial computation: A case study. In *ECAL*, 2011.

De Jong, Hidde. Modeling and simulation of genetic regulatory systems: a literature review. *Journal of computational biology*, 9(1):67–103, 2002.

Ellis, Tom, Wang, Xiao, and Collins, James J. Diversity-based, model-guided construction of synthetic gene networks with predicted functions. *Nature biotechnology*, 27(5):465, 2009.

Fonseca, Carlos M, Fleming, Peter J, et al. Genetic algorithms for multiobjective optimization: Formulationdiscussion and generalization. In *Icga*, volume 93, pp. 416–423. Citeseer, 1993.

Gardner, Timothy S, Cantor, Charles R, and Collins, James J. Construction of a genetic toggle switch in escherichia coli. *Nature*, 403(6767):339, 2000.

Grover, Aditya and Leskovec, Jure. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864. ACM, 2016.

Harris, Chris and Stephens, Mike. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pp. 10–5244. Citeseer, 1988.

Hiscock, Tom W. Adapting machine-learning algorithms to design gene circuits. *bioRxiv*, pp. 213587, 2017.

Joachimczak, Michał and Wróbel, Borys. Evolution of the morphology and patterning of artificial embryos: scaling the tricolour problem to the third dimension. In *European Conference on Artificial Life*, pp. 35–43. Springer, 2009.

Khalil, Ahmad S and Collins, James J. Synthetic biology: applications come of age. *Nature Reviews Genetics*, 11(5):367, 2010.

Kingma, Diederik P and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lim, Michael and Hastie, Trevor. Learning interactions via hierarchical group-lasso regularization. *Journal of Computational and Graphical Statistics*, 24(3):627–654, 2015.

Lu, Timothy K, Khalil, Ahmad S, and Collins, James J. Next-generation synthetic gene networks. *Nature biotechnology*, 27(12):1139, 2009.

Miller, Julian Francis. Evolving a self-repairing, self-regulating, french flag organism. In *Genetic and Evolutionary Computation Conference*, pp. 129–139. Springer, 2004.

Milo, Ron, Shen-Orr, Shai, Itzkovitz, Shalev, Kashtan, Nadav, Chklovskii, Dmitri, and Alon, Uri. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

Mutalik, Vivek K, Guimaraes, Joao C, Cambray, Guillaume, Lam, Colin, Christoffersen, Marc Juul, Mai, Quynh-Anh, Tran, Andrew B, Paull, Morgan, Keasling, Jay D, Arkin, Adam P, et al. Precise and reliable gene expression via standard transcription and translation initiation elements. *Nature methods*, 10(4):354, 2013.

Nagamochi, Hiroshi and Ibaraki, Toshihide. Computing edge-connectivity in multigraphs and capacitated graphs. *SIAM Journal on Discrete Mathematics*, 5(1):54–66, 1992.

Panait, Liviu and Luke, Sean. Cooperative multi-agent learning: The state of the art. *Autonomous agents and multi-agent systems*, 11(3):387–434, 2005.

Reil, Torsten. Dynamics of gene expression in an artificial genomeimplications for biological and artificial ontogeny. In *European Conference on Artificial Life*, pp. 457–466. Springer, 1999.

Wu, Gang, Yan, Qiang, Jones, J Andrew, Tang, Yinjie J, Fong, Stephen S, and Koffas, Mattheos AG. Metabolic burden: cornerstones in synthetic biology and metabolic engineering applications. *Trends in biotechnology*, 34(8):652–664, 2016.

Xing, Eric P and Karp, Richard M. Motifprototyper: a bayesian profile model for motif families. *Proceedings of the National Academy of Sciences of the United States of America*, 101(29):10523–10528, 2004.