Influence of Promoter Length on Network Convergence in GRN-based Evolutionary algorithms

Paul Tonelli, Jean-Baptiste Mouret, and Stéphane Doncieux

Institut des Systèmes Intelligents et Robotiques UPMC-Paris 6, CNRS UMR 7222
4 place Jussieu, 75252 Paris Cedex,

Abstract. Genetic Regulation Networks (GRNs) are a model of the mechanisms by which a cell regulates the expression of its different genes depending on its state and the surrounding environment. These mechanisms are thought to greatly improve the capacity of the evolutionary process through the regulation loop they create.

Some Evolutionary Algorithms have been designed to offer improved performance by taking advantage of the GRN mechanisms. A recent hypothesis suggests a correlation between the length of promoters for a gene and the complexity of its activation behavior in a given genome. This hypothesis is used to identify the links in in-vivo GRNs in a recent paper and is also interesting for evolutionary algorithms. In this work, we first confirm the correlation between the length of a promoter (binding site) and the complexity of the interactions involved on a simplified model. We then show that an operator modifying the length of the promoter during evolution is useful to converge on complex specific network topologies. We used the Analog Genetic Encoding (AGE) model in order to test our hypothesis.

1 Introduction

Evolutionary Algorithms (EA) are nowadays able to offer improved solutions for many problems and sometimes outperform engineering methods. Though, we are unable to obtain solutions as complex as what in-vivo evolution has produced. Some of the evolutionary mechanisms behind biological evolution are still unexplained. It is believed EA could benefit from understanding these mechanisms [1]. There are many of these mechanisms, as for example splicing, through which a single gene can encode multiple proteins [2, 3]. Genetic Regulation Networks (GRNs) are a model of other of theses mechanisms by which a cell regulates the expression of its different genes depending on its state[4]. A strong hypothesis is that the complexity of the interactions obtained thanks to the GRNs is, at least, partly responsible for the diversity increase of many living organisms [5], as well as improvements in the genome evolvability and robustness [6]. Though, the complexity of these mechanisms is still not fully understood [7].

We believe that understanding which in-vivo characteristics improve the performance of biological evolution is a key to designing efficient EA. In order to do so, it is necessary to identify which elements of these mechanisms are necessary and must be reproduced to improve performance. In this work, we investigated if one of these mechanisms is relevant: a mutation operator adding or removing one base in string based evolutionary algorithms. This operator is usually considered minor, as in-vivo RNA is read three bases at a time (codon) during protein synthesis, and adding or deleting a base in DNA shifts the sequence, creating a protein totally different from the original. Though, in the case of non-coding DNA, this operator may be more useful. In the following work, we asked ourselves two questions. First, does the add/remove operator provide a sufficient mechanism to obtain a correlation between the length of cis-regulatory sequences as stated in [8]. And more important, does this operator improve the performance of the algorithm in any way ?

2 Related Works

2.1 Gene Regulation Networks

GRNs rely on multiple mechanisms. One of them is the possibility for a protein, called a transcription factor to bind itself to a sequence of DNA located before or after a given gene. When a relevant protein binds itself to the site, the expression of the gene will either be enhanced (enhancer) or blocked (inhibitor). Most genes present such cis-regulatory sequences, which contain multiple binding sites for various proteins. These mechanisms can first be seen as a mean to create “programs”, enhancing the cell capacity to order the synthesis of proteins or reactions to adapt to specific stimuli [9].

GRNs are also believed to speed up evolution. A single mutation in a cis-regulatory region can have varying impact on an organism without modifying the gene itself, for example by removing or creating a new interaction between the regulated gene cis-regulatory sequence and another transcription factor. Duplication of a transcription factor gene or binding site also creates new interactions in a genome [10]. Therefore interactions provided by the GRNs provide another level impacted by evolution.

2.2 Existing Methods

The objective to find ways to harness the properties of these GRNs to improve the performance of evolutionary algorithms is stated in [1]. More precisely, the goal is to understand which GRN properties improve the evolvability of living organisms. Some properties have already been highlighted in several articles [11–13]. Algorithms have been created to take advantage of them as Artificial Ontogeny [14] or lately PBGA [15]. We tried to find a model closely related to the biological mechanisms while avoiding the overhead of more complex biology based models like HeRoN [16].

2.3 Research of relevant properties of GRNs

The first step to take advantage of the GRN capabilities is to understand their properties and the implications for evolvability. Examples of these properties are found in [11] which tries to understand how varying goals coupled to specific evolution mechanisms can change the evolvability of a genome to speed up convergence on specific problems. It highlights the fact that the specificity of transcription factors to multiple binding sites is, in itself, a way to encode evolutionary information. The mechanisms behind GRNs are quite complex and multiple parameters are still unknown. Here, we restrict our study to algorithms using both a string based encoding while keeping simple enough matching mechanisms. AGE is the only existing example we could find to fulfill these conditions. A work similar to this one was done by C. Mattiussi [17], who studied the impact of a mutation operator allowing the duplication of sequences in a genome to obtain convergence of the algorithm. Here, we do a similar work with the possibility to incrementally modify the length of the cis-regulatory sequences.

2.4 AGE

Analog Genetic Encoding is an indirect encoding mechanism which was designed to use some mechanisms of the GRNs [17] to generate networks by evolving a string based EA. It has recently shown impressive results for the reverse engineering of existing in-vivo GRNs[18]. AGE features complex generation mechanisms, many of which are not relevant to our problem, therefore, we chose to focus only on part of these mechanisms which will be described here. Our goal, as in [11] is to assess the capacity of a set of nodes to converge to a specific network topology. In AGE, a network is composed by a set of genes. Each gene is equivalent to a node in a network (see figure 1). Each gene is composed by output and input sequences, each of which is located in the gene and located between a start and an end sequence (which could be compared to start and stop codons). The links in the network are defined by comparing the input and output sequences of different nodes. Similar sequences will be considered as a strong link between two nodes while two totally different sequences will be considered as an absence of link. As expressed in [17], this is quite similar to the process by which micro RNA can repress the expression of other genes by blocking their DNA or RNA expression [19]. For this model, the strength of the link is computed using a local alignment score [20].

3 Experiments

For our experiment, we considered a fixed set of nodes (three to five, depending on the experiment). Each node is composed of two sequences of nucleotides. One input sequence which models the promoter sequence, and one output sequence, loosely modelling the transcription factor / microRNA. Our algorithm mutates theses two sequences by using three possible operators. The first (second) one is the addition (deletion) of a nucleotide at some point in the sequence. Another point is that we do not extend our algorithm to differentiating inhibitory and enhancing links.

The fitness of a network is obtained by comparing the network topology to a reference network. An optimal link strength (1.0) is defined by two sequence containing a common subsequence of three bases. The absence of link is defined as two sequences with a longest common subsequence between two sequences of one base. The intermediary is considered as a link with a strength of 0.5.

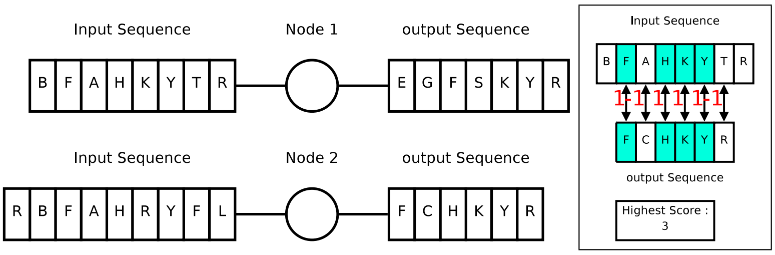

Fig. 1. Left: Two sample nodes in our simplified implementation of AGE; Right: the sequence comparison algorithm for definition of the link between the output sequence of the bottom node and the input sequence of the top node. Only contiguous bases are considered, the longest subsequence for this comparison is therefore of 3 bases even if both also have the F in common.

We made two different experiments related to the length of the sequence of the binding sites. The first one is used to test the correlation between the length of a sequence and the number of links this sequence has in the network. The second experiment tests the effect of switching on and off the add/remove operator on the convergence speed for a "complex" network. All the runs were done using the simplified version of AGE described previously where the only additional mutation operators possible are the exchange of a base in the sequence for another and the addition / deletion operators. In order to improve the performance, the size of the alphabet was set at 7 (this gives the best results for our networks). There is no order relationship between the bases in the alphabet (the replacing mutation operator switches randomly from one base to the other). The selection algorithm used for all the experiments was NSGA 2 [21], a commonly used tournament based multi objective selection algorithm as further experiments required multiple objectives. Each run was repeated at least 10 times. Figure 2 sums up all the parameters used for the experiments.

3.1 Convergence of Sequence Length

The first experiment was done in two steps. In the first step, we tried to evolve two simple networks of 3 nodes with homogeneous (2 output links and 2 input links per node) or heterogeneous links (2 outputs for each node but 1 to 3 inputs

Parameters			
alphabet size	7	population	200
maximum number of generation	10000	probability to delete base	0.01
probability to add base	0.02	probability to mutate base	0.1
max sequence length	20		

Fig. 2. Summary of the parameters.

per node) as shown in figure 3 and we analysed the length of all the sequences of the first individual to reach the optimal fitness in each run.

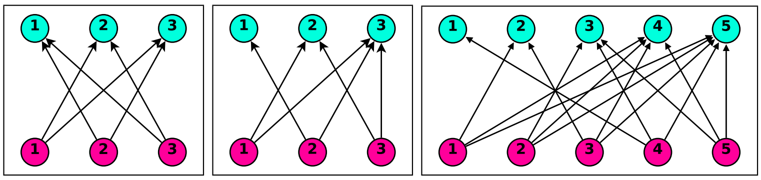

Fig. 3. Left: homogeneous network of three vertex (the top circles show the outputs, the bottom ones the inputs); Center: heterogeneous (different number of inputs and same numbrer of outputs) network with 3 nodes; Right: heterogeneous network with 5 nodes.

The results of these experiments show a convergence to a size of 7 for each sequence in the homogeneous network. In the experiment trying to converge on a 3 node heterogeneous network (center box of figure fig:networkstype),we obtained a correlation between the length of the sequence and the number of links connected to the node. As the differences were not significant, we made a similar experiment with a network containing 5 nodes and heterogeneous connections (nodes had 3 outputs each and respectively 1, 2, 3, 4 and 5 inputs). All the results (mean length of the sequences and standard deviation) are shown in figure 4.

For the results on the 5 nodes network, we have a significant difference (using Wilcoxon T-test) as the probability of the two sequences being from the same data is less than 1% between all the nodes having different numbers of inputs apart from between sequences 2 and 3 where this probability is 7.5%. These results confirm our first hypothesis, which is that the length of the sequences illustrates the complexity of the interactions the node is involved in. The add/remove mutation operator is therefore sufficient to obtain these results. This is also a confirmation of the results stated in [8].

3.2 How the operator affects performance of the EA

The first experiment showed that the length of the cis-regulatory sequence depends on the complexity of the interactions the node is involved in. Our next

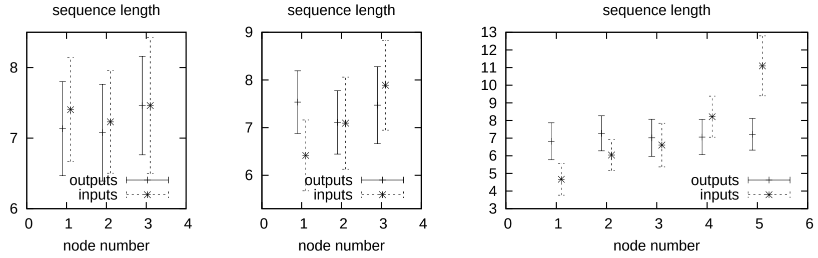

Fig. 4. Left: sequence length for an homogeneous network containing 3 nodes; Center: sequence length for an heterogeneous network containing 3 nodes; Right: sequence length for an heterogeneous network containing 5 nodes.

step was to test if a network correctly initialized could converge without length modification during the run or if the modification of the length improves performance. In evident cases (simple 3 nodes networks), the add/remove operator doesn't have a significant impact on the results unless initial parameters are initialized to abnormal values. Therefore, we experimented on a network which the algorithm has difficulties to converge to. In order to place ourselves in the worst possible scenario for our hypothesis (necessity of the add remove operator), we took an homogeneous symmetric target network where each node has 4 inputs and 4 outputs and tried to compare the performance. To do so, we first randomly initialized a 5 node network and made it converge to the target network with the addition / deletion operator enabled. It converged in all of the runs and showed a mean length of 8 bases per sequence and a standard deviation around 1. We then ran the same experiments with length modification operators disabled and a fixed initial length of 8 corresponding to the "optimal" length and compared it to a similar network without the length modification enabled.

The results were that the runs without sequence length modification were significantly faster than the runs with the operator enabled. However, this is a case where we specified the optimal length before running the algorithm, which is an unusual situation. Therefore, we made several additional runs of both algorithms by changing only the initial sequence length to compare their performance. The results are given in figure 5 and show that, if the initial length is not optimal, the add/remove operator is a good way to avoid seeing the algorithm get stuck because of insufficient initial complexity. It also helps the convergence rate in non optimal bootstrap cases as can be seen in figure 6. Therefore, we believe that, as the optimal situation of both an homogeneous network and predefined optimal sequence length is unusual, it is usually a good idea to enable the sequence length modification. The alternative being to define the optimal length by another mean before running the algorithm.

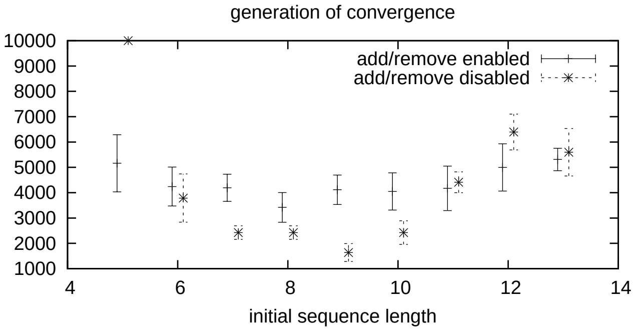

Fig. 5. Performance of the EA on a complex network for different bootstrap lengths.

bootstrap length	5	6	7	8	9	10	11	12	13
add/remove enabled	1	1	1	1	1	1	1	1	0.8
add/remove disabled	0	1	1	1	1	1	1	1	0.4

Fig. 6. Convergence rate of experiments depending on bootstrap length and specified operators.

4 Conclusion and Discussion

In this work we have first confirmed that there is a correlation between cis-regulatory sequence length and the complexity of interactions the following gene is involved in, but also that the dynamic modification of sequence length is a useful operator (sometimes to allow convergence of the EA, or to facilitate convergence in complex situations). We have also illustrated that a simple evolution mechanism is able to take advantage of these operators, at least in certain cases.

The results shown in the first experiment converge around a length of 7 for both the 3 nodes network while the 5 nodes network converge on a length of 8. Therefore, it could be argued that the optimal length is usually around 8 and that, as this length is sufficient, the add/remove operator can be disabled to improve performance. This is partly true for these sandbox networks (in experiment 2, the best performance is achieved for a bootstrap length of 9). The goal of AGE and other EA is eventually to solve complex problems, with potentially many more nodes and links. In these situations, using a default a length of 8 is raises a risk that the network might not offer enough complexity to converge, as was the case with the 5 bases long runs of the third experiment and therefore be unable to converge.

References

1. Banzhaf, W., Beslon, G., Christensen, S., Foster, J.A., Képès, F., Lefort, V., Miller, J.F., Radman, M., Ramsden, J.J.: GuidelinesFrom artificial evolution to computa-

tional evolution: a research agenda. Nature Reviews Genetics **7**(9) (2006) 729–735

2. Tischer, E., Mitchell, R., Hartman, T., Silva, M., Gospodarowicz, D., Fiddes, J.C., Abraham, J.A.: The human gene for vascular endothelial growth factor. Multiple protein forms are encoded through alternative exon splicing. Journal of Biological Chemistry **266**(18) (1991) 11947–11954
3. Modrek, B., Lee, C.: A genomic view of alternative splicing. Nature genetics **30**(1) (2002) 13–19
4. Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Joseph, B.Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., Others: Transcriptional regulatory networks in Saccharomyces cerevisiae. Science **298**(5594) (2002) 799–804
5. Levine, M., Tjian, R.: Transcription regulation and animal diversity. Nature **424** (2003) 147–151
6. Wagner, A.: Robustness and evolvability in living systems. Princeton University Press Princeton, NJ (2005)
7. Wittkopp, P.J.: Evolution of cis-regulatory sequence and function in Diptera. Heredity **97**(3) (2006) 139–147
8. Kristiansson, E., Thorsen, M., Tamas, M.J., Nerman, O.: Evolutionary forces act on promoter length: identification of enriched cis-regulatory elements. Molecular Biology and Evolution (2009)
9. Alon, U.: An introduction to systems biology: design principles of biological circuits. Chapman & Hall/CRC (2007)
10. Teichmann, S.A., Babu, M.M.: Gene regulatory network growth by duplication. Nature Genetics **36**(5) (2004) 492–496
11. Izquierdo, E.J., Fernando, C.T.: The Evolution of Evolvability in Gene Transcription Networks. Artificial Life **11** (2008) 265
12. Tanay, A., Regev, A., Shamir, R.: Conservation and evolvability in regulatory networks: The evolution of ribosomal regulation in yeast. Proceedings of the National Academy of Sciences **102**(20) (2005) 7203–7208
13. Chen, K., Rajewsky, N.: The evolution of gene regulation by transcription factors and microRNAs. Nature Reviews Genetics **8**(2) (2007) 93–103
14. Bongard, J.C.: Evolving modular genetic regulatory networks. In: Proceedings of The IEEE 2002 Congress on Evolutionary Computation (CEC2002). (2002) 1872–1877
15. Bellas, F., Becerra, J.A., Duro, R.J.: Using promoters and functional introns in genetic algorithms for neuroevolutionary learning in non-stationary problems. Neurocomputing (2008)
16. Gonçalves, A., Costa, E.: A Computational Model of Gene Regulatory Networks and its Topological Properties. Artificial Life **11** (2008) 204
17. Mattiussi, C.: Evolutionary synthesis of analog networks. PhD thesis, Università degli Studi di Trieste (2005)
18. Marbach, D., Mattiussi, C., Floreano, D.: Replaying the evolutionary tape: Biomimetic reverse engineering of gene networks. Annals of the New York Academy of Sciences (2008)
19. Ruvkun, G.: Molecular biology: glimpses of a tiny RNA world. Science's STKE **294**(5543) (2001) 797
20. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. Journal of Molecular Biology **147** (1981) 195–197
21. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation **6**(2) (2002) 182–197