

# Why Don't the Modules Dominate?

Zhenyue Qin  
Australian National University  
Canberra, ACT, Australia  
zhenyue.qin@anu.edu.au

Tom Gedeon  
Australian National University  
Canberra, ACT, Australia  
tom.gedeon@anu.edu.au

R I (Bob) McKay  
Australian National University  
Canberra, ACT, Australia  
robert.mckay@anu.edu.au

## ABSTRACT

Wagner's modularity inducing problem domain is a key contribution to the study of the evolution of modularity, including both evolutionary theory and evolutionary computation. We study its behavior under classical genetic algorithms. Unlike what we seem to observe in nature, the emergence of modularity is highly conditional and dependent, for example, on the eagerness of search. In nature, modular solutions generally dominate populations, whereas in this domain, modularity, when it emerges, is a relatively rare variant. Emergence of modularity depends heavily on random fluctuations in the fitness function; with a randomly varied but unchanging fitness function, modularity evolved far more rarely. Interestingly, high-fitness non-modular solutions could frequently be converted into even-higher-fitness modular solutions by manually removing all inter-module edges. Despite careful exploration, we do not yet have a full explanation of why the genetic algorithm was unable to find these better solutions.

## CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability;

## KEYWORDS

Biological networks, Evolution of modularity

### ACM Reference Format:

Zhenyue Qin, Tom Gedeon, and R I (Bob) McKay. 2018. Why Don't the Modules Dominate?. In *GECCO '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3205651.3205737>

## 1 INTRODUCTION

Modularity is the divisibility of structures or functions into sub-units that perform autonomously [5]. Lack of modularity has been shown as a key limitation of artificial biological systems in scaling to higher complexity [2]. Despite decades-long research interest in modularity, there is no consensus on its biological origin [1].

Espinosa-Soto and Wagner<sup>1</sup> studied the emergence of modularity in an artificial environment requiring gene specialization [1], in which gene regulatory networks need to be able to regulate toward

<sup>1</sup>Since we repeatedly need to refer to this crucial paper, we abbreviate it as ES&W

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

*GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan*

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5764-7/18/07.

<https://doi.org/10.1145/3205651.3205737>

multiple different patterns. This is a common occurrence in nature: the same collection of genes frequently exhibits different activity patterns at different phases of development or in different locations in an organism [1].

The results were persuasive, but also exhibited one key difference from nature: although modular structures emerged, they did not dominate populations, whereas in biology, once modular structures emerge, they become universal across a species. Larson et al. [3] subsequently extended this work (within a modified environment) to explore the effects of recombination on modularity emergence.

The remainder of this paper details our deeper exploration of the fitness landscape of this intriguing environment. We do not see this as a criticism of the original use of the model. It is its extended use as a test-bed for understanding modularity that is at issue here: whether further investigation of the comparative behavior of different algorithms on this problem gives any real insight into the equivalent behavior either in biology, or in real-world application of evolutionary algorithms.

Due to the page limitation, we can only present essential findings in this paper. We have put a more detailed paper on arXiv to provide more information about our work.

## 2 METHODS

Cells interpret the same genetic material in different ways so that their behaviors and structures vary. These distinct interpretations are due to regulation, among other mechanisms via the activation and repression of genes by other genes [1]. This mechanism can be usefully abstracted by a weighted directed graph. The term “gene activity pattern” describes the activeness status of the entire set of genes. We used ES&W's [1] representations of a gene activity pattern and a gene regulatory network. We adopted the Q scoring system to quantify modularity in a network, based on the algorithm proposed by Newman [4]. All are significance tested using Wilcoxon's Signed-Rank Test.

## 3 EXPERIMENTS AND RESULTS

### 3.1 Greedier Search Hampers Modularity

#### 3.1.1 Elitism Hampers Modularity.

We trialled simulations with an elite of 10 against simulations with no elites. Significantly lower best fitness (regulatory capability) and modularity arose when an elite was used.

#### 3.1.2 Proportional Selection generates Better Fitness than Tournament Selection, but Lower Modularity than Small Tournaments.

At least when populations are approaching convergence, tournament selection, especially with larger tournaments, imposes stronger selection pressure than proportional selection. In these results, we see fairly much the anticipated decline in ultimate fitness with increasing selection pressure (though none of the differences

are significant at the 1% level). Interestingly, proportional selection exhibits an unexpected (but non-significant) lower modularity than tournaments of size 2 or 3, but generates significantly more modularity than tournaments of size 10.

### 3.2 Dynamically Stochastic Fitness Evaluation generates Higher Fitness and Modularity than Static

We compared dynamically stochastic fitness evaluation (in which the perturbations are generated anew each generation) and static fitness evaluation (the perturbations are generated once for all at the start of each run). Dynamic fitness evaluation outcompeted static on both survivability and modularity, though the latter result is not significant at the 1% level.

## 4 ANALYSIS AND DISCUSSION

### 4.1 Modular systems did not gain dominance via selection for robust target recovery

While some algorithm variants permitted modularity to emerge and survive under selection for robust target recovery, it did not (in extreme contrast to natural systems [5]) come to dominate populations. We saw some indication that greedier mechanisms impeded the emergence of modularity. This implies that the most competitive elites in each generation did not have the most modular gene regulatory networks.

Overall, this suggests that while the ES&W framework has been useful to demonstrate particular properties of the emergence of modularity, in particular from pressure on gene specialization, it may not be sufficiently nature-like to function as a useful testbed to explore algorithmic effects on modularity emergence. If we see differences in algorithm behavior on this testbed, we will not know whether they arise from the abstractions from the natural environment. If not, any extrapolations would be moot.

**Table 1: Modularity dominance for data from the previous data**

Generation Range	Modularity	Fitness
(500, 2000)	0.5000	0.9482
	0.1736	0.9502

To further investigate the behavior of this system, we took more detailed measurements from the previous data. Specifically, from each run, we collected the fittest gene regulatory network among networks that were the most modular; and conversely, the network that was least modular among those that had the greatest fitness value. We expected the mean fitness of the latter to be lower than the former. Surprisingly, the situation was reversed: less modular networks generally recovered the target more robustly than more modular, as shown in Table 1. This does not reflect biological observations.

### 4.2 Inter-Module Connections Can Hamper Network Fitness

To further investigate this phenomenon, we took the 40 networks that were least modular among those having the greatest fitness value from the first experiment in subsection 4.1. We wondered what would happen if we simply removed all non-modular interconnections. So we did so with all 40 networks, and measured fitness after this removal (modularity was, of course, perfect).

Of these 40 relatively low-modularity (Q) but near-optimal fitness networks, 24 exhibited *even higher* fitness after manually deleting inter-module edges. That is, more than half these originally non-modular networks exhibited better fitness performance after removing all the inter-module connections. These 24 much-improved networks (higher fitness than any found in the run, but also much higher modularity) were not only available to the evolutionary algorithm – they were even relatively nearby (a few edge deletion mutations away). Yet the algorithm reliably did not find them!

## 5 CONCLUSIONS

It is of considerable importance to identify those characteristics of evolutionary algorithms that will lead to modular problem solutions. Our belief that this is generally feasible is heavily reliant on the ubiquity of modular solutions in biological evolution. So useful testbeds for exploring algorithms' propensity to generate modular solutions need to abstract the relevant aspects of the real world. We have seen that from this perspective, ES&W's environment exhibits a number of anomalies. It is highly sensitive to the eagerness of search. While it supports the emergence of modularity, it does not appear to support their dominance, in contrast to real-world behavior. More surprising, we can manually find high-fitness paths that lead to modular solutions of even higher fitness, and these paths are favored by the mutation bias, yet the algorithm does not find them. This appears to be a result of the dynamic nature of the fitness function: the paths to these modular solutions may not appear favorable to the algorithm because of the dynamic stochastic variations in the fitness landscape. Paradoxically, this dynamic aspect seems essential to the emergence of modularity at all. The static variant of this problem does not support the emergence of modularity under standard genetic algorithms (we accept that it does so under strongly diversity-encouraging algorithms such as age-Pareto algorithms). This appears to be linked to the genetic algorithms becoming trapped in flat, featureless regions of the fitness landscape, which the small fluctuations in the fitness landscape of the dynamic variant allow them to escape.

## REFERENCES

- [1] Carlos Espinosa-Soto and Andreas Wagner. 2010. Specialization can drive the evolution of modularity. *PLoS computational biology* 6, 3 (2010), e1000719.
- [2] Nadav Kashtan and Uri Alon. 2005. Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences of the United States of America* 102, 39 (2005), 13773–13778.
- [3] Ari Larson, Anton Bernatskiy, Collin Cappelle, Ken Livingston, Nicholas Livingston, John Long, Jodi Schwarz, Marc Smith, and Josh Bongard. 2016. Recombination Hotspots Promote the Evolvability of Modular Systems. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*. ACM, 115–116.
- [4] Mark EJ Newman. 2006. Modularity and community structure in networks. *Proceedings of the national academy of sciences* 103, 23 (2006), 8577–8582.
- [5] Gerhard Schlosser and Günter P Wagner. 2004. *Modularity in development and evolution*. University of Chicago Press.