Robustness, Evolvability, and Accessibility in Linear Genetic Programming

Ting Hu[1,⋆], Joshua L. Payne[1,⋆], Wolfgang Banzhaf[2], and Jason H. Moore[1]

[1] Computational Genetics Laboratory, Dartmouth Medical School, Lebanon, NH 03756, USA
{Ting.Hu,Joshua.L.Payne,Jason.H.Moore}@Dartmouth.edu
[2] Department of Computer Science, Memorial University of Newfoundland, St. John's, NL, Canada
banzhaf@mun.ca

Abstract. Whether neutrality has positive or negative effects on evolutionary search is a contentious topic, with reported experimental results supporting both sides of the debate. Most existing studies use performance statistics, *e.g.*, success rate or search efficiency, to investigate if neutrality, either embedded or artificially added, can benefit an evolutionary algorithm. Here, we argue that understanding the influence of neutrality on evolutionary optimization requires an understanding of the interplay between robustness and evolvability at the genotypic and phenotypic scales. As a concrete example, we consider a simple linear genetic programming system that is amenable to exhaustive enumeration, and allows for the full characterization of these properties. We adopt statistical measurements from RNA systems to quantify robustness and evolvability at both genotypic and phenotypic levels. Using an ensemble of random walks, we demonstrate that the benefit of neutrality crucially depends upon its phenotypic distribution.

1 Introduction

Redundant mappings between genotype and phenotype are common in Genetic Programming (GP). Such encodings often produce neutrality [1], where many mutational variants of a genotype produce identical phenotypes. Conflicting opinions have been proposed with regard to the effects of neutrality on evolutionary search. While some studies have found no benefit [2,3,4], others have claimed that neutrality provides a buffer against deleterious genetic perturbation [5,6] and reduces the risk of premature convergence through an expansion of the search space [7,8]. As argued in [9], the lack of consensus regarding the benefits of neutrality largely stem from the overly complex problems, representations, and search algorithms used in these analyses, which make it difficult to tease apart the effects of neutrality from other confounding factors. Further, neutrality is often artificially added to the problem representation and little attention is paid to how this alters the fitness landscape [9].

⋆ These authors contributed equally to this work.

S. Silva et al. (Eds.): EuroGP 2011, LNCS 6621, pp. 13–24, 2011.

In natural systems, neutrality is often discussed in terms of robustness, which can be defined as an organism's ability to maintain its phenotype in the face of genetic perturbation. Of particular interest is the relationship between robustness and evolvability [10,11], which can be defined as an organism's ability to generate novel phenotypes. At first glance, robustness and evolvability appear contradictory. While the former requires genetic alterations leave the phenotype intact, the latter requires these alterations be used for the exploration of new phenotypes. Despite this apparent paradox, empirical analyses of several natural systems have revealed that robustness can actually facilitate evolvability [12,13,14]. In the cytochrome P450 BM3 protein, for example, increased robustness increases the probability that mutants can exploit new substrates [12].

To elucidate the relationship between robustness and evolvability in natural systems, several theoretical models have been put forth (*e.g.*, [15,16,17,18]). Central to many of these analyses is the concept of a genotype network (a.k.a. a neutral network), where vertices represent genotypes and edges connect genotypes that share the same phenotype and can be interconverted via single mutational events [10]. In this framework, robust phenotypes correspond to large genotype networks, where most mutations are neutral and therefore leave the phenotype unchanged. Robust phenotypes are evolvable because a population can diffuse neutrally throughout the genotype network and build up genetic diversity [16], which facilitates access to novel phenotypes through non-neutral mutations into adjacent genotype networks [10].

One of the primary advantages of discussing neutrality in terms of robustness is that an exact measure of neutrality can be specifically assigned to each genotype and phenotype. This in turn allows for an assessment of the distribution of neutrality at both the genotypic and phenotypic scales. Further, it allows for a quantitative analysis of the relationship between robustness and evolvability. For GP, this offers a marked improvement over previous neutrality analyses, which typically use only performance statistics, such as success rate or search efficiency, to estimate how neutrality affects evolutionary search.

The exact distributions of neutrality, at the level of the genotype and phenotype, have never been explicitly characterized in a GP system. As such, the influence of robustness on evolvability, and in turn on search performance, is not well understood. Further, the ease with which a phenotype can be accessed in an evolutionary search, and how this relates to robustness and evolvability at the genotypic and phenotypic scales, has not been addressed. To elucidate these relationships, we extend the work of [19] by exhaustively characterizing the search space of a simple Linear Genetic Programming (LGP) representation used to solve a Boolean search problem. This system offers several advantages. First, the representation is compact; the set of all genotypes is finite and computationally enumerable. Second, neutrality is intrinsic to the system; a total of 2^{28} genotypes map to 16 phenotypes. Third, there is a clear delineation between genotype, phenotype, and fitness, allowing for a full description of their interplay.

By capitalizing on recent developments in the characterization of robustness and evolvability in RNA [11,20], we provide a comprehensive quantitative analysis of the genotype and phenotype spaces in this LGP system. Using a large ensemble of random walks, we then conduct a preliminary exploration of the relationships between robustness, evolvability, and mutation-based search. We discuss the implications of our results and present directions for future work.

2 Methods

2.1 Linear Genetic Programming

In the LGP representation, an individual (or program) consists of a set of L instructions, which are structurally similar to those found in register machine languages. Each instruction is made up of an operator, a set of operands, and a return value. In the programs considered in this study, each instruction consists of an operator drawn from the set {`AND`, `OR`, `NAND`, `NOR`}, two Boolean operands, and one Boolean return value. The inputs, operands, and return values are stored in registers with varying read/write permissions. Specifically, $\texttt{R}_0$ and $\texttt{R}_1$ are calculation registers that can be read and written, whereas $\texttt{R}_2$ and $\texttt{R}_3$ are input registers that are read-only. Thus, a calculation register can serve in an instruction as an operand or a return, but an input register can only be used as an operand. An example program with $L = 4$ is given below.

$$\begin{aligned}
\texttt{R}_1 &= \texttt{R}_2 \ \ \texttt{OR} \ \ \texttt{R}_3 \\
\texttt{R}_0 &= \texttt{R}_1 \ \ \texttt{AND} \ \ \texttt{R}_2 \\
\texttt{R}_1 &= \texttt{R}_0 \ \ \texttt{NAND} \ \ \texttt{R}_1 \\
\texttt{R}_0 &= \texttt{R}_3 \ \ \texttt{NOR} \ \ \texttt{R}_1
\end{aligned}$$

These instructions are executed sequentially from top to bottom. Prior to program execution, the values of $\texttt{R}_0$ and $\texttt{R}_1$ are initialized to `FALSE`. After program execution, the final value in $\texttt{R}_0$ is returned as output.

2.2 Genotype and Phenotype Networks

To facilitate the enumeration of the entire genotype and phenotype spaces, we consider a two-input, one-output Boolean problem instance with $L = 4$ instructions. This sequence of instructions is referred to as the *genotype*. Letting C and I denote the numbers of calculation and input registers, respectively, and O the cardinality of the operator set, there are a total of $(C \times (C + I)^2 \times O)^L$ genotypes in the LGP representation. We refer to this set of programs as the *genotype space*. In the system considered here ($L = 4, C = 2, I = 2, O = 4$), the genotype space comprises 2^{28} unique programs.

These genotypes map to a considerably smaller set of phenotypes, which are defined by the functional relationship between the input and output registers. Specifically, the *phenotype* is defined by the set of outputs observed across each of the four possible combinations of Boolean inputs. Since the outputs are also

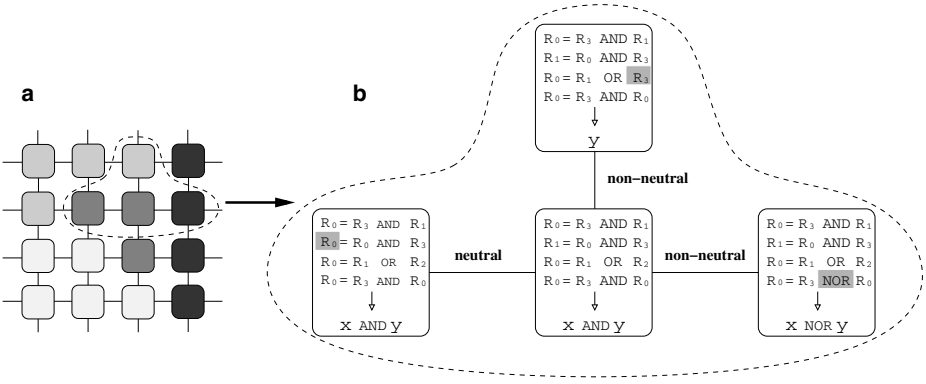

Fig. 1. (a) Schematic diagram of a subset of genotype space in linear genetic programming. Vertices correspond to genotypes, their color to phenotypes, and edges connect genotypes that can be interconverted via point mutations. (b) These point mutations can be neutral or non-neutral, depending on whether the phenotype is preserved. For visual clarity, we only depict a small subset of the potential point mutations to each genotype.

Boolean, there are $2^4 = 16$ unique phenotypes, the set of which we refer to as the *phenotype space*.

The mapping from genotype to phenotype is highly redundant. As previously mentioned, a convenient formalism for understanding this redundancy is a *genotype network*, in which genotypes are represented as vertices and edges connect genotypes that can be interconverted via neutral point mutations. In this study, we define a point mutation as a single change to an element of the instruction set of a program. This point mutation is neutral if it does not lead to a change in phenotype (Fig. 1).

The number of point mutations v_{ij} between the genotype networks of phenotypes i and j captures the number of unique mutational events that interconvert these two phenotypes. By considering the adjacency of all genotype networks in the genotype space, we can construct a phenotype network. Vertices correspond to phenotypes and are weighted according to the size of their underlying genotype network, and edges correspond to the adjacency of genotype networks and are weighted according to the number of non-neutral point mutations between genotype networks (Fig. 2). By characterizing the genotype and phenotype spaces in this way, we can describe the distribution of neutrality in this LGP system and gain insight into how this distribution influences robustness and evolvability.

2.3 Robustness, Evolvability, and Accessibility

Several measures of robustness and evolvability exist in the literature, at both the genotypic and phenotypic scales. Following [11], we define *genotypic robustness* as the fraction of the total number of possible point mutations to a given genotype that are neutral. *Genotypic evolvability* is defined as the fraction of the total

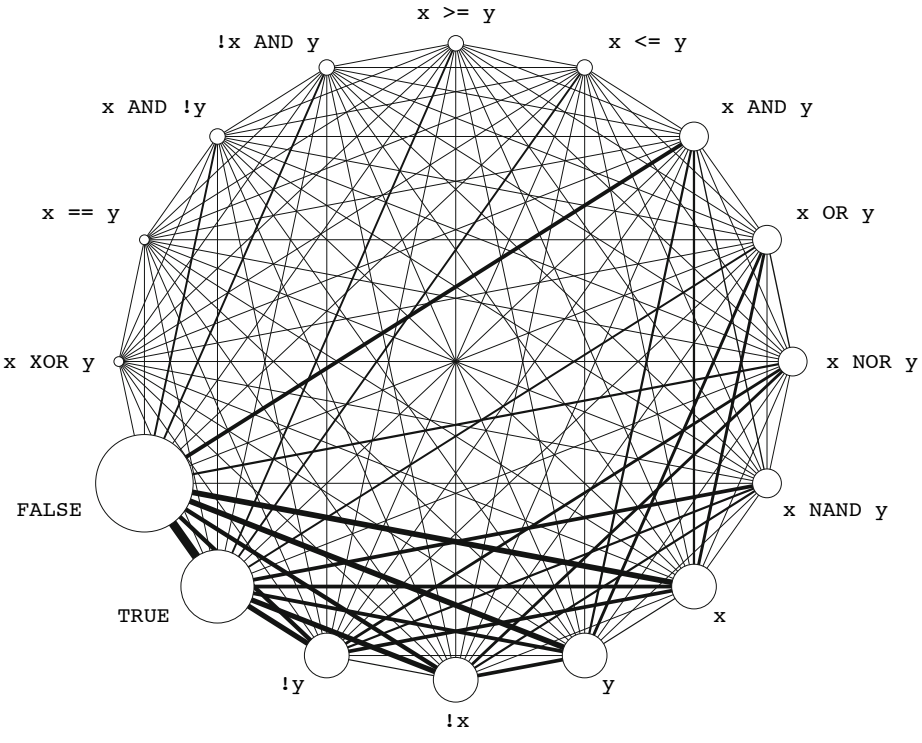

Fig. 2. Phenotype network for linear genetic programming with two inputs, one output, and four instructions. Each vertex comprises a genotype network, as depicted in Fig. 1, and thus vertex size corresponds to phenotypic robustness. Edge width denotes the number of non-neutral point mutations between two phenotypes (v_{ij}). Phenotypes are labeled according to their functional relationship between input and output, where `x` and `y` denote the inputs stored in registers $\mathtt{R}_2$ and $\mathtt{R}_3$, respectively.

number of possible phenotypes that are accessible through non-neutral point mutations to a single genotype. *Phenotypic robustness* is defined as the size of the phenotype's underlying genotype network.

For *phenotypic evolvability*, we consider two metrics. The first E_1 is simply the proportion of the total number of phenotypes that can be reached via non-neutral point mutations from a given phenotype [11]. The second E_2 provides a more nuanced analysis of the potential to mutate from one phenotype to another [20]. Letting

$$f_{ij} = \begin{cases} \frac{v_{ij}}{\sum_{k \neq i} v_{ik}}, & \text{if } i \neq j \\ 0, & \text{if } i = j \end{cases} \tag{1}$$

denote the fraction of non-neutral point mutations to genotypes of phenotype i that result in genotypes of phenotype j, we define the evolvability E_2 of phenotype i as

$$E_{2,i} = 1 - \sum_j f_{ij}^2. \tag{2}$$

This captures the probability that two randomly chosen non-neutral point mutations to genotypes of phenotype i result in genotypes with distinct phenotypes. This metric takes on high values when a phenotype is adjacent to many other phenotypes and its non-neutral mutations are uniformly divided amongst these phenotypes. It takes on low values when a phenotype is adjacent to only a few other phenotypes and its non-neutral mutations are biased toward a subset of these phenotypes.

In addition to measuring the propensity to mutate away from a phenotype, we also measure phenotypic accessibility [20], which is formally defined as

$$A_i = \sum_j f_{ji}. \tag{3}$$

This metric takes on high values if phenotype i is relatively easy to access from other phenotypes, and low values otherwise.

3 Results

3.1 Statistical Characteristics of Genotype and Phenotype Space

To characterize the genotype and phenotype spaces of the two-input, one-output LGP system of $L = 4$ instructions, we exhaustively enumerated the size and structure of the genotype networks corresponding to each of the 2^{28} individual genotypes.

In Fig. 2, we depict the corresponding phenotype network. The network is fully connected, such that any phenotype can be reached directly from any other. However, the number of non-neutral point mutations between phenotypes, depicted by edge width, is heterogeneous. Some phenotypes are mutationally biased toward a small subset of phenotypes (*e.g.*, Fig. 2, `x`), while others mutate nearly uniformly to all other phenotypes (*e.g.*, Fig. 2, `x XOR y`). Phenotypic robustness, depicted by vertex size, is also heterogeneous and ranges from a minimum of 24,832 genotypes to a maximum of 60,393,728 genotypes (occupying $\ll 1\%$ and 23% of genotype space, respectively). Each phenotype comprises a single genotype network, as opposed to multiple independent genotype networks. Thus, any genotype yielding a specific phenotype is reachable through a series of neutral point mutations from any other genotype with the same phenotype.

In Figs. 3a,b, we depict the distributions of genotypic robustness and genotypic evolvability for the representative phenotype `FALSE`. Within this phenotype, and all others in this system, the distribution of genotypic robustness is bimodal (Fig. 3a), while the distribution of genotypic evolvability is unimodal (Fig. 3b). These properties are inversely related, such that genotypes of greater robustness are less evolvable (Fig. 3c).

The means of the distributions of genotypic robustness and evolvability vary as a function of phenotypic robustness. Specifically, average genotypic evolvability

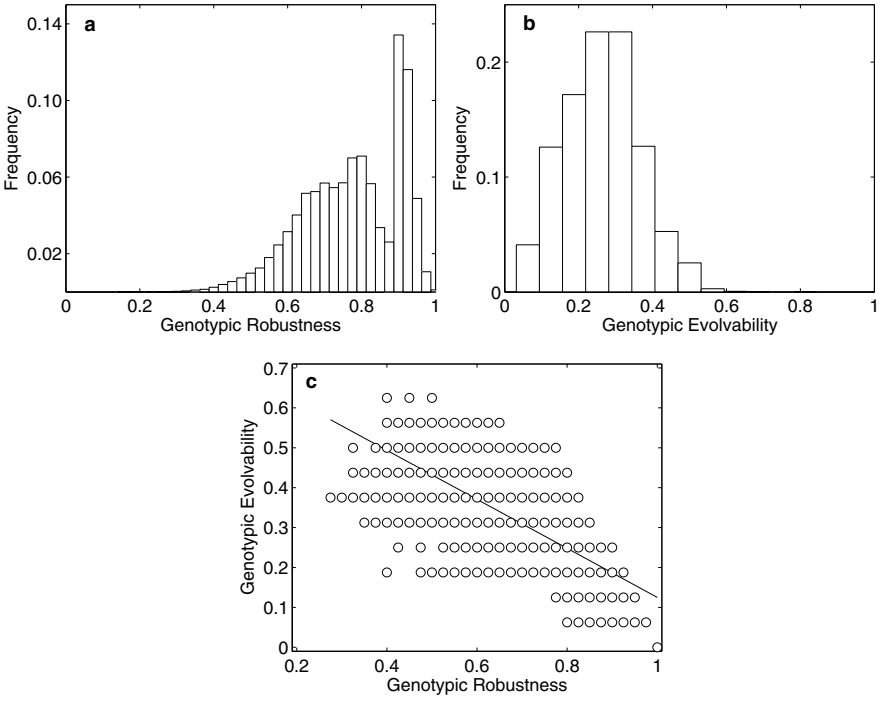

Fig. 3. Properties of genotypes within the phenotype `FALSE`. Distributions of (a) genotypic robustness and (b) genotypic evolvability for all ≈ 60 million genotypes. (c) Relationship between genotypic evolvability and genotypic robustness for 10,000 randomly sampled genotypes. The solid line represents the best linear fit to the data, and is provided as a guide for the eye.

decreases logarithmically as a function of phenotypic robustness (Fig. 4a, $R^2 = 0.95$). This intuitive observation implies that within robust phenotypes, most mutations are neutral and do not allow access to adjacent phenotypes. It follows that the individual genotypes that make up robust phenotypes are collectively more robust. Indeed, we observe that the average genotypic robustness increases logarithmically as a function of phenotypic robustness (Fig. 4b, $R^2 = 0.98$).

The relationship between phenotypic evolvability and phenotypic robustness is less intuitive. Because the phenotype network is fully connected, all phenotypes are equally and maximally evolvable according to E_1 (filled circles, Fig. 4c). In contrast, when mutational biases are taken into account with E_2, phenotypic evolvability exhibits a nonlinear relationship with phenotypic robustness (open circles, Fig. 4c). Phenotypic evolvability is lowest for phenotypes of intermediate robustness (`x AND !y`, `!x AND y`), and then increases logarithmically with increasing phenotypic robustness ($R^2 = 0.87$). The relationship is made non-monotonic by the high evolvability of the least robust phenotypes (`x XOR y`, `x == y`).

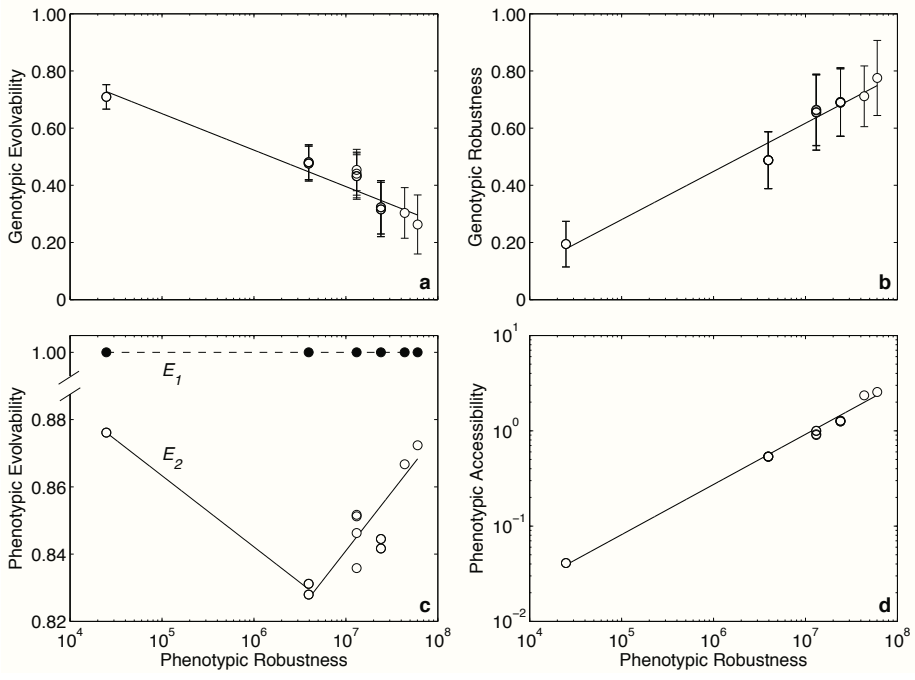

Fig. 4. Statistical properties of genotype and phenotype networks in linear genetic programming. Genotypic (a) evolvability and (b) robustness and phenotypic (c) evolvability and (d) accessibility as a function of phenotypic robustness. The data in (a,b) correspond to the average of all genotypes within a given phenotype and error bars denote their standard deviation. The solid lines correspond to the best (a,b) logarithmic, (c) piecewise logarithmic, and (d) power-law fit to the data, and are provided as a guide for the eye. In (c), two measures of phenotypic evolvability are shown.

Phenotypic accessibility increases monotonically as a function of phenotypic robustness, following a power-law (Fig. 4d, $R^2 = 0.99$). This implies that random mutations are more likely to lead to robust phenotypes than to non-robust phenotypes. Taken together, these results suggest that the most robust phenotypes are both easy to find (Fig. 4d) and highly evolvable (Fig. 4c), with the exception of the least robust phenotype, which is simultaneously the least accessible and the most evolvable of any of the phenotypes in this system.

3.2 Random Walks through Genotype and Phenotype Space

To begin to understand the implications of these observations for mutation-based search, we consider an ensemble of random walks. For each of the 16×16 possible combinations of phenotypes, we designate one phenotype as a source and the other as a target. We then perform 1000 random walks, starting from a randomly chosen genotype in the source phenotype and ending when the random

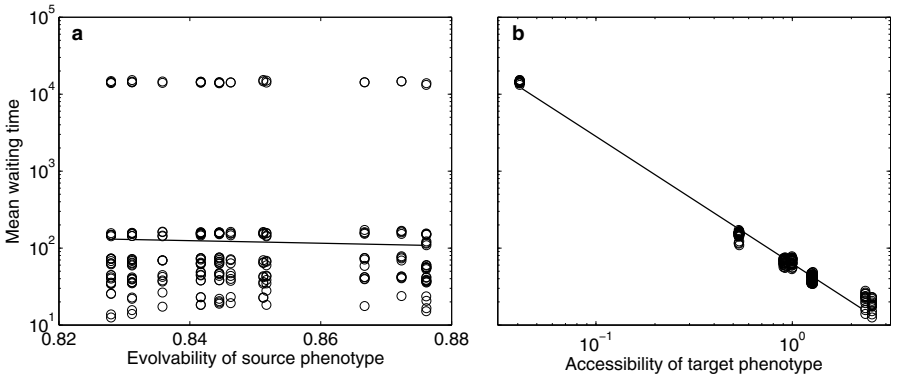

Fig. 5. Mean waiting time of a random walk as a function of (a) the source phenotype's evolvability and (b) the target phenotype's accessibility. The solid lines corresponds to the best (a) exponential and (b) power-law ($T \propto A^{-5/3}$) fit to the data. Both lines are provided as a guide for the eye.

walk reaches any genotype in the target phenotype. Each step in the random walk corresponds to a single point mutation. We record the average number of steps required to get from one phenotype to another, which we refer to as the mean waiting time.

In Fig. 5, we depict the mean waiting time of a random walk as a function of the source phenotype's evolvability (Fig. 5a) and the target phenotype's accessibility (Fig. 5b). While the mean waiting time is independent of the evolvability of the source phenotype ($R^2 = 0.001$), it is strongly correlated with the accessibility of the target phenotype ($R^2 = 0.99$). Specifically, the mean waiting time T decreases as a function of target accessibility according to the power-law $T \propto A^{-5/3}$. Thus, accessible phenotypes are found more rapidly by random mutation than less accessible phenotypes. As highly accessible phenotypes are also highly robust (Fig. 4d), random mutation leads to robust phenotypes.

4 Discussion

The results of this study have helped to clarify how the distributions of neutrality at the genotypic and phenotypic scales affect the relationship between robustness and evolvability in linear genetic programming (LGP). At the genotypic level, robustness and evolvability were found to be negatively correlated with one another (Fig. 3c), echoing previous results regarding RNA landscapes [11]. This intuitive observation suggests that robust genotypes are located far from the periphery of the genotype network, prohibiting direct mutational access to adjacent genotype networks. Further support for this claim could be obtained through a topological analysis of the genotype networks, which could also help to explain the observed bimodality in the distributions of genotypic robustness (Fig. 3a).

Within a given phenotype, the distribution of genotypic robustness was heterogeneous, where some genotypes were vastly more robust than others (Fig. 3a). In addition, all genotypes within a phenotype formed a single genotype network, such that any genotype was reachable from any other through a series of neutral point mutations. Taken together, these results suggest that genotypic robustness is an evolvable property in LGP. A caveat of this observation is that any selective pressure for genotypic robustness comes at the expense of reduced genotypic evolvability (Fig. 3c).

At the phenotypic level, the relationship between robustness and evolvability varied depending on how evolvability was defined (Fig. 4c). When defined by the total connectivity of a phenotype in the phenotype network (E_1), evolvability was independent of robustness; all phenotypes were maximally evolvable. In contrast, when mutational biases were taken into account (E_2), the relationship between evolvability and robustness was nonlinear, with phenotypes of intermediate robustness exhibiting the lowest evolvabilities. These results contrast with those made in RNA systems where E_1 was found to be positively correlated [11], and E_2 negatively correlated, with robustness [20]. However, accessibility and robustness were positively correlated (Fig. 4d), in line with observations made in RNA systems and supporting the intuitive notion that larger phenotypes are easier to access.

To explore the implications of these observations for evolutionary search, we considered an ensemble of random walks between source and target phenotypes. The mean waiting time of random mutation to reach a target phenotype was found to be uncorrelated with the evolvability of the source phenotype (Fig. 5a), a result that calls into question the utility of existing phenotypic evolvability measures. While these measures provide useful information concerning the immediate adjacency of phenotypes [11] and their mutational biases [20], they are too myopic to predict the length of an evolutionary trajectory from one phenotype to another. Consider, for example, that correlations may exist between the evolvabilities of adjacent phenotypes, such that high evolvability phenotypes are mutationally biased toward low evolvability phenotypes. As these correlations (a.k.a. mixing patterns [21]) are not taken into account, the applicability of current phenotypic evolvability measures are left severely constrained, at least for this system.

In contrast, the mean waiting time of random mutation to reach a target phenotype was strongly correlated with the target phenotype's accessibility (Fig. 5b). Our observations suggest that this relationship follows a power-law of the form $T \propto A^{-5/3}$. We believe this relationship may lend itself to analytical treatment, which could prove useful in generalizing our results to other phenotype networks. For instance, it remains to be seen whether target accessibility is predictive of mean waiting time in phenotype networks that are not fully connected. Recent analyses of random walks on weighted complex networks [22] may provide a useful starting point for this analysis.

There has been much debate regarding the benefit of neutrality in evolutionary computation. Using a simple set of experiments, recent analysis has

demonstrated this benefit to be problem dependent [9]. Our results provide a more subtle picture. Using the notion of robustness to compartmentalize neutrality into specific phenotypes, we have shown that it is not only the presence of neutrality that affects mutation-based search, but also how this neutrality is distributed amongst phenotypes. The more robust the target phenotype, the easier it will be for evolution to identify it.

To fully understand how robustness, evolvability, and accessibility influence evolutionary search, future analyses will have to take fitness into consideration. In the Boolean search problem considered herein, for example, fitness could be assigned as a function of phenotypic distance from the target [19]. As only five distinct fitness values exist in our LGP system, this form of fitness assignment would add a new layer of robustness and evolvability to the analysis. Specifically, this layer would consist of a set of vertices that each comprise multiple phenotypes, and links between these vertices would capture all mutational events that lead from one fitness value to another. While fitnesss would therefore modify the transition probabilities between phenotypes, selection would modify the transition probabilities between fitness values. Understanding how these layers interact to influence evolutionary search is left for future research.

Future work will also consider population-based evolutionary search and the role of recombination. We are particularly interested in analyzing population diffusions throughout genotype and phenotype networks [23], under mutational and recombinative variation operators, to understand how robustness and evolvability change as a function of phenotypic distance to the target.

Acknowledgments

This work was partially supported by NIH grants R01-LM009012, R01-LM010098, and R01-AI59694. J.L.P. was supported by NIH grant K25-CA134286. W.B. acknowledges support from NSERC Discovery Grants, under RGPIN 283304-07.

References

1. Banzhaf, W.: Genotype-phenotype mapping and neutral variation - a case study in genetic programming. In: Davidor, Y., Männer, R., Schwefel, H.-P. (eds.) PPSN 1994. LNCS, vol. 866, pp. 322–332. Springer, Heidelberg (1994)
2. Smith, T., Husbands, P., O'Shea, M.: Neutral networks and evolvability with complex genotype-phenotype mapping. In: Kelemen, J., Sosík, P. (eds.) ECAL 2001. LNCS (LNAI), vol. 2159, pp. 272–282. Springer, Heidelberg (2001)
3. Smith, T., Husbands, P., O'Shea, M.: Neutral networks in an evolutionary robotics search space. In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 136–145 (2001)
4. Collins, M.: Finding needles in haystacks is harder with neutrality. Genetic Programming and Evolvable Machines 7, 131–144 (2006)
5. Yu, T., Miller, J.F.: Through the interaction of neutral and adaptive mutations, evolutionary search finds a way. Artificial Life 12, 525–551 (2006)

6. Soule, T.: Resilient individuals improve evolutionary search. Artificial Life 12, 17–34 (2006)
7. Harvey, I., Thompson, A.: Through the labyrinth evolution finds a way: A silicon ridge. In: Proceedings of the First International Conference on Evolvable Systems: From Biology to Hardware, pp. 406–422 (1996)
8. Ebner, M., Shackleton, M., Shipman, R.: How neutral networks influence evolvability. Complexity 7(2), 19–33 (2002)
9. Galván-López, E., Poli, R.: An empirical investigation of how and why neutrality affects evolutionary search. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 1149–1156 (2006)
10. Wagner, A.: Neutralism and selectionism: a network-based reconciliation. Nature Reviews Genetics 9, 965–974 (2008)
11. Wagner, A.: Robustness and evolvability: a paradox resolved. Proceedings of the Royal Society London B 275, 91–100 (2008)
12. Bloom, J.D., Labthavikul, S.T., Otey, C.R., Arnold, F.H.: Protein stability promotes evolvability. Proceedings of the National Academy of the Sciences 103(15), 5869–5874 (2006)
13. Ferrada, E., Wagner, A.: Protein robustness promotes evolutionary innovations on large evolutionary time-scales. Proceedings of the Royal Society London B 275, 1595–1602 (2008)
14. Isalan, M., Lemerle, C., Michalodimitrakis, K., Horn, C., Beltrao, P., Raineri, E., Garriga-Canut, M., Serrano, L.: Evolvability and hierarchy in rewired bacterial gene networks. Nature 452, 840–846 (2008)
15. Draghi, J.A., Parsons, T.L., Wagner, G.P., Plotkin, J.B.: Mutational robustness can facilitate adaptation. Nature 463, 353–355 (2010)
16. Huynen, M.A., Stadler, P.F., Fontana, W.: Smoothness within ruggedness: The role of neutrality in adaptation. Proceedings of the National Academy of the Sciences 93, 397–401 (1996)
17. Newman, M., Engelhardt, R.: Effects of selective neutrality on the evolution of molecular species. Proceedings of the Royal Society London B 265, 1333–1338 (1998)
18. Whitacre, J., Bender, A.: Degeneracy: a design principle for achieving robustness and evolvability. Journal of Theoretical Biology 263, 143–153 (2010)
19. Banzhaf, W., Leier, A.: Genetic Programming Theory and Practice III. In: Genetic Programming Theory and Practice III, pp. 207–221. Springer, Heidelberg (2006)
20. Cowperthwaite, M.C., Economo, E.P., Harcombe, W.R., Miller, E.L., Meyers, L.A.: The ascent of the abundant: how mutational networks constrain evolution. PLoS Computational Biology 4(7), e10000110 (2008)
21. Newman, M.: Mixing patterns in networks. Physical Review E 67, 026126 (2003)
22. Noh, J.D., Rieger, H.: Random walks on complex networks. Physical Review Letters 92(11), 118701 (2004)
23. van Nimwegen, E., Crutchfield, J., Huynen, M.: Neutral evolution of mutational robustness. Proceedings of the National Academy of the Sciences 96, 9716–9720 (1999)