Quantitative Analysis of Evolvability using Vertex Centralities in Phenotype Network

Ting Hu
Department of Computer Science
Memorial University
St. John's, Canada
ting.hu@mun.ca

Wolfgang Banzhaf
Department of Computer Science
Memorial University
St. John's, Canada
banzhaf@mun.ca

ABSTRACT

In an evolutionary system, robustness describes the resilience to mutational and environmental changes, whereas evolvability captures the capability of generating novel and adaptive phenotypes. The research literature has not seen an effective quantification of phenotypic evolvability able to predict the evolutionary potential of the search for novel phenotypes. In this study, we propose to characterize the mutational potential among different phenotypes using the phenotype network, where vertices are phenotypes and edges represent mutational connections between them. In the framework of such a network, we quantitatively analyze the evolvability of phenotypes by exploring a number of vertex centrality measures commonly used in complex networks. In our simulation studies we use a Linear Genetic Programming system and a population of random walkers. Our results suggest that the weighted eigenvector centrality serves as the best estimator of phenotypic evolvability.

Keywords

Evolvability; Robustness; Neutrality; Phenotype network; Vertex centrality; Linear Genetic Programming; Genotype network; Neutral network; Redundant genotype-to-phenotype mapping; Representation

1. INTRODUCTION

The representation schemes in both natural and computational evolutionary systems often possess high redundancy, i.e. multiple genotypes can map to the same phenotype [13, 15, 23, 26, 27, 28, 29, 35, 42]. The pervasive existence of such a redundant genotype-to-phenotype mapping, especially in high-level living organisms, suggests its functional contribution to adaptive evolution [6, 17, 20, 50].

A redundant genotype-to-phenotype mapping is common in evolutionary algorithms and allows evolutionary populations to expand in neutral genotypic regions where mutations to a genotype do not alter the phenotypic outcome. Neutral genetic variations by mutation possess the potential for creating novel phenotypes [26, 46]. They serve as a staging ground for long-term adaptation and innovation. Such neutrality provides a buffer against deleterious mutational perturbations, and accumulates genetic variations that might be non-neutral under changes of the environmental context [9, 10, 23, 27, 45].

Robustness [24, 44] and evolvability [22, 33, 46] are often discussed to describe the two sides of neutrality. Essentially, both properties reflect how evolutionary systems respond to changes. Robustness enables them to remain intact in the face of deleterious changes, whereas evolvability allows them to innovate to better fit the survival pressures of the environment. At a first glance, these two properties may seem contradictory, however, numerous theoretical and empirical studies have reported that robustness and evolvability cooperate [24, 49], and robustness, in fact, can facilitate evolvability [6, 8, 9, 10, 11].

Quantitative analysis of robustness and evolvability should help to better understand the mechanisms of neutrality, and recently many research studies in the fields of evolutionary biology [7, 30, 48] and evolutionary computing [13, 19, 36, 37, 51] have addressed this issue. It has been proposed that the relationship of robustness and evolvability crucially depends on the distribution of genotypic redundancy and the mutational interconnections among phenotypes. Robustness promotes high evolvability if genotypic redundancy leads to more connections to different phenotypes.

Network notions have often been used to model how genotypic redundancy is distributed and how different phenotypes are connected through mutational changes to their underlying genotypes. Genotype networks, a.k.a. neutral networks, have been demonstrated to be very useful vehicles for quantitative studies of robustness and evolvability [34, 39, 47]. In genotype networks, genotypes are represented as vertices, and reversible mutational connections, as in common evolutionary systems, are represented as undirected edges between pairs of genotypes. A genotype network is comprised of all genotypes that encode for the same phenotype. A phenotype network can be constructed by representing each vertex as a phenotype. Phenotypes are connected through non-neutral point mutations between their underlying genotypes.

In the framework of genotype and phenotype networks, evolvability can then be quantitatively characterized. *Genotypic evolvability* is often measured as the total number of different phenotypes that a genotype can access through single-step mutations [48]. *Phenotypic evolvability* can be

GECCO '16, July 20-24, 2016, Denver, CO, USA

DOI: http://dx.doi.org/10.1145/2908812.2908940

measured as the total number of possible phenotypes that are adjacent to a given phenotype (i.e. via phenotypically-non-neutral single-point mutations to its underlying genotypes) [48] or as the diversity of a phenotype's mutational connections to other phenotypes [7]. However, both phenotypic evolvability measures have been shown to have shortcomings at predicting the capability and efficiency of a phenotype finding other novel phenotypes through point mutations [18]. The existing phenotypic evolvability measures only describe the immediate hops accessing other phenotypes but fail to predict evolutionary trajectories based on the global structure of the genotype and phenotype networks.

In this study, we propose to use network centrality measures to quantify phenotypic evolvability in the context of the phenotype network. Vertex centralities are widely used in complex and social network analysis to capture the importance of an individual vertex in the global network structure. They serve as promising candidates for quantifying the evolvability of a phenotype in the phenotype network. We adopt a Boolean Linear Genetic Programming (LGP) algorithm as our evolutionary model system, and construct its phenotype network by sampling the mutational connections among phenotypes. We explore a number of centrality measures in this network and test their predictive power with respect to the evolvability of phenotypes. Using simulations of a population of random walkers, we observe that weighted eigenvector centrality outperforms existing evolvability measures and serves as the best quantification of phenotypic evolvability among all the studied centrality measures.

2. METHODS

2.1 Linear Genetic Programming System

We use a Linear Genetic Programming (LGP) algorithm as our artificial evolutionary system for the quantitative study of phenotypic evolvability. LGP is a branch of Genetic Programming (GP), in the big family of Evolutionary Computation (EC), where the chromosomal representation is a set of instructions that are executed sequentially [5]. Although LGP follows a linear instructional structure, it is very powerful and capable of modeling complex nonlinear relationships among multiple attributes. LGP is one of the three generic representatives, i.e., tree, linear, and graph, of Genetic Programming [1] and has gained increasing popularity due to its fast speed of program execution and individual evaluation [4, 14, 41].

In this study, we consider a three-input, one-output Boolean function modeling problem. Each LGP instruction is comprised of one return, two operands, and one Boolean operator producing the return value from the operands. Registers $\texttt{R}_1$, $\texttt{R}_2$, and $\texttt{R}_3$ store the three Boolean input values. Register $\texttt{R}_0$ takes a default initial Boolean value and its final value after the execution of all instructions is returned as the LGP program's output. To enhance the computational capacity of LGP programs, we add an extra calculation register $\texttt{R}_4$. Calculation registers $\texttt{R}_0$ and $\texttt{R}_4$ can serve as both return or operands, whereas input registers $\texttt{R}_1$, $\texttt{R}_2$, and $\texttt{R}_3$ are read-only and can only serve as operands such that their input contents are protected from overwriting. The Boolean operator in each LGP instruction is chosen from a pre-defined operator set *opr*. An example Boolean LGP program with a length $L = 3$ is given below.

$$\begin{aligned} \texttt{R}_4 &= \texttt{R}_2 \quad \texttt{AND} \quad \texttt{R}_3 \\ \texttt{R}_0 &= \texttt{R}_1 \quad \texttt{OR} \quad \texttt{R}_4 \\ \texttt{R}_0 &= \texttt{R}_3 \quad \texttt{NAND} \quad \texttt{R}_0 \end{aligned}$$

A single-point mutation alters one of the four element/locus of an instruction in a LGP program. A mutation point will be chosen and a replacing allele will be decided randomly and uniformly from the set of all possible alleles at each locus.

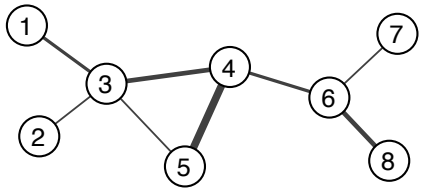

Figure 1: Schematic graph of a phenotype network. Vertices represent phenotypes. An edge connects two phenotypes if they have at least one pair of genotypes that can be traversed from one to the other through one single-point mutation (a single change to any of the four elements of an LGP instruction). Edges can have varying weights, indicating the total number of single-point mutations that can change genotypes of one phenotype to the other.

2.2 Genotype and Phenotype

We consider each LGP program with a unique representation as a *genotype*. Since the two calculation registers can be used as return and all five registers can serve as operands, the total number of unique instructions is $2 \times 5 \times 5 \times |opr|$, where $|opr|$ is the number of Boolean operators in the operator set *opr*. The total number of possible genotypes is thus calculated as $\sum_i (50 \times |opr|)^{L_i}$, i.e. the sum of all possible combinations of L_i instructions. This number increases exponentially with the program length L and quickly reaches large magnitudes with common LGP program length settings.

For our LGP system, we define the three-input, one-output Boolean function $f : \mathbf{B}^3 \rightarrow \mathbf{B}$, where $\mathbf{B} = \{\texttt{TRUE}, \texttt{FALSE}\}$, represented by a LGP program as its *phenotype*. The total number of possible phenotypes is thus $2^{2^3} = 256$.

Similar to many GP and EA algorithms, our LGP system has a highly redundant genotype-to-phenotype mapping, i.e. there are many unique genotypes which map to the same phenotype. We define the *genotypic redundancy* of a phenotype as the total number of its underlying genotypes. A phenotype with a high genotypic redundancy is considered over-represented, and a phenotype with a low genotypic redundancy is under-represented [37].

2.3 Phenotype Network

Network methods are powerful tools in modeling entities and their complex relationships, and have seen numerous applications in a variety of areas including engineering, social sciences, and biology [32]. We use the notion of a *phenotype network* to depict the distribution of mutational connections among phenotypes. Those mutational connections

reflect the potential for moving from one phenotype to another. Thus the phenotype network can provide important insights into the dynamics of a search for novel phenotypes in an evolutionary system.

Figure 1 shows a schematic representation of a phenotype network. Each vertex is an unique phenotype, and two phenotypes are connected by an edge if they have at least one pair of genotypes that can be traversed from one to the other through a single-point mutation. The amount of all possible single-point mutations from one phenotype to another can be captured using the weight of an edge.

With compact evolutionary systems where all genotypes and phenotypes can be efficiently characterized, a phenotype network can be constructed using exhaustive enumeration of all mutational transitions among genotypes and phenotypes. However, for common GP or EC instances, the genotype and phenotype spaces are either infinite or simply too large to apply exhaustive enumeration.

In such cases, the sampling of random mutations can be used to estimate the genotypic redundancy of phenotypes and the mutational connections among them. Random sampling of a large number of genotypes can generate a good estimate of the distribution of genotypic redundancies among different phenotypes. In addition, allowing a large population of random walkers to explore the genotype and phenotype spaces and tracking every step of their single-point mutational trajectories can be used to sample the mutational connectivity among phenotypes.

2.4 Vertex Centralities

A phenotype network provides a global picture of how phenotypes are connected through single-point mutations. In such a framework, the *evolvability* of a phenotype, i.e. the innovative capability of finding novel phenotypes, can be quantitatively captured by the importance measures of individual vertices in the network.

When studying a network, it is often useful to measure the contribution of individual vertices to the network. There have been a number of such metrics from social network analysis. We introduce a number of such *centrality* measurements, and explore their applicability for quantifying phenotypic evolvability. Consider a network $G = (V, E)$ and its adjacency matrix $\mathbf{A}$.

1. **Degree centrality** — The degree centrality of a vertex is simply its degree. Intuitively, the more neighbors a vertex has, the more influence it may have in the network. This assumes that all neighbors are considered equally important. Formally, if we use $\mathbf{1}$ to denote the column vector $\{1, 1, 1, \ldots, 1\}$, the degree centrality of the vertices in the graph is the vector $\mathbf{A} \cdot \mathbf{1}$.

2. **Betweenness centrality** — The betweenness centrality quantifies the number of times a vertex v is part of the shortest path between any pair of vertices [12], represented as $\sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$, where σ_{st} is the total number of shortest paths from vertex s to vertex t and $\sigma_{st}(v)$ is the number of those paths that pass through vertex v. Betweenness captures how important a given vertex is for the connectivity of all other pairs of vertices.

3. **Closeness centrality** — The closeness centrality is denoted as $\frac{1}{\sum_{j \neq i} d_{ij}}$ of a vertex i, where d_{ij} is the distance, i.e. the shortest path, between vertices i and j [2, 38]. Closeness centrality describes how easily a given vertex can reach all other vertices. A higher closeness centrality indicates a more central position of a vertex in the network.

4. **Eigenvector centrality** — It would make sense to give greater weight to a more important neighbor when calculating the centrality of a vertex. Specifically, the centrality of a vertex is proportional to the sum of the centrality of its neighbors. Hence, if the centrality of the vertices of the network is denoted as a positive real column vector $\mathbf{x}$, it should satisfy

$$\mathbf{A}\mathbf{x} = c\mathbf{x},$$

for some constant c [3, 21]. That is, the relative values of the centrality across the network do not change after the incorporation of the neighbors' centrality. It turns out that this eigenvector centrality is always proportional to the leading eigenvector of $\mathbf{A}$. Note that the eigenvector centrality can also be defined iteratively from an "initial guess". In this case, after a sufficient number of iterations, the centrality thus defined always converges to the same leading eigenvector of $\mathbf{A}$. As a result, eigenvector centrality of a vertex is large either because it has a large number of neighbors or because it has "important" neighbors, or both.

The above centrality measures are discussed in the context of undirected networks. They can be applied to both unweighted and weighted networks. Their applications to weighted networks with slight modifications can be found in [25, 31].

3. RESULTS

In this section, we first discuss the genotype and phenotype spaces of our example LGP system. Then we present the phenotype network of the LGP system and characterize its network properties. Last, we explore various vertex centrality measures for the quantification of phenotypic evolvability, and validate them through predictions of the mean waiting time searching for a predefined target phenotype using an ensemble of simulated random walks.

3.1 Genotype and Phenotype Space

In our simulation, programs of variable lengths are considered, from a minimum of two instructions to a maximum of five. The operator set *opr* includes four fundamental Boolean functions `AND`, `OR`, `NAND`, and `NOR`. Table 1 shows the detailed configuration of our LGP system. In our configuration, there are 200 unique LGP instructions,

Table 1: LGP simulation configuration

Program length	$2 \sim 5$ instructions
Input registers	$\mathtt{R}_1$, $\mathtt{R}_2$, $\mathtt{R}_3$
Initial default register value	`FALSE`
Calculation registers	$\mathtt{R}_0$(output), $\mathtt{R}_4$
Operator set (*opr*)	`AND`, `OR`, `NAND`, `NOR`
Total genotype samples	one billion
Ensemble of random walkers	one million
Number of random walk steps	one thousand

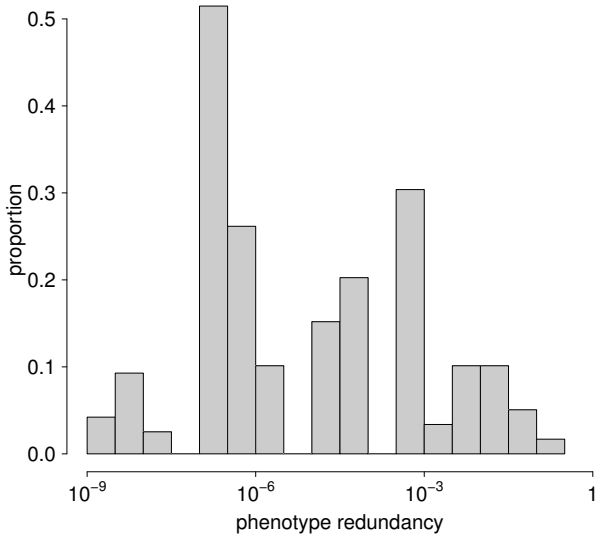

Figure 2: Distribution of phenotype's redundancies. The genotypic redundancy of a phenotype is the percentage of the total one billion sampled genotypes that map to it.

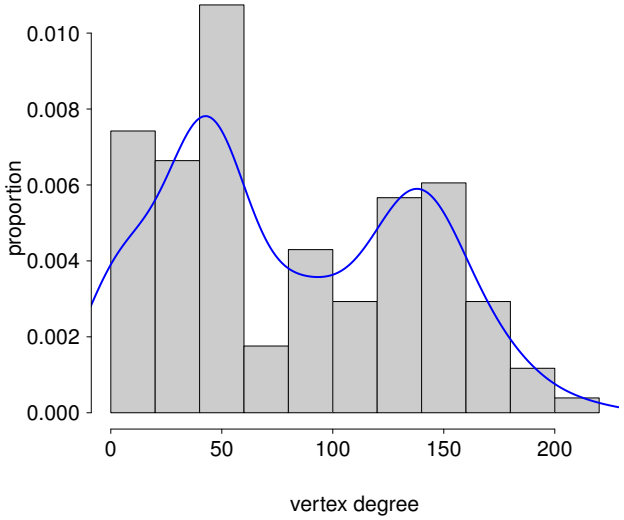

Figure 3: Distribution of vertex degree in the phenotype network. Histograms show the proportions of vertices in the network and the blue curve represents the best fitting density function.

Table 2: Phenotype network properties

Number of non-isolated vertices	236
Number of edges	10353
Connected components	1
Network diameter	4
Network centralization	0.546
Average vertex degree	87.737
Clustering coefficient	0.717

thus the total number of genotypes of our LGP system is $200^2 + 200^3 + 200^4 + 200^5 > 3.2 \times 10^{11}$, a highly redundant mapping to 256 phenotypes. Although this genotype space is finite, such a size renders exhaustive enumeration computationally infeasible.

One billion genotypes are randomly generated to sample the genotype space. Their phenotypes are computed such that the genotypic redundancy, i.e., the percentage of total underlying genotypes, of each phenotype can be estimated. Figure 2 shows the distribution of phenotype's redundancies. It is a highly heterogenous distribution with the most redundant phenotype 255 (numbered using the decimal value of its 8-bit binary output string for the permutation of all eight possible three-bit inputs) having 23% of the total one billion sampled genotypes and nine phenotypes having zero samples. Such heterogeneity of genotypic redundancy demonstrates that an evolutionary search for desired phenotypes crucially depends on which phenotypes are used as starting nodes and which phenotype are chosen as the target node. It also suggests the limitation of purely statistical considerations, as a number of phenotypes are plainly invisible.

3.2 Boolean LGP Phenotype Network

To estimate the mutational connectivity among various phenotypes, we use an ensemble of one million randomly initialized LGP programs and allow them to take one-thousand-step random walks in the genotype space. For each random walker, we record its visit to different phenotypes at any single step along its trajectory, and use the trajectories of all random walkers to estimate the mutational connectivity between any pair of different phenotypes.

Then we construct the approximate phenotype network of our LGP system with vertices representing 256 different phenotypes and edges connecting pairs of them if there exist at least one single-point mutation that can transfer genotypes from one phenotype to another. Edges are weighted using the total number of such single-point mutations.

Table 2 describes the properties of the Boolean LGP phenotype network. There are 236 non-isolated vertices, meaning that 20, i.e. $256 - 236$, phenotypes are never visited by any random walker. The network is comprised of only one giant component and is highly connected with a total number of 10353 edges. The network has a diameter of four, suggesting that any pair of the 236 phenotypes can be traversed from one to the other by at most four single-point mutations. The metric of network centralization measures relative differences among the centralities of all vertices in a network. An extreme value of zero suggests that all vertices in a network are equally central, whereas a value of one suggests that one vertex has maximal centrality and all others have minimal centrality. Our phenotype network has a centralization of 0.546, suggesting that there are many vertices that are more central than the rest. On average, in the network each vertex has 87.737 directly connected neighbors. The network also exhibits a "small world" property with a clustering coefficient 0.717, meaning that for most vertices, their neighbors are also connected.

The vertex degree distribution of the LGP phenotype network is shown in Figure 3. The degree distribution possesses a bi-modal shape, indicating that the network has a two-ring structure with a dense core and a thick peripheral.

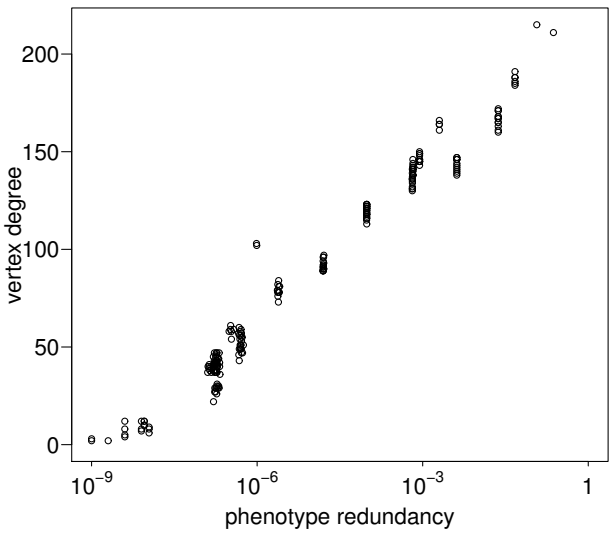

Figure 4: Vertex degree in relation to corresponding phenotype's genotypic redundancy. Each data point represents one phenotype and shows the relation of its degree in the phenotype network and its genotypic redundancy.

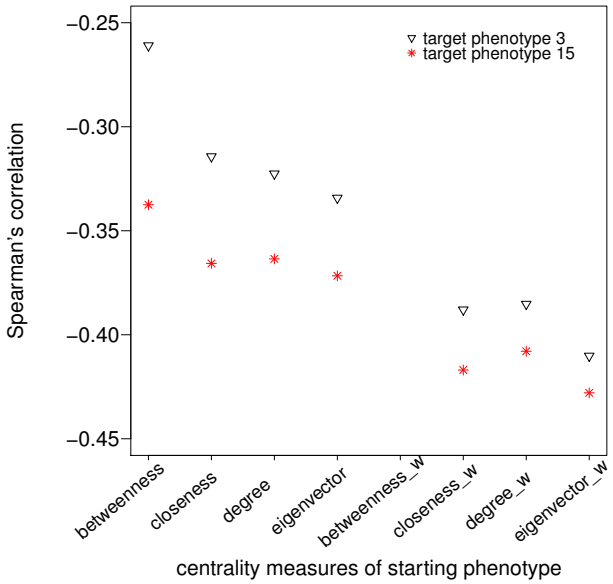

Figure 5: Spearman's rank correlation of mean waiting time finding a target and the centrality measures of the starting phenotypes. The notation "_w" means the application of centrality measures in weighted-network scenarios. Random walkers are genotypes randomly initialized and are classified based on their phenotypes. The total number of steps that a random walker takes to reach the target phenotype is recorded as its waiting time. The mean waiting time of a phenotype is the average waiting time of all its sampled underlying genotypes in the initial population.

Figure 4 shows the correlation of vertex degree and phenotype redundancy. Each data point represents one phenotype, and it depicts its vertex degree in the phenotype network and its genotypic redundancy. There is a strong and positive correlation (Spearman's rank correlation $\rho = 0.9799$, $p < 10^{-16}$), which suggests that, in general, phenotypes comprised of more genotypes tend to be able to reach more other phenotypes through single-point mutations.

3.3 Quantification of Phenotypic Evolvability

Using the framework of phenotype network, we compute vertex centrality measures for all phenotypes, including betweenness, closeness, degree, and eigenvector centralities in both unweighed and weighted scenarios, and test them as quantification measures of how evolvable a phenotype is. The Cytoscape [40] software and its embedded extension app CytoCNA [43] are used for centrality computation.

We set a fixed target phenotype and use a population of ten thousand randomly generated LGP programs to see how many steps it does take for genotypes from a specific phenotype to reach the target phenotype. Only single-point mutations are allowed and the maximum evolution time is limited to one million steps. The mean waiting time of phenotype p finding the target is calculated as $\frac{\sum_{\psi(g_i)=p} t(g_i)}{|g_i|_{\psi(g_i)=p}}$, where ψ maps a genotype to a phenotype, $\{g_i|\psi(g_i) = p\}$ represents all the genotypes of phenotype p, and $t(g_i)$ is the number of steps that a genotype g_i takes to find the target phenotype. The initial population is comprised of randomly generated LGP programs belonging to various phenotypes. A mean waiting time finding the target can be estimated for each of those phenotypes. The hypothesis is that the more evolvable the starting phenotype, the less time it will take to find the target. Note, however, that there is no fitness currently guiding the path formation.

We conduct two sets of evolution simulations with two different target phenotypes, namely phenotype 3 (genotypic redundancy 0.023, vertex degree 163) and phenotype 15 (genotypic redundancy 0.047, vertex degree 188). In both simulation sets, the mean waiting time of all starting phenotypes is calculated, and is used to evaluate how well the centrality measures can predict the evolvability of phenotypes. We perform Spearman's rank correlation test on the mean waiting time and centralities of each starting phenotype. Correlation coefficients ρ are shown in Figure 5, where only significant correlations ($p < 0.05$) are reported.

Centrality measurements are found to be negatively correlated with mean waiting time, implying that the more central a starting phenotype is in the phenotype network, the less time it takes to find the target, i.e. the more evolvable it is. The weighted centrality measures in general have better prediction power on how evolvable a phenotype is, except for the weighted betweenness centrality where no significant correlations are seen for both target settings. The best predicting measure turns out to be the weighted eigenvector centrality. The correlation of weighted eigenvector centrality and mean waiting time is highly significant ($p = 8.77 \times 10^{-5}$ for scenario using phenotype 3 as target, and $p = 2.092 \times 10^{-5}$ for using phenotype 15 as target).

In addition, weighted eigenvector centrality is positively correlated with the genotypic redundancy of the correspond-

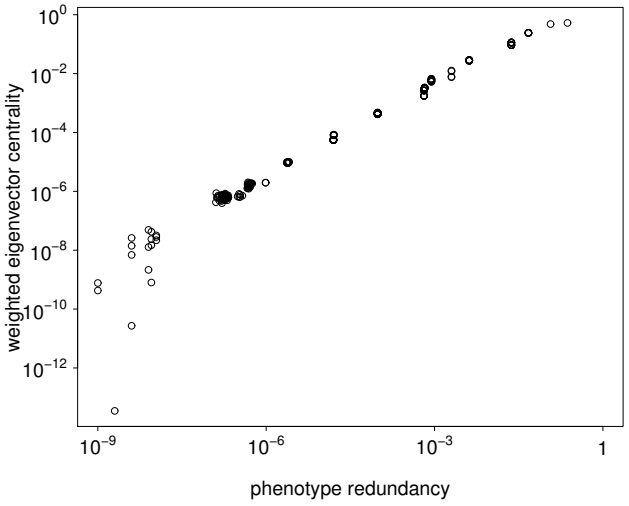

Figure 6: Weighted eigenvector centrality in relation to phenotype redundancy. Each data point is one phenotype and shows the weighted eigenvector centrality and the genotypic redundancy of the corresponding phenotype. A log-log scale is shown based on the best least-square regression fit.

ing phenotype. As shown in Figure 6, these two metrics have a power-law relationship, meaning that the best least-square regression fit (Pearson's correlation R^2=0.966, $p \ll 10^{-10}$) is with a log-log scale. This observation implies that in our LGP system, phenotypes with more underlying genotypes tend to take more central positions in the phenotype network. Although such a positive power-law correlation exists between a phenotype's genotypic redundancy and the best evolvability estimator, i.e. weighted eigenvector centrality, the redundancy itself does not serve as the best predictor on the mean waiting time (Spearman's rank correlation $\rho = -0.353$ and $\rho = -0.372$ for the two target phenotype settings respectively).

4. DISCUSSION

The redundancy in the mapping from genotype to phenotype is hypothesized as a mechanism to have resulted from adaptive evolution itself. Such a redundant mapping enables neutrality where mutations can appear neutral and do not alter the phenotypic outcome. Neutrality improves the robustness of evolutionary systems against random perturbations, as well as aggregates genetic variations that make evolutionary systems more evolvable generating novel adaptive phenotypes. Quantitative analysis of robustness and evolvability helps to elucidate the complex relationship of these two, and to understand core mechanisms of evolution.

Robustness and evolvability can correlate very differently at the genotypic and phenotypic levels, and thus their quantitative analysis should be separated as well. Quantitative measurements of genotypic and phenotypic robustness and evolvability have been proposed in empirical studies. However, phenotypic evolvability quantification using the vertex degree in the phenotype network [48] can be very limited for predicting the long-term evolutionary trajectory.

In this contribution, we propose to measure evolvability of phenotypes using vertex centralities in the phenotype network. We adopted a three-input, one-output Boolean LGP system as our evolutionary model system, and characterized its genotype and phenotype spaces using random sampling and random walks. We constructed the phenotype network and explored a number vertex centrality measures, commonly used in social network analysis, as quantification of phenotypic evolvability. The results of our simulation studies suggest that more sophisticated centrality measures, which consider the importance of a vertex on not only how many neighbors it has but also how important those neighbors are, better predict the long-term evolutionary capabilities of phenotypes. Among them, the weighted eigenvector centrality serves as the best quantification of phenotypic evolvability.

The predictive power of even the best centrality measure is still not strong (correlation coefficient $-0.5 < \rho < -0.4$). This suggests that there might be other confounding elements that can affect the search for novel phenotypes from a given starting phenotype. While phenotypes are compared and evaluated for adaptivity, genetic changes like point mutations applied here occur at the genotypic level. We have also seen that the distribution of genotypic mutational potential within phenotypes is highly heterogeneous. Robust genotypes are visited more frequently by an evolutionary population and their mutational biases can substantially influence phenotypic search [19]. This has not been incorporated in the evolvability measurement, and will be our next research objective.

In future studies, we also would like to include fitness in our evolutionary model. With fitness-based selection considered, single-point mutations will no longer be reversible, and thus the phenotype network becomes directed. As shown in our previous research [19], introducing fitness alters the structure and connectivity of genotype and phenotype networks significantly and invalidates many correlations of genotypic and phenotypic properties. This certainly will add more complexity to the network analysis, but will better simulate the scenarios in living organisms and evolutionary algorithms.

Moreover, the phenotype network is well connected in our example LGP system, and it is interesting to analyze other problem instances with less a connectivity, meaning that some phenotypes may be very difficult to reach through mutations. We expect that the centrality measures may provide an improved prediction power in such a scenario.

Further inquiries are also necessary to make centrality measures applicable in real search processes where the genotype and phenotype networks are constantly increasing in size and where iterative methods might be required to calculate approximate evolvability measures.

Last, most robustness and evolvability studies focus on point mutation, however, other forms of genetic changes such as recombination and gene duplication could play an important role linking genotypes as well. We have conducted preliminary investigations on recombinational robustness and evolvability [16, 17], and expect to include more analysis on recombination or gene duplication in future studies.

This line of research serves as simulation studies on evolution theories, and ultimately should inspire more sophisticated and evolvable evolutionary algorithms.

5. ACKNOWLEDGEMENTS

This research is supported by the Ignite Grant from the Research and Development Corporation of Newfoundland and Labrador and the Discovery Grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) to TH. WB acknowledges support from the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant. The simulation was feasible with the help from the IBM HPC cluster at the Center for Health Informatics & Analytics (CHIA), Faculty of Medicine, Memorial University.

6. REFERENCES

[1] W. Banzhaf, P. Nordin, R. E. Keller, and F. D. Francone. *Genetic Programming: An Introduction*. Morgan Kaufmann, 1998.

[2] A. Bavelas. Communication patterns in task-oriented groups. *Journal of the Acoustical Society of America*, 22:725–730, 1950.

[3] P. Bonacich. Power and centrality: A family of measures. *American Journal of Sociology*, 92(5):1170–1182, 1987.

[4] M. F. Brameier and W. Banzhaf. A comparison of linear genetic programming and neural networks in medical data mining. *IEEE Transactions on Evolutionary Computation*, 5(1):17–26, 2001.

[5] M. F. Brameier and W. Banzhaf. *Linear Genetic Programming*. Springer, 2007.

[6] S. Ciliberti, O. C. Martin, and A. Wagner. Innovation and robustness in complex regulatory gene networks. *Proceedings of the National Academy of Sciences*, 104(34):13591–13596, 2007.

[7] M. C. Cowperthwaite, E. P. Economo, W. R. Harcombe, E. L. Miller, and L. A. Meyers. The ascent of the abundant: How mutational networks constrain evolution. *PLoS Computational Biology*, 4(7):e1000110, 2008.

[8] B. C. Daniels, Y.-J. Chen, J. P. Sethna, R. N. Gutenkunst, and C. R. Myers. Sloppiness, robustness, and evolvability in systems biology. *Current Opinion in Biotechnology*, 19:389–395, 2008.

[9] J. A. G. M. de Visser, J. Hermission, G. P. Wagner, L. A. Meyers, H. Bagheri-Chaichian, and et al. Evolution and detection of genetic robustness. *Evolution*, 57(9):1959–1972, 2003.

[10] J. A. Draghi, T. L. Parsons, G. P. Wagner, and J. B. Plotkin. Mutational robustness can facilitate adaptation. *Nature*, 463:353–355, 2010.

[11] E. Ferrada and A. Wagner. Protein robustness promotes evolutionary innovations on large evolutionary time-scales. *Proceedings of The Royal Society B*, 275:1595–1602, 2008.

[12] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.

[13] E. Galvan-Lopez and R. Poli. An empirical investigation of how and why neutrality affects evolutionary search. In M. Cattolico, editor, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1149–1156, 2006.

[14] A. Guven. Linear genetic programming for time-series modeling of daily flow rate. *Journal of Earth System Science*, 118(2):137–146, 2009.

[15] T. Hu and W. Banzhaf. Neutrality and variability: Two sides of evolvability in linear genetic programming. In F. Rothlauf, editor, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 963–970, 2009.

[16] T. Hu, W. Banzhaf, and J. H. Moore. Robustness and evolvability of recombination in linear genetic programming. In *Proceedings of the European Conference on Genetic Programming*, volume 7831 of *Lecture Notes in Computer Science*, pages 97–108, 2013.

[17] T. Hu, J. H. Moore, and W. Banzhaf. The effects of recombination on phenotypic exploration and robustness in evolution. *Artificial Life*, 20(4):457–470, 2014.

[18] T. Hu, J. L. Payne, W. Banzhaf, and J. H. Moore. Robustness, evolvability, and accessibility in linear genetic programming. In S. Silva, J. A. Foster, M. Nicolau, P. Machado, and M. Giacobini, editors, *Proceedings of the European Conference on Genetic Programming*, volume 6621 of *Lecture Notes in Computer Science*, pages 13–24, 2011.

[19] T. Hu, J. L. Payne, W. Banzhaf, and J. H. Moore. Evolutionary dynamics on multiple scales: A quantitative analysis of the interplay between genotype, phenotype, and fitness in linear genetic programming. *Genetic Programming and Evolvable Machines*, 13:305–337, 2012.

[20] M. A. Huynen, P. F. Stadler, and W. Fontana. Smoothness within ruggedness: The role of neutrality in adaptation. *Proceedings of the National Academy of Sciences*, 93:397–401, 1996.

[21] H. Jeong, S. P. Mason, A. L. Barabasi, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411:41–42, 2001.

[22] M. Kirschner and J. Gerhart. Evolvability. *Proceedings of the National Academy of Sciences*, 95:8420–8427, 1998.

[23] C. R. Landry, B. Lemos, S. A. Rifkin, W. J. Dickinson, and D. L. Hartl. Genetic properties influcing the evolvability of gene expression. *Science*, 317:118–121, 2007.

[24] R. E. Lenski, J. E. Barrick, and C. Ofria. Balancing robustness and evolvability. *PLoS Biology*, 4(12):e428, 2006.

[25] M. Li, J. Wang, H. Wang, and Y. Pan. Identification of essential proteins from weighted protein-protein interaction networks. *Journal of Bioinformatics and Computational Biology*, 11(3):1341002, 2013.

[26] J. Masel and M. V. Trotter. Robustness and evolvability. *Trends in Genetics*, 26:406–414, 2010.

[27] R. C. McBride, C. B. Ogbunugafor, and P. E. Turner. Robustness promotes evolvability of thermotolerance in an RNA virus. *BMC Evolutionary Biology*, 8:231, 2008.

[28] J. F. Miller, editor. *Cartesian genetic programming*. Springer, 2011.

[29] J. F. Miller and P. Thomson. Cartesian genetic programming. In *Prodeedings of the 3rd European Conference on Genetic Programming*, volume 1802 of *Lecture Notes in Computer Science*, pages 121–132. Springer, 2000.

[30] C. L. Nehaniv. Measuring evolvability as the rate of complexity increase. In C. C. Maley and E. Boudreau, editors, *Artificial Life VII Workshop Proceedings*, pages 55–57, 2000.

[31] M. E. J. Newman. Analysis of weighted networks. *Physical Review E*, 70(5):56–131, 2004.

[32] M. E. J. Newman. *Networks: An Introduction*. Oxford University Press, 2010.

[33] M. Pigliucci. Is evolvability evolvable? *Nature Review Genetics*, 9:75–82, 2008.

[34] C. Reidys, P. F. Stadler, and P. Schuster. Generic properties of combinatory maps: neutral networks of RNA secondary structures. *Bulletin of Mathematical Biology*, 59(2):339–397, 1997.

[35] J. Reisinger and R. Miikkulainen. Acquring evolvability through adaptive representation. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1045–1052, 2007.

[36] J. Reisinger, K. O. Stanley, and R. Miikkulainen. Towards an empirical measure of evolvability. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 257–264, 2005.

[37] F. Rothlauf and D. E. Goldberg. Redundant representations in evolutionary computation. *Evolutionary Computation*, 11(4):381–415, 2003.

[38] G. Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, 1966.

[39] P. Schuster, W. Fontana, P. F. Stadler, and I. L. Hofacker. From sequences to shapes and back: A case study in RNA secondary structures. *Proceedings of The Royal Society B*, 255:279–284, 1994.

[40] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Research*, 13:2498–2504, 2003.

[41] D. Song, M. I. Heywood, and A. Zincir-Heywod. A linear genetic programming approch to intrusion detection. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 2724 of *Lecture Notes in Computer Science*, pages 2325–2336. Springer-Verlag, 2001.

[42] T. Soule. Resilient individuals improve evolutionary search. *Artificial Life*, 12:17–34, 2006.

[43] Y. Tang, M. Li, J. Wang, Y. Pan, and F.-X. Wu. CytoNCA: A cytoscape plugin for centrality analysis and evaluation of protein interaction networks. *BioSystems*, 127:67–72, 2015.

[44] E. van Nimwegen, J. P. Crutchfield, and M. A. Huynen. Neutral evolution of mutational robustness. *Proceedings of the National Academy of Sciences*, 96(17):9716–9720, 1999.

[45] V. K. Vassilev and J. F. Miller. The advantage of landscape neutrality in digital circuit evolution. In *Proceedings of the 3rd International Conference on Evolvable Systems: From Biology to Hardware*, volume 1801 of *Lecture Notes in Computer Science*, pages 252–263. Springer, 2002.

[46] A. Wagner. Robustness, evolvability, and neutrality. *Federation of European Biochemical Societies Letters*, 579(8):1772–1778, 2005.

[47] A. Wagner. Neutralism and selectionism: A network-based reconciliation. *Nature Review Genetics*, 9:965–974, 2008.

[48] A. Wagner. Robustness and evolvability: A paradox resolved. *Proceedings of The Royal Society B*, 275(1630):91–100, 2008.

[49] J. Whitacre and A. Bender. Degeneracy: A design principle for achieving robustness and evolvability. *Journal of Theoretical Biology*, 263:143–153, 2010.

[50] C. O. Wilke. Adaptive evolution on neutral networks. *Bulletin of Mathematical Biology*, 63:715–730, 2001.

[51] T. Yu and J. F. Miller. Through the interaction of neutral and adaptive mutations, evolutionary search finds a way. *Artificial Life*, 12:525–551, 2006.