Effects of Genetic Architecture on Evolution of Multiple Traits

Winfried Just
Department of Mathematics
Ohio University
Athens, OH 45701

Fang Zhu
Department of Mathematics
Ohio University
Athens, OH 45701

Abstract

The phenomena of pleiotropy, the influence of individual genes on multiple traits of an organism, and epistasis, the interaction of multiple genes in influencing a single trait, are ubiquitous characteristics of organization of genomes of living organisms. In evolutionary computation, a high degree of epistasis and pleiotropy is often considered an indication of hardness of the problem. The experiments reported here indicate that even for a very simple separable fitness function, pleiotropy and epistasis can enhance the effectiveness of evolution as an optimizing procedure when compared to a straightforward problem representation.

1 INTRODUCTION

Most traits of real organisms seem to be governed by the interaction of multiple genes, and the effect of any one of these genes often depends on the alleles of the other genes that influence the trait. This phenomenon is known as epistasis. On the other hand, many, perhaps most, genes of a real organism influence multiple traits of the organism. This phenomenon is known as pleiotropy. However, genomes of real organisms appear to be organized in modules (Bolker, 2000; Raff, 1996; Wagner and Altenberg, 1996), with little pleiotropic interactions between different modules. The evolutionary origin of modules is an active area of current research; for a survey of recent work see (Wagner et al., to appear).

In a sense, modules are limitations of pleiotropy. Thus modules appear to embody Bonner's Low Pleiotropy Principle (Bonner, 1974) which postulates that the genotype-phenotype map should exhibit low pleiotropy. So why is there so much pleiotropy in real genomes after all? One possible reason is that biological evolution is an opportunistic tinker who reuses whatever raw material (such as genes) is at hand to solve new problems that arise over time. A second possible reason is that there appear to be upper bounds for the size of viable genomes (Ridley, 2001), and organisms simply have to reuse genes for new purposes rather than adding new ones. A third possibility is that pleiotropy and epistasis are unavoidable consequences of the chemical and physical mechanisms by which DNA in real organisms controls phenotypes. Pleiotropy has also been proposed as a factor that can increase the degree of variability maintained in quantitative characters (Hastings and Hom, 1990; Gimmelfarb, 1996). We are interested here in exploring another possibility, namely that pleiotropy and epistasis may make evolution more efficient at optimizing multiple traits simultaneously. The model we are presenting allows us to distinguish between effects of reduced genome size and effects of pleiotropy and epistasis, and there is no premium on maintaining variability.

In evolutionary computation, pleiotropy and epistasis are considered properties of the fitness landscape. Problems that can be represented by fitness landscapes without any epistasis are known as separable problems. Davidor (1991) introduced the notion of epistasis variance of a fitness landscape as one measure of difficulty of a problem for a genetic algorithm. A similar measure was introduced in (Rochet, 1997). Measures of epistasis can be understood as indications how far the fitness function is from separability; see (Naudts and Kallel, 2000) for a recent review. Fitness landscapes that show a high degree of epistasis are usually considered difficult for genetic algorithms; however, there are are some known exceptions to this rule (Manela and Campbell, 1992; Reeves and Wright, 1995; Rochet et al., 1998). Pleiotropy has been studied in the evolutionary computation literature in the context of the evolution of the genotype-phenotype map. Altenberg (1995) found a tendency towards low pleiotropy in simulated evolution of the genotype-phenotype map. In contrast, Kwaśnicka (1997, 1998) reportedly found that pleiotropy and polygene effects can enhance the effectiveness of evolution (Kwaśnicka, 1999). The research quoted above has concentrated on the role of epistasis in non-separable problems, with variants of Kauffman's NK-model (Kauffman, 1989) figuring most prominently.

In this study we demonstrate that pleiotropy and epistasis can enhance the effectiveness of evolution as an optimizing procedure even for separable problems. While we realize that realistic problems of interest to evolutionary computation are non-separable, it is hoped that the very simplicity of our model will make it more amenable to a mathematical treatment than the existing models for which effects of pleiotropy and epistasis have been studied. Of the models published in the literature, our model is closest to the MQT model of Taylor and Higgs (2000); an important difference being that chromosomes in the latter model are bit strings, whereas chromosomes in our model are strings of real numbers. Moreover, the questions studied in (Taylor and Higgs, 2000) are different from the one we are investigating.

2 THE MODEL

We assume that a certain species of animals will repeatedly play games against nature during their lifetime. If the animal wins the game, its fitness will increase by an amount V (value of the resource); if it loses, its fitness will decrease by an amount C (cost). Before each game, the animal is given the probability of winning the game and has to decide whether or not to play the game. This decision is coded by a genetically determined threshold t_j: The animal will play the game if, and only if, the probability of winning is at least t_j. Before each game, the threshold is randomly selected from a list of n thresholds (one can think of these thresholds as representing a variety of situations an animal may face during its life, like hunting for prey of different species or under different conditions). These n thresholds are the multiple traits that evolution is supposed to optimize.

Note that if probabilities of winning the game are distributed uniformly over the interval $[0, 1]$, then the expected fitness gain from playing one round of the game is given by the formula:

$$\sum_{j=1}^{n} \frac{1}{n} \int_{t_j}^{1} pV - (1-p)C \, dp \tag{1}$$

$$= \sum_{j=1}^{n} \frac{1}{n} \Big(\frac{V-C}{2} - \frac{t_j^2}{2}(V+C) + t_j C \Big).$$

The latter quantity is maximized if $t_j = \frac{C}{V+C}$ for all j. The challenge for evolution is to optimize all thresholds simultaneously. Note that the larger the number of thresholds, the smaller the selection pressure on each individual threshold, and the harder it should be to optimize the overall fitness of the animal.

3 OUR SIMULATIONS

In our simulations, we explored two ways of coding these multiple thresholds. In separable coding, each threshold t_j was represented by a real number g_j between 0 and 1. This type of coding was tested for $n = 2, 3, 4, 5, 6, 20$. The other type of coding, non-separable coding, was tested only for $n = 20$. Here the organism's genome consisted of six real numbers $g_1, \dots, g_6$ between 0 and 1. Each of the 20 thresholds was represented by the (arithmetic or geometric) mean of three of these numbers, with each of the 20 combinations of three genes representing a different threshold.

Let us describe the user-definable parameters that were kept constant throughout our simulations. We simulated the evolution of populations of $1,000$ digital animals. Each animal lived for 10 mating seasons and had 20 chances per mating seasons to play a game against nature. The initial fitness of each animal was set to 25 at the beginning of each mating season and then decreased or increased according to the outcomes of the games played by this animal. Note that the simulation of actual outcomes of games, as opposed to simulations where fitnesses are simply calculated from formula (1), adds an extra level of noise to the system. One might expect that this would make it more difficult for evolution to optimize thresholds. To control for this effect, we ran an additional set of simulations were fitness was simply calculated from formula (1) and no actual games were simulated. After each mating season, the oldest animals were replaced by newcomers. Parents of each newcomer were selected randomly, with chances of being selected proportional to fitness attained in this mating season. The genome of the newcomer was inherited from the parents with uniform crossover and possible mutations. Mutations occurred with probability 0.01 at each locus g_j. In

Table 1: Results for first series of simulations

T	**C**	**B. O. F.**	**Sd**
2	sep	0.36%	0.13%
3	sep	0.47%	0.14%
4	sep	0.56%	0.18%
5	sep	0.63%	0.18%
6	sep	0.72%	0.20%
20	sep	2.42%	2.26%
20	ns/am	0.52%	0.22%
20	ns/gm	0.50%	0.18%

Table 2: Results for second series of simulations

T	**In**	**C**	**SEASONS**	**Sd**
5	0.8	sep	2,853	487
6	0.8	sep	3,309	523
20	0.8	ns/am	2,936	457
20	0.8	ns/gm	2,886	367
5	0.2	sep	8,951	1,203
6	0.2	sep	10,269	1,142
20	0.2	ns/am	9,214	1,257
20	0.2	ns/gm	9,788	1,310

Table 3: Results for third series of simulations

T	**In**	**C**	**SEASONS**	**Sd**
5	0.8	sep	4,210	645
6	0.8	sep	5,147	791
20	0.8	ns/am	4,299	623
20	0.8	ns/gm	4,060	522
5	0.2	sep	13,286	2,042
6	0.2	sep	15,445	1,730
20	0.2	ns/am	12,798	1,599
20	0.2	ns/gm	13,902	1,780

the case of a mutation, the new value for g_j was calculated as $g_j = e^{0.05\xi + \ln G_j}$ where G_j represents the value inherited at this locus from the parent and ξ is a random variable with standard normal distribution. We set $V = C = 5$ in each simulation. Thus the theoretically optimal value of each t_j was 0.5, and for players following the optimal strategy, the expected fitness after the 20 games of each mating season would be 50. The program kept track of the actual average fitness of all players at the end of each mating season, of the average expected fitness of all players at the end of a given mating season (calculated by formula (1)), and of the average of the latter over the last $1,000$ seasons.

4 RESULTS

We ran four series of experiments. In the first series, we run 180 simulations each for separable coding of n thresholds with $n = 2, 3, 4, 5, 6, 20$; as well as 180 simulations each with non-separable coding of $n = 20$ thresholds by six genes, where thresholds were computed as arithmetic or geometric averages of three of the genes. Simulations were started from populations of random strategies, where the means of the genes in the initial population were varied from 0.1 to 0.9 in increments of 0.1 for 20 simulations each, with standard deviations of 0.2. Individual simulations were run for $40,000$ mating seasons each. Table 1 shows the results for these simulations. The first column shows the number of thresholds to be optimized, the second column shows the type of coding used, where, e.g., ns/am stands for non-separable coding with arithmetic averages. Column three ("Below optimal fitness") shows the percentages by which the average expected fitness in the last $1,000$ seasons of each simulation were below the theoretically optimal expected fitness of 50. Standard deviations between individual simulations are computed for each run of 180 simulations and shown in the last column.

The results of the first series of experiments do not allow us to distinguish between the effects on stabilizing vs. directional selection. In order to observe the results of directional selection, we started simulations were all genes in the initial population were set to 0.2 or 0.8 and run simulations until the average fitness of the population in a given mating season reached for the first time a value of over 49, which is within 2% of the theoretical optimum. We run 50 simulations each with separable coding for $n = 5$ and $n = 6$ thresholds, as well as with non-separable coding of 20 thresholds by six genes with each threshold represented by the arithmetic and geometric average of three genes. Results are shown in Table 2. The second column indicates the values of the genes in the initial population. Column four shows the average length of the simulation, and column five shows the standard deviation of the latter between the 50 simulations of each run.

The third series of experiments was similar to the second one, but this time we let the simulations run until the average expected fitness of the population, calculated from formula (1), reached for the first time a value of over 49. This was done to control for fluctuations of fitness that might be due to a large number of players being exceptionally lucky in a given season.

The fourth series of experiments was similar to the third one, but this time we did not simulate actual games but instead set the fitness of each individual in

Table 4: Results for fourth series of simulations

T	In	C	SEASONS	Sd
5	0.8	sep	4,405	693
6	0.8	sep	5,015	768
20	0.8	ns/am	4,318	629
20	0.8	ns/gm	4,115	615
5	0.2	sep	12,756	1,818
6	0.2	sep	14,660	1,985
20	0.2	ns/am	12,137	1,393
20	0.2	ns/gm	13,003	1,521

each mating season to the expected fitness calculated from formula (1). This approach further reduces the noise in the system due to random outcomes of games.

5 CONCLUSION

The results with separable coding show clearly that the more thresholds an animal has to optimize, the more difficult it is for evolution to optimize all of these thresholds simultaneously. But the way thresholds are coded greatly matters: In the first series of experiments, the results for non-separable coding of 20 thresholds by six genes were better than the results of separable coding of four thresholds by four genes and significantly better than for separable coding of five thresholds by five genes. A two-sample *t*-statistic gives a P-value of less than 0.002 for comparing the simulations with $n = 5$ to non-separable coding with geometric average and a P-value of close to 0.05 for comparing these simulations to non-separable coding with arithmetic average. This occurs despite the fact that on the phenotypic level there is a fourfold stronger selection pressure on each individual threshold for $n = 5$ than for $n = 20$.

These results strongly suggest that even for a separable fitness function, pleiotropy and/or epistasis can have a beneficial effect on the power of evolution as an optimization procedure. Note that if the effect were simply due to the reduced genome size, then in the simulations where 20 thresholds were coded by six genes we should see results similar to the results obtained by encoding six thresholds by six genes, not results that were better than when encoding four or five thresholds by individual genes.

In the second series of experiments we observed that while directional selection with non-separable coding of 20 thresholds no longer outperformed directional selection with separable coding of five thresholds, it still significantly outperformed directional selection of six thresholds with separable coding. The corresponding P-values of a two-sample *t*-statistic are about 0.06 for comparing non-separable coding with geometric mean to separable coding of six thresholds for the simulations that started from initial thresholds 0.2, and are less than 0.001 for comparing each of the remaining three runs with non-separable coding to the corresponding run with separable coding of six thresholds. It thus appears that the effect reported here mainly improves stabilizing selection, but also has a significant impact on directional selection, which is more relevant for evolutionary computation.

The results of the third and fourth series of simulations show that the observed benefits of non-separable coding are not simply due to the noise in the system. On the contrary, it appears that the inherent noise may in some cases mask the effects reported here. Note that the simulations in the third and fourth series ran for a much larger number of seasons before reaching an average fitness of 49 than in the second series. This appears to be due to chance effects in the second series, where occasionally the population reaches substantially higher actual average fitness in a season than the average fitness calculated from formula (1).

Acknowledgment

This research was partially supported by NSF grant DBI-9904799 to W. J.

References

L. Altenberg (1995). Genome Growth and the Evolution of the Genotype-Phenotype Map. In W. Banzhaf and F. H. Eeckman (eds.), *Evolution and Biocomputation: Computational Models of Evolution.* Berlin, Germany: Springer Verlag.

J. A. Bolker (2000). Modularity in development and why it matters in Evo-Devo. *American Zoologist* **40**(5):770–776.

J. T. Bonner (1974). *On Development: The Biology of Form.* Cambridge, MA: Harward University Press.

Y. Davidor (1991). Epistasis variance: A viewpoint on GA-hardness. In G. J. E. Rawlings (ed.), *Foundations of Genetic Algorithms,* 23–35. San Mateo, CA: Morgan Kaufmann.

A. Gimelfarb (1996). Pleiotropy as a factor maintaining genetic variation in quantitative characters under stabilizing selection. *Genetical Research* **68**:65–73.

A. Hastings and C. L. Hom (1990). Multiple equilibria and maintenance of additive genetic variance in a model of pleiotropy. *Evolution* **44**(5):1153–1163.

S. A. Kauffman (1989). Adaptation on rugged fitness landscapes. In D. L. Stein (ed.), *lectures in the Sciences of Complexity 1,* 527–618. Redwood City, CA: Addison Wesley.

H. Kwaśnicka (1997). Efficiency of Genetic Algorithms with Pleiotropy and Polygene Effect—Simulation Study. In *Proceedings of the Workshop on Intelligent Systems, June 9–13, 1997, Zakopane.* Malbork, Poland: Polish Academy of Sciences.

H. Kwaśnicka (1998). Pleiotropy and polygene effects in evolutionary algorithms. In *VI Proceedings of the Workshop on Intelligent Systems, June 9–13, 1997, Zakopane.* Malbork, Poland: Polish Academy of Sciences.

H. Kwasńicka (1999). K-MODEL—an Evolutionary Algorithm with New Schema of Representation. In *CIMAF'99, II Symposium on Artificial Intelligence, Adaptive Systems, March 22-26, 1999.* Havana, Cuba.

M. Manela and J. A. Campbell (1992). Harmonic analysis, epistasis and genetic algorithms. In R. Männer and B. Manderick (eds.), *Proceedings of the 2nd Conference on Parallel Problim Solving from Nature,* 57–64. Amsterdam, The Netherlands: North Holland.

B. Naudts and L. Kallel (2000). A Comparison of Predictive Measures of Problem Difficulty in Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation* **4**(1):1–15.

R. A. Raff (1996). *The shape of life.* Chicago, IL: University of Chicago Press.

C. Reeves and C. Wright (1995). Epistasis in genetic algorithms: An experimental design perspective. In L. J. Eshelman (ed.), *Proceedings of the 6th International Conference on Genetic Algorithms,* 217–230. San Mateo, CA: Morgan Kaufmann.

M. Ridley (2001). *The Cooperative Gene: How Mendel's Demon Explains the Evolution of Complex Beings.* New York, NY: The Free Press.

S. Rochet (1997). Epistasis in Genetic Algorithms Revisited. *Information Sciences* **102**:133-155.

S. Rochet, G. Venturini, M. Sliman, and E. M. El Kharoubi (1998). A critical and empirical study of epistasis measures for predicting GA performances: A summary. In J.-K. Hao, E. Lutton, E. Ronald, M. Schoenauer, and D. Snyers (eds.) *Artificial Evolution '97,* 275–285. Berlin, Germany: Springer Verlag.

C. F. Taylor and P. G. Higgs (2000). A Population Genetics Model for Multiple Quantitative Traits Exhibiting Pleiotropy and Epistasis. *Journal of theoretical Biology* **203**(4):419–437.

G. P. Wagner and L. Altenberg (1996). Complex adaptations and the evolution of evolvability. *Evolution* **50**(3):967–976.

G. P. Wagner, J. Mezey, and R. Calabretta (to appear). Natural Selection and the Origin of Modules. In W. Callebout and D. Rasskin-Gutman (eds.), *Modularity. Understanding the development and evolution of complex natural systems.* Cambridge, MA: MIT Press.