Limitations of the Fitness-Proportional Negative Slope Coefficient as a Difficulty Measure

Leonardo Vanneschi
Department of Informatics, Systems and Communication
University of Milano-Bicocca, Milan, Italy
vanneschi@disco.unimib.it

Andrea Valsecchi
Department of Informatics, Systems and Communication
University of Milano-Bicocca, Milan, Italy
valsecchi@disco.unimib.it

Riccardo Poli
Department of Computer Science
University of Essex, UK
rpoli@essex.ac.uk

ABSTRACT

Fitness-Proportional Negative Slope Coefficient is a fitness landscapes measure that has recently been introduced as a potential indicator of problem hardness for optimisation. It is inspired to an older measure, the Negative Slope Coefficient, and it has been theoretically modelled. Preliminary experiments have suggested that it may be a good predictor of problem hardness. However, this measure has not undergone any convincing and comprehensive empirical testing. Our objective is to fill this gap. So, we perform empirical tests using a large set of invertible functions of unitation. We find that while this measure may correctly predict the degree of evolvability of a landscape, this does not necessarily correlate with the difficulty of problems. Some landscapes may show, for example, limited evolvability and yet be easy to solve because either solutions are already present in the initial population or the computational resources provided exceed evolvability obstacles. Or it may be impossible to solve them irrespective of their evolvability simply because they are far too vast for the computational resources provided. These situations are hardly captured by the Fitness-Proportional Negative Slope Coefficient.

Categories and Subject Descriptors

I.2.2 [**Artificial Intelligence**]: Automatic Programming

General Terms

Algorithms, Performance

Keywords

Problem Difficulty, Fitness Landscapes, Fitness Clouds, Fitness-Proportional Negative Slope Coefficient

In the last fifteen years there have been several attempts of using algorithm-independent statistical measures (often related to fitness landscapes) to characterise the intrinsic difficulty of searching a given problem space. The work presented in [2] introduced a measure of problem difficulty called Fitness-Proportional Negative Slope Coefficient (*fpnsc*), inspired to the Negative Slope Coefficient *nsc* [3], for which a theoretical formulation can be derived. The *fpnsc*, like the *nsc* is based on the concept of fitness cloud [4], i.e., a plot of the fitness values of a sample of individuals against the fitness values of their neighbours, where a neighbour is obtained by applying one step of a genetic operator to the individual; *nsc* works by partitioning the cloud into a number of bins representing as many different regions of the fitness landscape and it is calculated by joining the bins centroids by segments and summing all their negative slopes. In its original definition, neighbours are generated by applying tournament selection of a prefixed size to neighbourhoods. *Fpnsc* is a modified version of *nsc* which applies fitness-proportional selection to neighbourhoods. As demonstrated in [2], *fpnsc* can be expressed as: $fpnsc = \sum_i \min(0, (E[F''|f_{i+1}] - E[F''|f_i])/(f_{i+1} - f_i))$, where f_i and f_{i+1} are two consecutive fitness values, F'' is a stochastic variable representing the fitness of the offspring of an individual of fitness f after mutation and selection and $E[F''|f]$ is the conditional expected value of F''. For invertible functions of unitation, a precise equation to express $E[F''|f]$ for any fitness value f has been given in [2]. This allows one to estimate the *fpnsc* for all invertible functions of unitation.

In this paper we have tested the *fpnsc* on a large number of invertible functions of unitation and for various search spaces and GA configurations. A subset of the obtained results are reported in Figure 1, where each point in the plots represents the relationship between *fpnsc* (ordinates) and success rate (abscissas) for one invertible function of unitation.

Let us focus on plots (a), (b), (c) first: most test functions were easily solved by the GA irrespective of variations in mutation rate. However, their *fpnsc* values are spread over a relatively wide range (approximately from -4 to 0). Perhaps the easiest interpretation is that most problems of this size are trivial and so the GA is largely unaffected by differences in evolvability. While this is disappointing, it is not entirely unexpected. Jones [1] noted that algorithm-independent landscape difficulty measures can only evaluate intrinsic/relative problem difficulty and are often unaffected by changes in problem scale. However, the success rate of a search algorithm may be *very* affected by problem size. It is likely that at $\ell = 6$ unitation problems are simply too easy for *fpnsc*-based landscape analysis to be have any predictive power. In fact, let u be the unitation value of the global optimum. Then, it is possible to prove that the average (over all functions of unitation) probability of the initial population containing the global optimum is given by $\sum_{u=0}^{\ell}(1-(1-\binom{\ell}{u}/2^{\ell})^s)/\ell$. For the case of $\ell = 6$, these probabilities are 54.2% for population size $s = 5$, 73.0% for population size $s = 10$, 87.5% for population size $s = 20$ and 94.1% for population size $s = 30$. Thus, it is clear that most unitation problems are solved in generation 0, thereby effectively bypassing the need and benefits of evolvability. This analysis could easily be extended to show that whenever a solution is not in the initial population, with $\ell = 6$ solutions are rarely more than Hamming distance 1 away from at least one member of the population. It is then no surprise to see that in many cases solutions can be found in a handful of generations.

Now, let us focus on plots (d), (e) and (f). Here almost all the points fall in two tight vertical bars of equal height, one near the *fpnsc*-axis and one near the right side of the plot. Since in this case

GECCO'09, July 8–12, 2009, Montréal Québec, Canada.
ACM 978-1-60558-325-9/09/07.

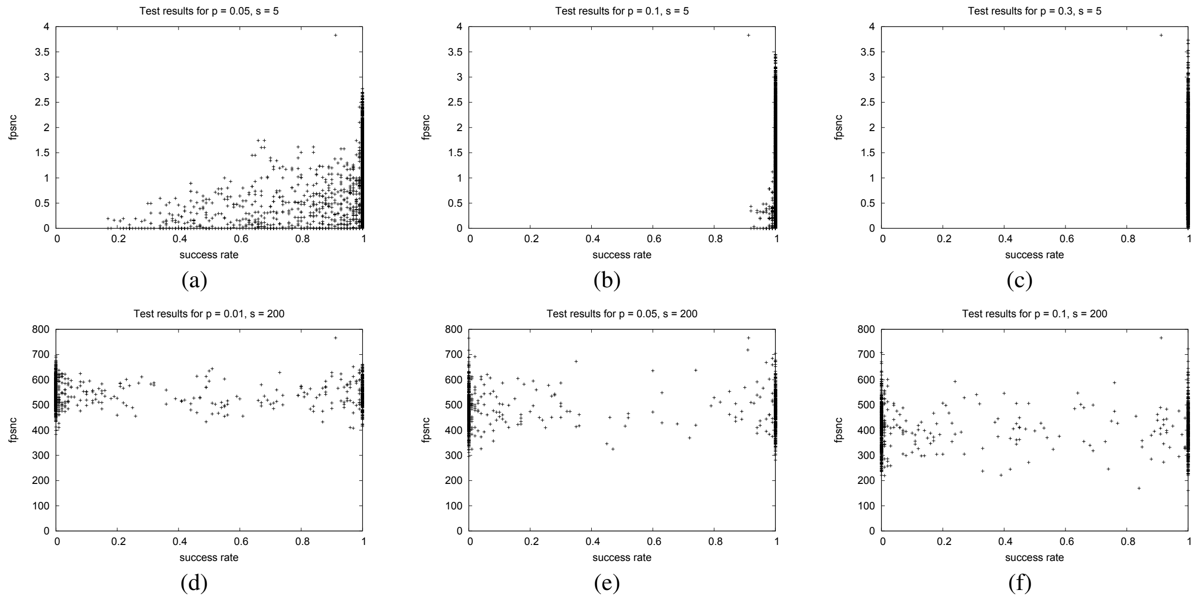

Figure 1: Success rate against *fpnsc*. Plots (a), (b) and (c) report results obtained for *all* the 5040 invertible functions of unitation for bit strings of length $\ell = 6$. The GA has been executed with a population size of 5 individuals. Different values of the mutation rate p_m were used: plot (a) reports results obtained for $p_m = 0.05$, plot (b) for $p_m = 0.1$ and plot (c) for $p_m = 0.3$. Plots (d), (e) and (f) report results obtained for a *sample* of 1000 invertible functions of unitation (chosen with uniform probability over all these invertible functions of unitation) for bit strings of length $\ell = 100$. The GA has been executed with a population size of 200 individuals. Different values of the mutation rate p_m have been used: plot (d) reports results obtained for $p_m = 0.01$, plot (e) for $p_m = 0.05$ and plot (f) for $p_m = 0.1$.

$s = 200$ and $\ell = 100$, the probability of initial populations containing the global optimum is 25.4%. So, again, the combinatorics of functions of unitation implies that a sizable portion of all functions of unitation is solved in very few generations irrespective of how evolvable their landscape is. The remaining functions, however, may be very hard to optimise, because by assigning fitnesses to unitation classes randomly, we effectively create pseudo-random landscapes.

If we discount the generation-0 effects of the combinatorics of functions of unitation, however, we start seeing a rather rosier picture. In particular we should note that while for $\ell = 6$ *fpnsc* values for most functions and parameter settings were in the range $[-4, 0]$, in the case $\ell = 100$ *fpnsc* values are in a much wider range (approximately between -800 and -160) and never near 0. This suggests that there are much greater evolvability obstacles for $\ell = 100$ than for $\ell = 6$ and indeed, success rates confirm this. Also, we can see that *fpnsc* values are markedly modulated by the mutation probability p_m for $\ell = 100$ and indeed looking at plots (d), (e) and (f) one sees that the number of points corresponding to a success rate of 100% increases with p_m.

In conclusion, it is clear that we cannot expect hardness estimators based on landscape features only to correctly predict the performance of algorithms which are either given the option of sample the search space for so long that eventually they will stumble on a solution or are not given enough resources to even get near the global optimum. While the internal behaviour of these algorithms will be affected by the intrinsic evolvability/difficulty of the landscape, their measured performance (e.g., whether they succeeded or not) may be only marginally influenced by it. So, we will either need to come up with ways of scaling evolvability measures, such as *fpnsc*, based on resources and landscape sizes or we will have to accept the fact that such measures will correlate with performance only when computational resources and landscapes sizes are in ranges where evolvability may make the difference between solving a problem or failing to solve it.

Nonetheless, *fpnsc* appeared to predict correctly that the landscapes produced by functions of unitation are rough and potentially difficult to search, correctly predicting also the increase in difficulty as problem size grew. So, we think that in less "deceptive" domains than the rather peculiar functions of unitation and when computational resources are correctly sized for the complexity of the task at hand, *fpnsc* may still play a useful role in assessing problem difficulty.

1. REFERENCES

[1] T. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In L. J. Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 184–192. Morgan Kaufmann, 1995.

[2] R. Poli and L. Vanneschi. Fitness-proportional negative slope coefficient as a hardness measure for genetic algorithms. In D. Thierens *et al.*, editor, *Genetic and Evolutionary Computation Conference, GECCO'07*, pages 1335–1342. ACM Press, 2007.

[3] L. Vanneschi. *Theory and Practice for Efficient Genetic Programming*. Ph.D. thesis, Faculty of Science, University of Lausanne, Switzerland, 2004. Downlodable version at: http://personal.disco.unimib.it/Vanneschi.

[4] S. Verel. *Étude et Exploitation des Réseaux de Neutralité dans les Paysages Adaptatifs pour l'Optimisation Difficile*. PhD thesis, University of Nice – Sophia Antipolis, France, 2005. In French Language.