Analyzing the Behaviour of Population-Based Algorithms Using Rayleigh Distribution

Gabriel Luque and Enrique Alba

Universidad de Málaga, Spain,
Dpto. de Lenguajes y Ciencias de la Computación
E.T.S.I. Informática
Campus Teatinos 29071, Málaga, Spain
{gabriel,eat}@lcc.uma.es

Abstract. This paper presents a new mathematical approach to study the behaviour of population-based methods. The calculation of the takeover time and the dynamical growth curves is a common analytical approach to measure the selection pressure of an EA and any algorithm which manipulates a set of solutions. In this work, we propose a new and more accurate model to calculate these values. This new model also includes other very interesting features, such as the characterization of the complete behaviour of the methods using a single value, the Rayleigh distribution parameter. We also extend the study to consider the effect of the mutation (or in general, any neighborhood exploration operator) and we show several advanced uses of this models such as building self-adaptive techniques or comparing algorithms.

Keywords: Growth Curves, Takeover Time, Rayleigh Distribution.

1 Introduction

Optimizing (or searching or learning) is a commonly practiced sport in designing a new metaheuristic that beats others on a given problem or set of problems. This kind of experimental research finishes by establishing the superiority of a given technique over others. In this scenario, researchers should not be limited to establishing *that* one metaheuristic is better than another in some way, but also to investigate *why*, i.e., they must understand how the algorithms work.

These last studies usually are developed using mathematical tools (run-time analysis, takeover time study or landscapes analysis). In this work, we focus on the takeover time and the growth curves. This approach measures the converge time (time in which all the population is only composed by the best individual) under several assumptions (the best possible individual is in the initial population and only selection operators are employed) This approach was successfully used to analyze GA [1], and other types of population-based algorithms [2,3].

In this work, we propose a new model to calculate the growth curves and takeover time. This new model is more accurate than existing ones, and it has another important feature, it allows to characterize the behaviour of the method

C.A. Coello Coello et al. (Eds.): PPSN 2012, Part I, LNCS 7491, pp. 417–427, 2012.

using a single value (the Rayleigh parameter). Later, we will also extend this model to consider a more realistic case, in which we incorporate the mutation to the method. We study how the new parameter can give us some information of the effect of mutating solutions. Finally, we will show some advanced uses of this mathematical model which can be useful for any researcher. In particular, we show how we can build a new self-adaptive technique and how we can use this model to classify and compare algorithms.

This paper is organized as follows. Section 2 is an introduction containing some preliminary background about some basic concepts, previous models for takeover time, and our proposed approach. In Section 3, we analyze the predicted takeover times provided by the models. Section 4 studies how our model can capture the effect of the mutation. Some advanced uses of the proposed model are given in Section 5. In the last section we summarize the conclusions and give some hints on the future work.

2 Growth Curves and Takeover Times

In this section, we first give a brief definition of growth curves and takeover times (Subsection 2.1). Later, in Subsection 2.2 we describe some existing models for the calculation of these vales and we finish this section analyzing our new approach and its possible advantages.

2.1 Definitions

A common analytical approach to study the selection pressure of an EA is to characterize its takeover time [1], i.e., the number of generations it takes for the best individual in the initial population to fill the entire population under selection only. The growth curves are another important issue to analyze the dynamics of the population-based methods. These growth curves are functions that associate the number of generations of the algorithm with the proportion of the best individual in the whole population. In Fig. 1 we show an example of these two concepts.

Now, we give a mathematical definition of these concepts. Let us start by formally defining what the growth curve is, since it is the basis to define the takeover time.

Definition. Given a population-based algorithm, under selection only, with an initial population containing exactly one individual of the best fitness class; the function $P_{sel} : \mathbb{N} \rightarrow [0, 1]$ that maps the proportion of copies of the best individual in the population to each generation step, is termed as the *growth curve.*

As result of applying selection only in an population-based method (without variation operators), at every generation the number of copies of the best individual potentially grows up. The number of generations it takes for the algorithm to completely fill the population is what is called the takeover time, formally defined as follows:

Definition. Let P_{sel} be the function defining the growth curve induced by a selection method, the value $t_{sel} = min\{t : P_{sel}(t) = 1\}$ is called the *takeover time* associated to the given selection method.

Several models have been proposed to estimate the takeover time for most common selection methods. We can classify the selection techniques in two categories: (a) methods which only depend on the order among the individuals in the population (order-based), and (b) techniques which take the fitness of each individuals into account (fitness-based). In this work, we focus on tournament and proportional selection, which are representative of these two cases, respectively. In next subsection, we discuss some existing mathematical models to estimate the takeover time for these two selection methods, and later, we present our proposal.

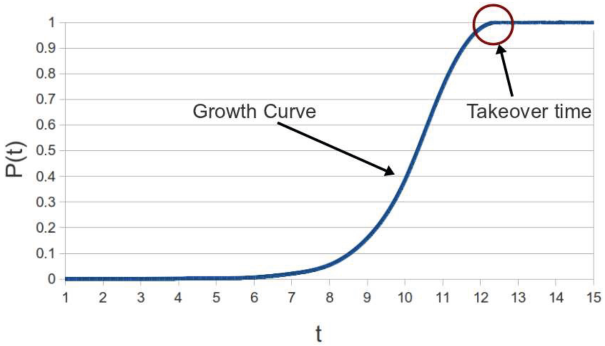

Fig. 1. Growth curve and takeover time for a population-based method using binary tournament selection method

2.2 Existing Models

G&D model. We start analyzing the theoretical models proposed in the original work of Goldberg and Deb [1]. They propose different models for each selection method. In tournament selection, two or more individuals are chosen at random for a fitness-based competition and the best of them is selected with a high probability for breeding. Their model for calculating the takeover time is:

$$t_{tour}^{G\&D} = \frac{1}{\ln s}[\ln \mu + \ln(\ln \mu)] \tag{1}$$

where μ is the population size and s stands for the *tournament size*, i.e., the number of individuals chosen for competition. You can notice that this function only depends of the population size and tournament size, which are both parameters of the algorithm.

In proportionate selection individuals are chosen according to their fitness values, so that the fittest members have a higher chance of being selected. The takeover time in this case is defined to be:

$$t_{prop}^{G\&D} = \frac{1}{c}\mu \log \mu \tag{2}$$

where c is a constant value, which is a function of some fitness-based feature of the initial population.

Logistic model. Let us continue by discussing the work of Sarma and De Jong [2]. In that work, they proposed a simple quantitative model for cellular EAs (which can also be used in panmictic scenarios) based in the logistic family of curve. In summary, the proposed equation for calculating growth curve is (3):

$$P^{LOG}(t) = \frac{1}{1 + b \cdot e^{-at}} \ . \tag{3}$$

where a and b are growth coefficients. This approach uses the same model for any kind of selection method, since the effect of the selection technique is incorporated in the adjustable values a and b. To calculate the takeover time using this model, we can make use of the definition of takeover time of the previous subsection, and iterate this models (starting with $P(0) = 1/\mu$, being μ the population size), searching the lowest t value, which makes $P(t) \approx 1$.

Hypergraph model. Sprave [4] has proposed a unified description for any non-panmictic population structured EA, that could even end in an accurate model for panmictic populations (since they can be considered as fully connected structured populations). He modelled the population structure by means of *hypergraphs*. A hypergraph is an extension of a canonical graph. The basic idea of a hypergraph is the generalization of edges from pairs of vertexes to arbitrary subsets of vertices's.

He developed a method to estimate growth curves and takeover times. This method is based on the calculation of the diameter of the actual population structure and on the probability distribution induced by the selection operator. In fact, Chakraborty *et al.* [5] calculated the success probabilities for the most common selection operators (p_{select}), what represents an interesting complement for putting hypergraphs to work in practice. A complete description of the hypergraph model can be found in [4].

Topology model. Alba and Luque [3] proposed an accurate model for distributed evolutionary algorithms but it could be used for panmictic populations considering this case as special one of a distributed method using an appropriate migration policy. Their proposed model is the following:

$$\begin{aligned} P^{TOP}(t) = \sum_{i=1}^{i=d(T)} & \frac{1/N}{1 + a \cdot e^{-b \cdot (t - per \cdot (i-1))}} \\ & + \frac{N - d(T)/N}{1 + a \cdot e^{-b \cdot (t - per \cdot d(T))}} \end{aligned} \tag{4}$$

where $d(T)$ is the diameter of the topology, *per* is the migration frequency, N is the number of islands, and a and b are growth coefficients . This expression is a combination of the logistic model plus our previous model. In fact, in the panmictic case [1] ($d(T) = 0, per = 0$, and $N = 1$), this equation is the same as the logistic one, and therefore it will not be used in the rest of the paper.

Other models. Some other models were proposed in the literature [6,7,8,9]. These models usually are very accurate ones, but they are linked to specialized algorithms or specific parameter settings. Therefore, in the rest of the paper, we only compare our proposed model against logistic, hypergraph, and Goldberg & Deb models.

2.3 Our Proposed Model

In this paper, we propose a new model for calculating the takeover time (and the growth curves) based on the cumulative function of Rayleigh distribution [10]. The Rayleigh probability density function is:

$$f(x; \sigma) = \frac{x}{\sigma^2} e^{-x^2/2\sigma^2}, \quad x \geq 0, \tag{5}$$

for parameter $\sigma > 0$, and cumulative distribution function:

$$F(x) = 1 - e^{-x^2/2\sigma^2} \tag{6}$$

for $x \in [0, \infty)$.

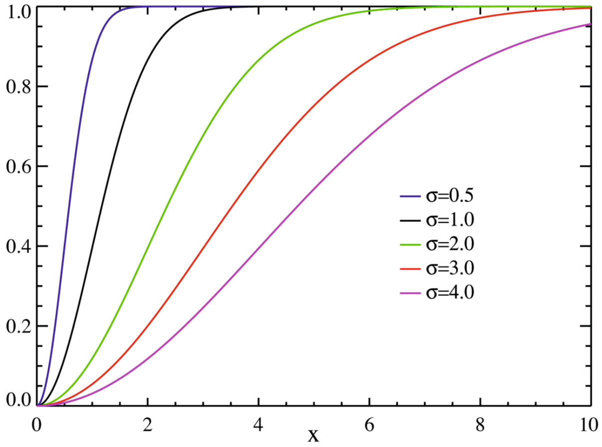

Fig. 2. Cumulative distribution function of Rayleigh distribution

If we observe the cumulative function of this distribution (see Fig. 2), we can notice that it is very similar to the growth curve shape presented in the previous section (Fig. 1) and it is controlled by a single parameter σ. This was one of the reason why we select this model. In additional, this distribution have been deeply studied and some interesting properties have been defined. Maybe, one of the most important properties (from our work point of view) is that it is possible calculate an accurate estimation of the parameter σ with some points (N) of the Rayleigh distribution, using the next equation:

$$\hat{\sigma} \approx \sqrt{\frac{1}{2N}\sum_{i=1}^{N} x_i^2}. \tag{7}$$

Summarizing, the proposed model for the calculation of the growth curves is:

$$P^{RAY}(t) = 1 - e^{-t^2/2\sigma^2} \tag{8}$$

And the expected advantages of this models are:

- Simplicity: A single model that can be used to describe the behaviour of any population-based algorithm under any selection method (and maybe some variations operator as we will showed in Section 4).
- A single control parameter: We can describe the complete behavior of the method using a single real value.
- Ability to predict the control parameter: We do not execute the whole algorithm to calculate the σ parameter, but this value can be estimated with a relative low number of steps of the technique, maintaining a quite accurate result.

3 Accuracy of Models for Takeover Times

In this section we study the precision of the different models to predict the real values of the takeover time for binary tournament and proportional selections. We divide this section into three parts: in the first one, we describe the methodology followed to perform the experiments and the comparisons; later, we compare the accuracy of the existing models and our new approach; finally, we study the ability of our model to calculate the Rayleigh parameter (and therefore, the final takeover time) using only a small number of generations instead of all ones.

3.1 Methodology

We have performed experiments with binary tournament and proportional selection. In the experiments, we use a population of 4096 individuals ($\mu = 4096$), with ($\mu + \mu$) strategy and the initial population is randomly generated with individual fitness between 0 and $\mu - 1$ and then we introduce a single best individual (fitness $= \mu$). In hypergraphs we have used an expected level of accuracy of $\varepsilon = 2.5 \cdot 10^{-4}$. For the actual curves we have performed 100 independent runs.

In order to compare the accuracy of the models we proceeded to calculate the mean square error (9) between the actual values and the theoretically predicted ones (where k is the number of points of the predicted curve).

$$MSE(model) = \frac{1}{k}\sqrt{\sum_{i=1}^{k} (model_i - experimental_i)^2} \ . \tag{9}$$

Using this metric, as we said in the previous section, we compare the accuracy of our proposed model against some existing one. In concrete, the comparison will be performed against the Goldberg, logistic, and hypergraph models.

3.2 Analysis of the Results

Now, we analyze the mean square error for the different models. Fig. 3 contains the error of all the models analyzed in this work for binary tournament and a proportional selection method.

Several conclusions can be obtained from this figure. First, we can notice that the error of the models predicting the takeover time value for the proportional selection is slightly larger than the mean error for tournament selection, but in both cases, the behaviour of the models is quite similar. It is clear that the hypergraph model is not able to capture the dynamics of a panmictic algorithms. This is not a surprising result since this model was proposed for structured populations, and although some configurations makes the behaviour of a non-panmictic algorithm be closer to a panmictic one, there exist still some differences as it is proved by the error that produces this model. The logistic model and the model of Goldberg and Deb obtain quite similar and accurate results, but they are significantly worse than the proposed one.

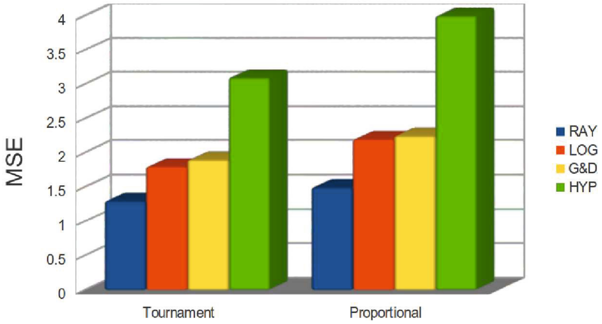

Fig. 3. Mean square error for binary tournament and proportionate selection methods

The first advantage that we expect of our model (the accuracy) is fulfilled, and therefore, in the next subsection, we analyze the second one. The second important expected advantage is the ability to estimate the σ parameter using only the results of a few generations of the method instead of using the results of the complete execution of the technique (as we do in this subsection).

3.3 Accuracy of the σ Estimation

In this section, we use the Equation 7 to estimate the σ parameter. To do it, we calculate the approximation of σ using different amount of generations (measured as a different percentage of the global execution). In Fig. 4, we show the error of our method for the different estimation.

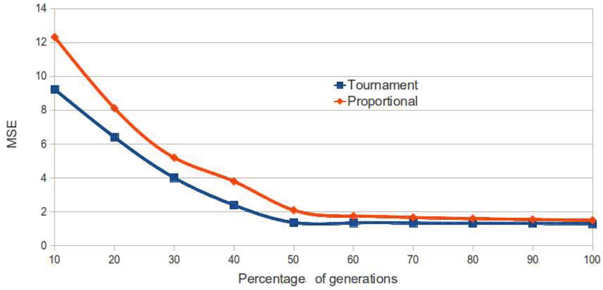

Fig. 4. Mean square error when the σ value is estimated using a different amount of points

As it was expected, the larger the number of generation used for the prediction, the more accurate the results are. We can notice that using about 50-60% of the global execution (depending of the selection method) allows to obtain a very accurate prediction, and adding new data from more generations only gets insignificant improvements. Then, for a good estimation of the σ parameters in this case we only need to run half of the expected total execution. That percentage is quite high (we expected a lower number) but it is due to the takeover time for the tested selection method is quite low (around 25-40 generations), and even a small number of generation implies a high percentage. Anyway, we have observed that this model allows to estimate the parameter controlling its behaviour without the need of completing the execution.

4 Analyzing the Effect of the Mutation

In this section, we tackle a more realistic scenario in which the mutation operator is used. In this preliminary work, we use a simple mutation operator. In concrete, we analyze the bit-flip operator with different intensity (considering this parameter as the number of bits changed in the solution by the operator). Since the calculation of the growth curves only considers increasing successions, in the experiments, we will only take into account the applications of the mutation operator in which the fitness of the resulting solution will be equal or better than the original individual. As test problem for the experiments, we will use the academic OneMax problem with a bitstring of 10000 elements.

In Fig. 5a we show the MSE of our model. We can observe that the error for all the intensities are quite small and similar to the obtained in the previous section (without mutation). Therefore, our model is able to capture successfully the mutation effect without including additional terms in the equation.

Another interesting topic is analyzing how the σ parameter varies with the different mutation intensity and if it is possible to find a relation between both parameters. This relation is shown in Fig. 5b. In that figure, we can observe that there is a clear (inverse) relation between the σ and the intensity of the mutation. This result is quite expected since a higher value of mutation intensity

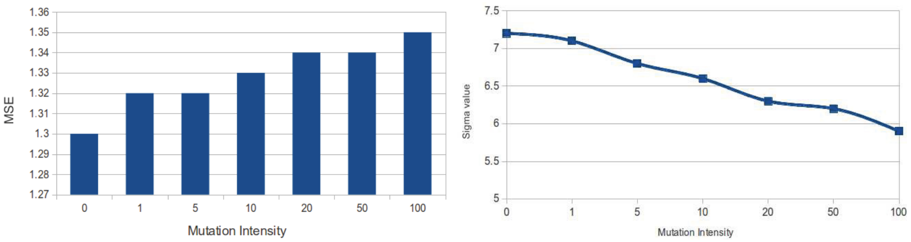

Fig. 5. (a) Mean square error of our model for different mutation intensity and (b) relation between σ parameter and mutation intensity for OneMax problem

represent a faster convergence of the method (more solutions can be the optimal one in a small number of generations), and the σ value controls the slope of the growth curve. This result is quite promising since the σ parameter can be used to summarize the configuration (behaviour) of the method in a single value, but a deeply study is needed.

5 Advanced Uses of the Proposed Model

In this section, we discuss briefly two possible advanced uses of the proposed models

- *Building self-adaptive methods*: As it can analyzed in the previous section, there exists a relation between the parameter of the method and the parameter, σ, which controls the model, and we can use that information to change the setting of the algorithm to force a specific behaviour. However, our model makes several assumptions that cannot be met in real scenarios where the method is to be applied. Therefore, our first challenge consists in adjusting the mathematical models to work for real cases but some initial results about this topic can be found in [11].
- *Comparing and classifying algorithms*: Since the parameter used by our model allows to summarize the behaviour of the complete algorithm in a single real value, it can be used as a metric to compare algorithms (combined with other existing ones as execution time, solution quality, ...) and even to detect classes of equivalence among the different values of the configuration parameters of optimization techniques.

6 Conclusions

In this paper we have performed an analysis of the growth curves and takeover regime of population-based algorithms. We compared the well-known Goldberg model, logistic model, a hypergraph model and a newly proposed model based on the cumulative function of the Rayleigh distribution. In this work we have shown how our models is able to capture the behaviour of the method, obtaining the most accurate prediction of the takeover time.

Also, we have describe how it is possible to calculate a good estimation of the parameter which controls our model without the necessity of execute the complete algorithm. We have extended this work to consider some easy variation operators (mutation one), and we have observed that our model captures its effect accurately, and it is possible to establish a clear relation between the mutation intensity and the σ parameter. Finally, we have shown several practical uses of this mathematical models: to build new algorithms and to compare (or classify) population-based techniques.

As a future work we plan to check the results presented in this paper on additional operator (even recombination ones) and problems. We can also want to perform a comprehensive study about the meaning of σ parameter, and its utilization in practical cases.

Acknowledgments. Authors acknowledge funds from the Spanish Ministry of Sciences and Innovation European FEDER under contract TIN2008-06491-C04-01 (M* project, available in URL `http://mstar.lcc.uma.es`), TIN2011-28194 (RoadME project available in URL `http://roadme.lcc.uma.es`, TIN2011-28194, and CICE Junta de Andalucía under contract P07-TIC-03044 (DIRICOM project, available in URL `http://diricom.lcc.uma.es`).

References

1. Goldberg, D.E., Deb, K.: A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. In: Rawlins, G.J. (ed.) Foundations of Genetic Algorithms, pp. 69–93. Morgan Kaufmann, San Mateo (1991)
2. Sarma, J., De Jong, K.: An Analysis of Local Selection Algorithms in a Spatially Structured Evolutionary Algorithm. In: Bäck, T. (ed.) Proceedings of the 7th International Conference on Genetic Algorithms, pp. 181–186. Morgan Kaufmann, San Francisco (1997)
3. Alba, E., Luque, G.: Theoretical Models of Selection Pressure for dEAs: Topology Influence. In: Corne, D., et al. (eds.) 2005 IEEE Congress on Evolutionary Computation (CEC 2005), Edinburgh, UK, pp. 214–222 (2005)
4. Sprave, J.: A Unified Model of Non-Panmictic Population Structures in Evolutionary Algorithms. In: Angeline, P.J., Michalewicz, Z., Schoenauer, M., Yao, X., Zalzala, A. (eds.) Proceedings of the Congress of Evolutionary Computation, Mayflower Hotel, Washington D.C., USA, vol. 2, pp. 1384–1391. IEEE Press (July 1999)
5. Chakraborty, U.K., Deb, K., Chakraborty, M.: Analysis of Selection Algorithms: A Markov Chain Approach. Evolutionary Computation 4(2), 133–167 (1997)
6. Gorges-Schleuter, M.: An Analysis of Local Selection in Evolution Strategies. In: Banzhaf, W., Daida, J., Eiben, A.E., Garzon, M.H., Honavar, V., Jakiela, M., Smith, R.E. (eds.) Proceedings of the Genetic and Evolutionary Computation Conference, Orlando, Florida, USA, vol. 1, pp. 847–854. Morgan Kaufmann (July 1999)
7. Rudolph, G.: Takeover Times in Spatially Structured Populations: Array and Ring. In: Lai, K.K., Katai, O., Gen, M., Lin, B. (eds.) 2nd Asia-Pacific Conference on Genetic Algorithms and Applications, pp. 144–151. Global-Link Publishing (2000)

8. Cantú-Paz, E.: Migration, Selection Pressure, and Superlinear Speedups. In: Efficient and Accurate Parallel Genetic Algorithms, pp. 97–120. Kluwer (2000)
9. Giacobini, M., Tomassini, M., Tettamanzi, A.G.B.: Modeling Selection Intensity for Linear Cellular Evolutionary Algorithms. In: Liardet, P., Collet, P., Fonlupt, C., Lutton, E., Schoenauer, M. (eds.) EA 2003. LNCS, vol. 2936, pp. 345–356. Springer, Heidelberg (2004)
10. Forbes, C., Evans, M., Hastings, N., Peacock, B.: Rayleigh Distribution. In: Statistical Distributions. John Wiley & Son (2010)
11. Osorio, K., Luque, G., Alba, E.: Distributed Evolutionary Algorithms with Adaptive Migration Period. In: 2011 11th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 259–264 (November 2011)