There is Noisy Lunch: A Study of Noise in Evolutionary Optimization Problems

Juan J. Merelo[1], Federico Liberatore[1], Antonio Fernández Ares[1], Rubén García[2], Zeineb Chelly[3], Carlos Cotta[4], Nuria Rico[5], Antonio M. Mora[6] and Pablo García-Sánchez[1]

[1]*Dept. ATC, University of Granada, Granada, Spain*
[2]*Escuela de Doctorado, University of Granada, Granada, Spain*
[3]*Lab. RODCP, Institut Supérieur de Gestion, Granada, Tunisia*
[4]*Dept. LCC, University of Málaga, Málaga, Spain*
[5]*Dept. EIO, University of Granada, Granada, Spain*
[6]*Dept. TSTC, University of Granada, Granada, Spain*

Keywords: Games, Evolutionary Optimization, Noise, Uncertainty, Noisy Fitness.

Abstract: Noise or uncertainty appear in many optimization processes when there is not a single measure of optimality or fitness but a random variable representing it. These kind of problems have been known for a long time, but there has been no investigation of the statistical distribution those random variables follow, assuming in most cases that it is distributed normally and, thus, it can be modelled via an additive or multiplicative noise on top of a *non-noisy* fitness. In this paper we will look at several uncertain optimization problems that have been addressed by means of Evolutionary Algorithms and prove that there is no single statistical model the evaluations of the fitness functions follow, being different not only from one problem to the next, but in different phases of the optimization in a single problem.

1 INTRODUCTION

Optimization methods usually need a crisp and fixed value to work correctly. This value, usually called *cost* or *fitness*, informs the procedure on how good is the solution and is used to select particular solution over others. This does not imply that these methods need a single floating point number; since they are based on comparisons, it is usually enough that the values can be partially ordered. Multiobjective optimization, for instance, just need to know when comparing two solutions whether one or the other is the best or there can be no comparison between them. In either case, the answer to the question "Is this solution better than the other?" needs to be either a crisp 'Yes' or 'No', or simply "Impossible to know".

In many cases, however, the fitness or cost of a solution cannot be described by a crisp value. In those cases where there is *uncertainty* in the measure, that is, in most real world, physical cases, such as the one described in (Chiaberge et al., 1994), where a control system was optimized , or in the procedure used to evaluate the solution, for instance, when using a stochastic procedure to make that measure, the best way to describe a solution will be a random variable, not a single, or even a vector, value. In our research we have found this happens in many different optimization problems:

- When optimizing the layout of a web-page using Simulated Annealing (SA) (Peñalver and Merelo, 1998). Since SA is a stochastic procedure, the fitness obtained by a solution will be a random variable.
- When training any kind of neural network, such as those in (Chiaberge et al., 1994; Merelo-Guervós et al., 2001); in the second case we dealt with a physical installation, introducing another kind of randomness. Since training a neural network is a stochastic procedure, the error rate obtained after every training run will also follow a statistical distribution.
- When evolving game bots (autonomous agents) (Mora et al., 2010). In this case, the uncertainty arises from the problem itself; in games, several factors such as the initial positions of the players

Merelo, J., Liberatore, F., Ares, A., García, R., Chelly, Z., Cotta, C., Rico, N., Mora, A. and García-Sánchez, P..
There is Noisy Lunch: A Study of Noise in Evolutionary Optimization Problems.
In *Proceedings of the 7th International Joint Conference on Computational Intelligence (IJCCI 2015) - Volume 1: ECTA*, pages 261-268
ISBN: 978-989-758-157-1

or the opponent's behavior add certain stochastic component so that final score will also be *uncertain* or *noisy*; in some cases, too, the bot itself will rely on probabilities to generate its behavior (Fernández-Ares et al., 2014b), in which case two different runs with exactly the same initial conditions and opponent will also yield different scores.

In all these cases it cannot be said that there is *noise* added to a *crisp* fitness. The fitness itself is a statistical variable whose value arises from a stochastic process, evaluation or training, however, we have not seen an exhaustive research of the behavior of fitness as a random variable.

That is why, after some initial study of noise in a particular game in (Merelo et al., 2014), where our findings indicated that, in some cases, noise followed a Gamma, that is a skewed normal distribution and proposing a solution to this using Wilcoxon comparison as a selection operator, we dug into data discovering that, even if the distribution in that particular case was always a gamma, the parameters of the distribution were different, which meant that the random variable behaved in different ways depending on the particular individual, the state of evolution and, of course, the particular problem.

This initial conclusion disagrees with the usual assumptions in optimization in uncertain environments, where it is frequent to assume that the noise is normally distributed and with a fixed sigma (Arnold, 2001). For instance, in the Black Box Optimization Benchmarks (Hansen et al., 2009) the uncertainty was simulated by adding noise centered in 0 and with a Cauchy that is, a centered, sharp bell shaped distribution, with different widths. Either multiplicative or additive noise has been used in different occasions.

That is why in this paper we have collected data from three different problems, which will be presented later on in this paper and tried to find a model for the fitness using statistical tools. Our aim is to eventually find a model that is as general as possible and that is able to account for most sources of uncertainty; failing that, to try and find selection operators that are able to work with random variables in a natural way. However, this is not the focus of this paper and, if it is eventually needed, is left as future work.

The rest of the paper is organized as follows. Next we present the state of the art in evolutionary algorithms in uncertain environments, to be followed by a short presentation of the three problems with uncertainty whose measures will be used in this paper in Section 3. Results will be presented in Section 4, followed by our conclusions.

2 STATE OF THE ART

The most recent and comprehensive review of the state of the art in evolutionary algorithms in *uncertain* environments was done by (Jin and Branke, 2005), although recent papers such as (Qian et al., 2013; Bhattacharya et al., 2014) and (Qian et al., 2014) include brief updates. In that first survey the authors state that uncertainty is categorized into noise, robustness issues, fitness approximation, and time-varying fitness functions, and then, different options for dealing with it are proposed. In principle, the approach presented in this paper was designed to deal with the first kind of uncertainty, noise or uncertainty in fitness evaluation, although it could be argued that there is uncertainty in the true fitness as stated in the third category; however, we do not think that is the case and, in general, that third issue refers to the case in which expensive fitness functions are substituted by surrogate functions which carry a certain amount of error. They suggest several methods, based either on using averaging or using a selection threshold over which one or other individual is selected. But since then, several other solutions have been proposed.

For scientists not concerned on solving the problem of noise, but on a straightforward solution of the optimization problem without modification of existing tools and methodologies, an usual approach is just to disregard the fact that the fitness is noisy and use whatever value is returned by a single evaluation or after re-evaluation each generation. This was the option in our previous research in games although one evaluation in some of those works consists, in fact, in an average of several evaluations, in different maps or considering different opponents, for instance. (Mora et al., 2010; Mora et al., 2012; Liberatore et al., 2015) and evolution of neural networks (Castillo et al., 1999; Merelo-Guervós et al., 2001) and leads, if the population is large enough, to an *implicit averaging* as mentioned in (Jin and Branke, 2005).

In fact, selection used in evolutionary algorithms is also stochastic, so noise in fitness evaluation will have the same effect as randomness in selection or a higher mutation rate, which might make the evolution process easier and not harder in some particular cases (Qian et al., 2013). In fact, Miller and Goldberg proved that an infinite population would not be affected by noise (Miller and Goldberg, 1996) and Jun-Hua and Ming studied the effect of noise in convergence rates (Jun-hua and Ming, 2013), proving that an elitist genetic algorithm finds at least one solution, although with a lowered convergence rate.

But real populations are finite, so the usual approach to dealing with fitness with a degree of randomness is to increase the population size to a value bigger than would be needed in a non-noisy environment. In fact, it has been recently proved that using *sex*, that is, crossover, is able to deal successfully with noise (Friedrich et al., 2015), while an evolutionary algorithm based mainly on mutation, such as the μ+1 EA, or evolutionary programming, would suffer a considerable degradation of performance. However, crossover is part of the standard kit of evolutionary algorithms, so using it and increasing the population size has the advantage that no special provision or change in the implementation has to be made, just different values of the standard parameters.

Another more theoretically sound way is using a statistical central tendency indicator, which is usually the *average*; which happens to be equal to the median in the case of the random variable following the normal distribution. This strategy is called *explicit averaging* by Jin and Branke and is used, for instance, in (Jun-hua and Ming, 2013). Averaging decreases the variance of fitness but the problem is that it is not clear in advance what would be the sample size used for averaging (Aizawa and Wah, 1994). We have used it in some cases but in a different way: not re-evaluating individuals every additional generation and computing the average but computing the fitness using the average of several evaluations, usually five or more (Mora et al., 2012). Most authors use several measures of fitness for each new individual (Costa et al., 2013), although other averaging strategies have also been proposed, like averaging over the neighbourhood of the individual or using *resampling*, that is, more measures of fitness in a number which is decided heuristically (Liu et al., 2014). This assumes that there is, effectively, an average of the fitness values which is true for Gaussian random noise and other distributions such as Gamma or Cauchy, but not necessarily for all distributions.

To the best of our knowledge, other measures like the median which might be more adequate for certain noise models, but which is the same for the normal distribution usually attributed to noise, have not been tested; the median always exists, while the average might not exist for non-centrally distributed variables. Besides, most models keep the number of evaluations fixed and independent of its value, which might result in bad individuals being evaluated many times before being discarded; some authors have proposed *resampling*, (Rada-Vilela et al., 2014; Rakshit et al., 2014), which will effectively increase the number of evaluations and thus slow down the search. In any case, using average is also a small change to the algorithm framework, requiring only using as new fitness function the average of several evaluations.

These two approaches that are focused on the evaluation process might be complemented with changes to the selection process. For instance, using a threshold (Rudolph, 2001; Rakshit et al., 2014) that is related to the noise characteristics to avoid making comparisons of individuals that might, in fact, be very similar or statistically the same; this is usually called *threshold selection* and can be applied either to explicit or implicit averaging fitness functions. The algorithms used for solution, themselves, can be also tested, with some authors proposing, instead of taking more measures, testing different solvers (Cauwet et al., 2014), some of which might be more affected by noise than others. However, recent papers have proved that sampling might be ineffective (Qian et al., 2014) in some types of evolutionary algorithms, adding running time without an additional benefit in terms of performance. This is one lead we will use in the current paper.

Any of these approaches do have the problem of statistical representation of the *true* fitness, even more so if there is not such a thing, but several measures that represent, *as a set* the fitness of an individual. This is what we are going to use in this paper, where we present a method that uses resampling via an individual memory and use either explicit averaging or statistical tests like the non-parametric Wilcoxon test. First we will examine and try to find the shape of the noise that actually appears in some games and other optimization problems; then we will check in this paper what is the influence on the quality of results of these two strategies and which one, if any, is the best when working in noisy environments.

3 PROBLEMS USED IN THIS PAPER

The fitness of three different problems, all of them related to computational intelligence in games, has been used in this paper: generation of character backstories in artificial worlds, described in subsection 3.1, optimization of bots for playing the real time strategy game Planet Wars in 3.2, and optimization of the ghost team in Ms. Pac-Man, which will be described in subsection 3.3. These three problems have been chosen for several reasons, the most important of which is that we have been working on them and thus have data available; another reason is that the origin of the uncertainty is different in the three cases. In the case of MADE, fitness is computed through a simulation; in the case of Planet Wars, the bot themselves have a random component, with its representation including probabilities of different courses of action; and finally in Ms. Pac-Man, it is due to the nature of the game itself. It is not a complete representation of all causes of uncertainty, but the sample is big enough so that we can generalize the results obtained, which will be presented in the next section.

3.1 Creation of Character Backstories

MADE (Massive Drama Engine for non-player characters) (García-Ortega et al., 2014) is a framework for the automatic generation of virtual worlds that allow the emergence of backstories for secondary characters that can later on be included in videogames. In this context, an archetype is a well-known behaviour present in the imaginary collective (for example, a "hero" or a "villain"). Given a fitness to model the existence of different N_a archetypes for a virtual world, MADE uses a genetic algorithm to optimize the parameter values of a Finite State Machine (FSM) that model the agents of that world. For the evaluation, a world is simulated using this parameter set, and the log is analyzed to detect behaviours of the world agents to match with the desired archetypes.

As the evolved parameters are the probabilities to jump from one state to another in the FSM, each fitness evaluation is performed executing the virtual world five times per individual, obtaining the average fitness. Selection is, therefore, performed comparing this average fitness, using a binary tournament in this case. Fitness values range from 0 and N_a and are calculated taking into account the rate of occurrence of the archetypes in the execution log.

3.2 Real Time Strategy: Planet Wars

Planet Wars (Fernández-Ares et al., 2011) is a simple Real-Time strategy (RTS) game. RTS games are not turn-based and their objective is to defeat the enemy using resources available in the map to build units and structures.

Computational intelligence methods have been applied to Planet Wars since it provides a simplification of the elements of the RTS: one kind of units (spaceships) and one kind of resources and structures (planets). Spaceships are automatically generated in the planets owned by the player and they are used to conquer the enemy planets, as this is the objective of the game.

In this paper we are using the results obtained from the Genebot algorithm (García-Sánchez et al., 2014). This algorithm optimizes the parameters of a hand-coded FSM that indicates how many ships send from each planet to attack or reinforce another planet depending of some other values (such as the distance between planets). The generated bot is not deterministic, as some of the jumps of the states are based in probabilities. Fitness is calculated confronting five times the bot obtained from the parameter set of the FSM against a competitive hand-coded bot. The result of each match takes into account the 'slope' of the number of player spaceships during the time of the match. Positive results mean that the bot won, as the slope will be positive, and vice versa. Theoretical values are in the range $[-1,1]$, although these values are impossible to attain in the game. A value of -1 would indicate that the player lost all their ships in the initial time, while 1 would mean the contrary: it generated all the spaceships and won in the initial time. The fitness of an individual is the sum of all five results, and therefore being in the range $[-5,5]$. This fitness has been explained in more detail in (Fernández-Ares et al., 2014a).

3.3 Ghost Team Optimization

Ms. Pac-Man is a variant of the famous Pac-Man game that extends its mechanics with features such as the inclusion of a random event that reverses the direction of the ghosts. This game is used in the Ms. Pac-Man vs Ghosts competition, where participants can submit controllers for both Ms. Pac-Man and the Ghost Team, the first trying to maximize its score, the second trying to minimize Ms. Pac-Man's. The framework used to test the methodology analyzed defines the following restrictions for the Ghost Team:

- A ghost can never stop and if it is in a corridor it must move forward.
- A ghost can choose its direction only at a junction.
- Every time a ghost is at a junction the controller has to provide a direction from the set of feasible directions.
- After 4000 game ticks, a level is considered completed and the game moves on to the next one.

Also, in this method, which was published in (Liberatore et al., 2015), the fitness of each individual is computed as the maximum score obtained by eight different Ms. Pac-Man controllers. Some of these controllers were the best in past editions of the international competition, so they are very tough rivals for the ghost team.

4 EXPERIMENTS AND RESULTS

With the problems presented above, data on fitness was collected by selecting a few random individuals in every generation and measuring its fitness 100 times. Thus, every individual is represented by a random variable with the 100 measures taken with its fitness. According to the usual assumptions, this random variable should follow a normal distribution, with probably different σ and centered on the *true* fitness value. In order to check that hypothesis, we plotted the *skewness*, that is, asymmetry of the distribution, and kurtosis, which is a parameter related to the shape of the distribution. A symmetrical distribution like the normal distribution has a skewness and kurtosis equal to 0; asymmetric distributions, such as the Gamma that we had found in previous papers (Merelo et al., 2014), has non-zero skewness and kurtosis which are related to their α and κ parameters, for instance. Any random variable has skewness and kurtosis at any rate, and we have computed and plotted them in the next figures, where skewness is plotted as x axis against kurtosis in the y axis.

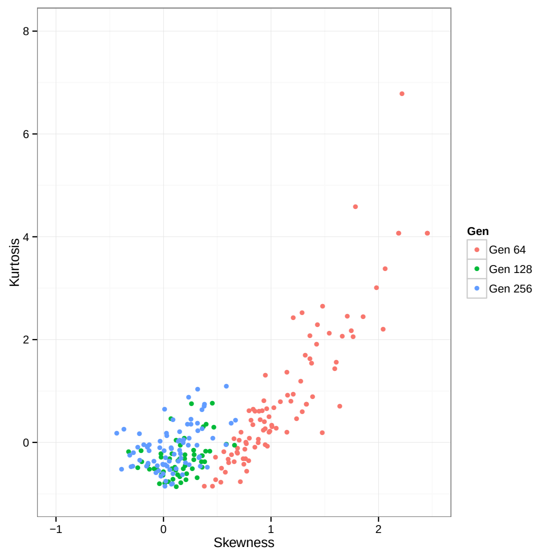

Figure 1: Skewness and kurtosis for fitness in several generations of the MADE problem. Different colors represent diffent generations.

Figure 1 represents them for the MADE problem for which we took measures for a variable amount of individuals every generation, from 100 in generation 64 to around 50 in the latest generation. A curious convergence, but without reaching, the normal distribution is observed as generations proceed; in the first generations, values of skewness and kurtosis are quite high and correspond to arbitrary distribution (Beta or uniform), however, as the simulation proceeds, values approach zero. However, they do not converge exactly to 0, meaning that, even if uncertainty can be approached by a normal distribution, that approximation would only be correct for the latest generations of the simulation. In general, individual fitness will follow an arbitrary distribution with a general shape and asymmetry.

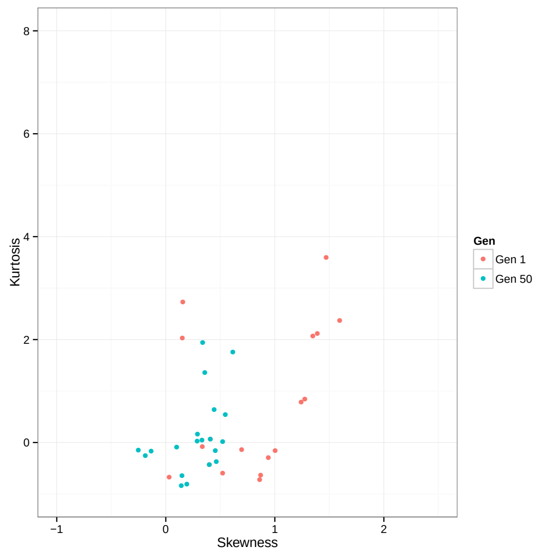

Figure 2: Skewness and kurtosis for fitness in several generations of the Planet Wars problem. Different colors represent different generations.

The shape of the graph for the Planet Wars problem, shown in Figure 2 for two different generations, is different but has some similarities. The dispersion also decreases as evolution proceeds, with shape becoming closer to normal distribution in generation 50. However, initial kurtosis is quite high and values above 2 and below 0 are found even late in the evolution. Noise is, thus, *noisy* and does not conform to a single shape, even less a normal one.

The graph for the final problem, Ms. Pac-Man, is different in several aspects, and is shown in Figure 3. First we have to take into account, as explained in 3.3, that differently from the previous cases, the fitness for a ghost team is the maximum, not an average of several values. This causes a curious behavior of fitness: in the first generation, several individuals have *crisp* values; however, this is decreasingly so, becoming more “random” as generations proceed, that is, the set of values the fitness has got starts to have many different values while in the first generations it had one or a few. That is why the behavior shown in the graph is completely different: distributions get increasingly asymmetric and its shape more different from a normal distribution and more like a Beta distribution. Even if the trend is different from the other two problems, the overall aspect is the same: there is no single distribution that is able to describe the shape of fitness with an uncertainty component.

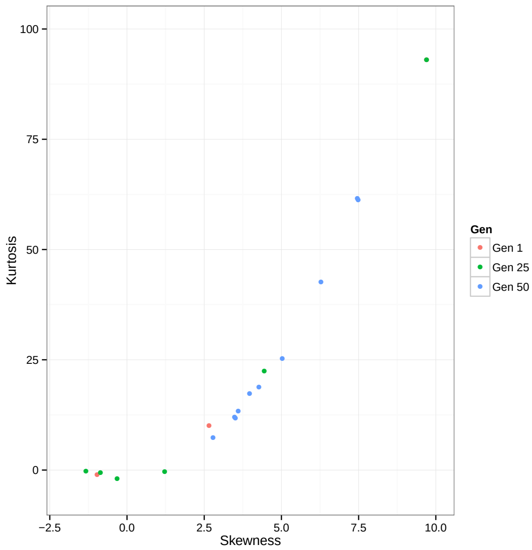

Figure 3: Skewness and kurtosis for fitness in several generations of the Ms. Pac-Man problem. Different colors represent different generations.

5 CONCLUSIONS

In this paper we set out to study the statistical distribution followed by the fitness of single individuals in several problems in the area of games in which we have worked. These problems: MADE, Planet Wars, and Ms. Pac-Man, have different natures and ways to compute the fitness, but all of them have in common that fitness is not a fixed number but a random variable. We have set out to do to prove the hypothesis that not only noise does not follow the normal, or Gaussian, distribution or other centrally-distributed models such as Cauchy, which have been used repeatedly in the literature, but that it does not follow a single distribution even considering a single problem.

The study presented here proves that hypothesis. The best way to describe statistical variables is using two parameters: kurtosis and skewness. These two parameters have been computed and plotted for each one of the problems, proving that not only distributions are asymmetrical and not bell-shaped, but that its shape changes within a single problem and in different stages of the computation. In some case, like MADE, it seems clear that due to the fact that averages are used as a representative for selection, those individuals whose fitness is closer to a central shape are oversampled and thus selected preferably, with almost-central individuals in the latest stages being a consequence of this fact. In other cases, when fitness is computed in a different way or selection takes another form, the effect is exactly the opposite. At any rate, using averages, after the study done in this paper, is discouraged since in many cases and almost always in the early stages of the evolution, fitness, being a random variable, does not pass a centrality test and it might not even have an average. A better way of comparing any fitness with uncertainty would be, as proposed by the authors, using non-parametric tests such as the Wilcoxon test that impose a partial order on the individuals (Merelo et al., 2014); this partial order can be used, in several different ways, for selection.

The fact that there is no single model representing the distribution of fitness also implies that it is an error to use centrally distributed random variables added to a crisp fitness to test operators and algorithms that operate in uncertainty. Either real values should be used, such as the ones proposed above, or a distribution with varying shape and symmetry such as Beta should be used. However, in this case we should take into account that "true" or "crisp" fitness *does not really exist*, so any modelization of uncertain fitness that uses noise added to crisp fitness is, in the more general case, wrong, although it might obviously be true in some cases. If the fitness evaluation is expensive and tests want to be performed for some new operators, the best way to model uncertainty would be to use *different* models applied to every individuals, with different skewness and kurtosis. However, this would be only a first-order approximation and it might favor methods that use averages.

What remains to be done is to effectively apply Wilcoxon-based comparisons to the problems above, but since they are costly to evaluate, we will try to create a benchmark for problems with uncertainty which reflects in the best possible way how fitness is organized in a wide array of problems. In order to do that we will try to examine as many uncertain problems as possible and deduce what would be the most general model.

ACKNOWLEDGEMENTS

This work has been supported in part by projects TIN2014-56494-C4-3-P (Spanish Ministry of Economy and Competitiveness), SPIP2014-01437 (Dirección General de Tráfico), PRY142/14 (Fundación Pública Andaluza Centro de Estudios Andaluces en la IX Convocatoria de Proyectos de Investigación), and project V17-2015 of the Microprojects program 2015 from CEI BioTIC Granada.

REFERENCES

Aizawa, A. N. and Wah, B. W. (1994). Scheduling of genetic algorithms in a noisy environment. *Evolutionary Computation*, 2(2):97–122.

Arnold, D. (2001). Evolution strategies in noisy environments-a survey of existing work. In *Theoretical aspects of evolutionary computing*, pages 239–250. Springer-Verlag.

Bhattacharya, M., Islam, R., and Mahmood, A. (2014). Uncertainty and evolutionary optimization: A novel approach. In *Industrial Electronics and Applications (ICIEA), 2014 IEEE 9th Conference on*, pages 988–993.

Castillo, P. A., González, J., Merelo-Guervós, J.-J., Prieto, A., Rivas, V., and Romero, G. (1999). G-Prop-III: Global optimization of multilayer perceptrons using an evolutionary algorithm. In *GECCO-99: Proceedings Of The Genetic And Evolutionary Computation Conference*, page 942.

Cauwet, M.-L., Liu, J., Teytaud, O., et al. (2014). Algorithm portfolios for noisy optimization: Compare solvers early. In *Learning and Intelligent Optimization Conference*.

Chiaberge, M., Merelo, J. J., Reyneri, L. M., Prieto, A., and Zocca, L. (1994). A comparison of neural networks, linear controllers, genetic algorithms and simulated annealing for real time control. In *Proceedings of the European Symposium on Artificial Neural Networks. Index available from http://www.dice.ucl.ac.be/esann/proceedings/esann1994/content.htm*, pages 205–210. D facto (Brussels). Available from http://polimage.polito.it/ marcello/articoli/esann.94.jj.pdf and a scanned version from http://www.dice.ucl.ac.be/Proceedings/esann/esannpdf/es1994-533-S.pdf.

Costa, A., Vargas, P., and Tinós, R. (2013). Using explicit averaging fitness for studying the behaviour of rats in a maze. In *Advances in Artificial Life, ECAL*, volume 12, pages 940–946.

Fernández-Ares, A., García-Sánchez, P., Mora, A. M., Castillo, P. A., and Guervós, J. J. M. (2014a). Designing competitive bots for a real time strategy game using genetic programming. In Camacho, D., Gómez-Martín, M. A., and González-Calero, P. A., editors, *Proceedings 1st Congreso de la Sociedad Española para las Ciencias del Videojuego, CoSECivi 2014, Barcelona, Spain, June 24, 2014.*, volume 1196 of *CEUR Workshop Proceedings*, pages 159–172. CEUR-WS.org.

Fernández-Ares, A., Mora, A. M., Arenas, M. G., Guervós, J. J. M., García-Sánchez, P., and Valdivieso, P. A. C. (2014b). Co-evolutionary optimization of autonomous agents in a real-time strategy game. In Esparcia-Alcázar, A. I. and Mora, A. M., editors, *Applications of Evolutionary Computation - 17th European Conference, EvoApplications 2014, Granada, Spain, April 23-25, 2014, Revised Selected Papers*, volume 8602 of *Lecture Notes in Computer Science*, pages 374–385. Springer.

Fernández-Ares, A., Mora, A. M., Guervós, J. J. M., García-Sánchez, P., and Fernandes, C. (2011). Optimizing player behavior in a real-time strategy game using evolutionary algorithms. In *IEEE Congress on Evolutionary Computation*, pages 2017–2024. IEEE.

Friedrich, T., Kötzing, T., Krejca, M., and Sutton, A. M. (2015). The Benefit of Sex in Noisy Evolutionary Search. *ArXiv e-prints*.

García-Ortega, R. H., García-Sánchez, P., Mora, A. M., and Merelo, J. (2014). My life as a sim: evolving unique and engaging life stories using virtual worlds. In *ALIFE 14: The Fourteenth Conference on the Synthesis and Simulation of Living Systems*, volume 14, pages 580–587.

García-Sánchez, P., Fernández-Ares, A., Mora, A. M., Valdivieso, P. A. C., González, J., and Guervós, J. J. M. (2014). Tree depth influence in genetic programming for generation of competitive agents for RTS games. In Esparcia-Alcázar, A. I. and Mora, A. M., editors, *Applications of Evolutionary Computation - 17th European Conference, EvoApplications 2014, Granada, Spain, April 23-25, 2014, Revised Selected Papers*, volume 8602 of *Lecture Notes in Computer Science*, pages 411–421. Springer.

Hansen, N., Finck, S., Ros, R., and Auger, A. (2009). Real-parameter black-box optimization benchmarking 2009: Noisy functions definitions.

Jin, Y. and Branke, J. (2005). Evolutionary optimization in uncertain environments - a survey. *IEEE Transactions on Evolutionary Computation*, 9(3):303–317. cited By (since 1996)576.

Jun-hua, L. and Ming, L. (2013). An analysis on convergence and convergence rate estimate of elitist genetic algorithms in noisy environments. *Optik - International Journal for Light and Electron Optics*, 124(24):6780 – 6785.

Liberatore, F., Mora, A., Castillo, P., and Merelo, J. (2015). Comparing heterogeneous and homogeneous flocking strategies for the ghost team in the game of ms. pacman. *Computational Intelligence and AI in Games, IEEE Transactions on*, PP(99):1–1.

Liu, J., St-Pierre, D. L., and Teytaud, O. (2014). A mathematically derived number of resamplings for noisy optimization. In *Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation Companion*, GECCO Comp '14, pages 61–62, New York, NY, USA. ACM.

Lucas, S. M. (2007). Ms pac-man competition. *ACM SIGEVOlution*, 2(4):37–38.

Merelo, J. J., Castillo, P. A., Mora, A., Fernández-Ares, A., Esparcia-Alcázar, A. I., Cotta, C., and Rico, N. (2014). Studying and tackling noisy fitness in evolutionary design of game characters. In Rosa, A., Merelo, J. J., and Filipe, J., editors, *ECTA 2014 - Proceedings of the International Conference on Evolutionary Computation Theory and Applications*, pages 76–85.

Merelo-Guervós, J.-J., Prieto, A., and Morán, F. (2001). *Optimization of classifiers using genetic algorithms*, chapter 4, pages 91–108. MIT press. ISBN: 0262162016; draft available from http://geneura.ugr.es/pub/papers/g-lvq-book.ps.gz.

Miller, B. L. and Goldberg, D. E. (1996). Genetic algorithms, selection schemes, and the varying effects of noise. *Evolutionary Computation*, 4(2):113–131.

Mora, A. M., Fernández-Ares, A., Merelo-Guervós, J. J., García-Sánchez, P., and Fernandes, C. M. (2012). Effect of noisy fitness in Real-Time Strategy games player behaviour optimisation using evolutionary algorithms. *J. Comput. Sci. Technol.*, 27(5):1007–1023.

Mora, A. M., Montoya, R., Merelo, J. J., Snchez, P. G., Castillo, P. A., Laredo, J. L. J., Martnez, A. I., and Espacia, A. (2010). Evolving Bots AI in Unreal. In di Chio et al., C., editor, *Applications of Evolutionary Computing, Part I*, volume 6024 of *Lecture Notes in Computer Science*, pages 170–179, Istanbul, Turkey. Springer-Verlag.

Peñalver, J. G. and Merelo, J.-J. (1998). Optimizing web page layout using an annealed genetic algorithm as client-side script. In *Proceedings PPSN, Parallel Problem Solving from Nature V*, number 1967 in Lecture Notes in Computer Science, pages 1018–1027. Springer-Verlag. http://www.springerlink.com/link.asp?id=2gqqar9cv3et5nlg.

Qian, C., Yu, Y., Jin, Y., and Zhou, Z.-H. (2014). On the effectiveness of sampling for evolutionary optimization in noisy environments. In Bartz-Beielstein, T., Branke, J., Filipic, B., and Smith, J., editors, *Parallel Problem Solving from Nature PPSN XIII*, volume 8672 of *Lecture Notes in Computer Science*, pages 302–311. Springer International Publishing.

Qian, C., Yu, Y., and Zhou, Z.-H. (2013). Analyzing evolutionary optimization in noisy environments. *CoRR*, abs/1311.4987.

Rada-Vilela, J., Johnston, M., and Zhang, M. (2014). Population statistics for particle swarm optimization: Resampling methods in noisy optimization problems. *Swarm and Evolutionary Computation*, 0(0):–. In press.

Rakshit, P., Konar, A., and Nagar, A. (2014). Artificial bee colony induced multi-objective optimization in presence of noise. In *Evolutionary Computation (CEC), 2014 IEEE Congress on*, pages 3176–3183.

Rudolph, G. (2001). A partial order approach to noisy fitness functions. In *Proceedings of the IEEE Conference on Evolutionary Computation, ICEC*, volume 1, pages 318–325.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.