DOI: http://dx.doi.org/10.1109/CEC.2013.6557772

Effects of the Lack of Selective Pressure on the Expected Run-Time Distribution in Genetic Programming

David F. Barrero and María D. R-Moreno
Departamento de Automática
Universidad de Alcalá
Crta. Madrid-Barcelona Km. 33,6
Alcala de Henares, Madrid, Spain
Email: david,mdolores@aut.uah.es

Bonifacio Castaño
Departamento de Matemáticas
Universidad de Alcalá
Crta. Madrid-Barcelona Km. 33,6
Alcala de Henares, Madrid, Spain
Email: bonifacio.castano@uah.es

David Camacho
Departamento de Informática
Universidad Autónoma de Madrid
C/ Francisco Tomás y Valiente 11
Madrid, Spain
Email: david.camacho@uam.es

***Abstract*—Run-time analysis is a powerful tool to analyze algorithms. It is focused on studying the time required by an algorithm to find a solution, the expected run-time, which is one of the most relevant algorithm attributes. Previous research has associated the expected run-time in GP with the lognormal distribution. In this paper we provide additional evidence in that regard and show how the algorithm parametrization may change the resulting run-time distribution. In particular, we explore the influence of the selective pressure on the run-time distribution in tree-based GP, finding that, at least in two problem instances, the lack of selective pressure generates an expected run-time distribution well described by the Weibull probability distribution.**

I. Introduction

The run-time has a direct impact on the algorithm utility. Understanding it may conduct to better practices [1] and clues about the answer to some important theoretical open questions [2], [3]. In order to understand the run-time behavior of algorithms, it is important to characterize its statistical properties, and in particular to find a probability distribution able to model it. Knowing some run-time statistical properties opens powerful parametric statistics to the study of the run-time, enhancing, the often criticized [4], experimental methods in GP, among other applications.

A method widely used to study the run-time is to plot it. To the authors' knowledge, Feo *et al.* [5] introduced this method, and they were followed by several authors, that, with different names and shapes, used that tool. The term that is probably most widely used is Run-Time Distribution, which is the cumulative distribution function of the run-time [6]; in case that time is measured in an architecture-independent way the term Run-Length Distribution is preferred [7]. Some authors prefer constraining the run-time analysis to those runs that found a solution, using different terms to name it, such as expected run-time [3] or time-to-target [1] or generation-to-success [8].

There is a large literature devoted to the experimental analysis of run-time distributions in random search algorithms. Several studies conclude that the run-time follows an exponential or shifted exponential distribution in a wide range of algorithms (backtracking, GA, ACO, WSAT, GWSAT with tabu-lists, TMCH, WMCG or ILS) applied to a variety of problems (TSP, 3SAT, SAT or CSP) [1], [9], [10]. However, other studies points to more complex distributions. In [11], Chiarandini *et al.* found that the run-time fits well a Weibull distribution in ILS, ACO, Random Restart Local Search and two variants of SA applied to the course timetabling problem. However, they report that in some hard problem instances the run-time follows a shifted exponential.

The presence of the exponential distribution is so general that has induced some software packages to only consider this distribution [12], or to conjecture that this distribution is intrinsic to stochastic local search optimization [7]. However, Hoos*et al.* they observed that the run-time distribution may depend on some factors, such as the problem difficulty or parameter settings. To be more specific, optimal parameter settings induced some algorithms run-times to follow an exponential distribution, while suboptimal parameters generate a Weibull [13], [14], [7]. Similarly, they found that easy problems deviate the run-time distribution from the exponential, even when the tail remains as exponential [15].

Curiosly, despite the interest of run-time analysis, there is little empirical work done in the context of classical Koza's style GP. In [8], we proposed a model of success probability based on its decomposition on two components, the expected run-time distribution and the success rate. In order to give an analytic model, we performed a run-time analysis of some common GP problems, finding that the expected run-time, measured in generations, fits well a lognormal distribution.

This paper extends [8] and addresses a new research question: The relationship between the expected run-time distribution and the lack of selective pressure. In contrast to [8] , this study includes two new problem instances to study their run-time lognormality in a standard parameter setting and analyzes the influence of the selective pressure to the expected run-time distribution. The results draw a richer -and more complex- scenario. Under the light of the problem instances considered, the lognormal distribution describes well the run-time distribution in usual conditions, but the lack of selective pressure makes the Weibull distribution a better model. From a practical perspective, understanding the underlying run-time distribution might help to determine the optimal restarting point of an evolutionary algorithm.

The paper is structured as follows. First, we briefly introduce the experimental setup used in the study. Then, in section three we analyze the run-time distribution of six well-known problems in GP. In section four, we analyze the run-time distribution in absence of selective pressure. To complete the picture, section five studies the run-time distribution with a low selective pressure. Finally, some conclusions and future work are outlined.

II. Experimental setup

We only need (to a first approximation) to execute the algorithm on some problems and assess whether their expected run-time follows any known distribution. We consider some classical problem instances widely used by the GP literature belonging to four problem classes: The artificial ant, k-multiplexer, even k-parity and regression without Ephemeral Random Constants (ERC)[1]. Two instances of each binary problem (parity and multiplexer) were considered; 6 and 11 lines were used in the multiplexer, while the parity problem used 4 and 5 lines. The trail used in the artificial ant problem was Santa Fe, as described by Koza. We should mention that the optimal solution of all these problems is known, with the exception of the regression. In this case we have set a fitness limit.

In total six problem instances were used in the experiment, all of them implemented in the well tested ECJ framework. There are founded doubts about the convenience of those problems [16], however, we consider that they are enough given the exploratory nature of this study. Given the limited scope of the problems, any generalization of the results should be taken with caution.

In all the cases we used the implementation and default parameter settings found in ECJ v18, with minor exceptions. The population size and cut off number of generations were modified to tune the algorithm according to the problem difficulty, for instance, 5-parity required $4,000$ individuals in the population and 800 generations to find solutions. The number of timesteps used in the artificial ant was increased to 600. A summary of the settings used in this experiment is shown in Table I.

The algorithm was run a number of times (n) in each problem to obtain a sufficient number of successful runs (k). Some problems were run a huge number of times, $100,000$, because they were reused from previous publications where that number of runs was needed. Other problem instances were run fewer times, enough for the purpose of this study. The number of runs, n, was chosen depending on the computational resources needed by the experiment, which is strongly correlated with the population size and problem difficulty. The number of runs and number of successful runs used in the preliminary experiment is shown in Table II. In order to provide complete reporting on the run-time, the table includes the observed success rate ($\hat{p}$) and its confidence interval computed with Wilson and $\alpha = .05$ [18].

III. Run-time distribution of some classical GP problems

This section briefly studies which statistical distribution fits the run-time. We only consider the expected run-time, and therefore only successful runs are included in the study. We also should mention that the number of generations is a discrete measure, but it will be approximated using continuous distributions in order to compare the results with the literature more easily.

The unit used to measure time is the generation, and since each generation involves a constant number of evaluations, the results can be extrapolated to other measures such as the number of evaluations. Let us denote the expected run-time measured in generations as τ, the run-time in evaluations as T, and the population size as λ, then τ and T are related as $\lambda(\tau - 1) \leq T \leq \lambda\tau$ [3].

The resulting run-time distribution was compared to several distributions (normal, lognormal, Weibull and logistic), finding that the closest one is the lognormal [8], which can be converted into normal just by taking logarithms. With this relationship in mind, we tested the normality of the six empirical distributions by representing the quantile plots of their logarithm, the result is in Fig. 1. As we could expect, the quantile plots show that in general the normal distribution fits well the log(run-time). In the case of the regression problem the fit is excellent, little worse in the 6-multiplexer, regression and the artificial ant. The exception is the two hard Boolean problems (5-Parity and 11-Multiplexer), whose tails clearly deviate from the normal distribution.

With some limitations (hard Boolean problems), the lognormal distribution seems a reasonable choice to model the expected run-time at least in these 6 problems. Additionally, it has an interesting property: Lognormal data can be converted easily into normal, and then all the well-known parametric statistics can be used. At this point, a natural question that raises is to which extent the lognormality is a general property of run-time in GP.

IV. Distribution of the expected run-time with random selection

Even though the problem instances so far studied have shown lognormal expected run-times, the generality of this observation is unclear. In order to assess the limits of the run-time lognormality we have carried out an experiment with an extreme parameter setting. Theoretical studies in simple EAs have shown that the balance between mutation and selection has a direct impact on the run-time behavior of the

[1] All the code, configuration files, scripts and datasets needed to reproduce the experiments reported in this paper are published on http://atc1.aut.uah.es/~david/cec2013/.

TABLE I

Tableau for the problem: Artificial ant with the Santa Fe trail, 6-multiplexer, 11-multiplexer, even 4-parity, even 5-parity and symbolic regression without ERC.

Parameter	**Artificial ant**	**6/11-multiplexer**	**4/5-parity**	**Regression**
Population	500	500	4,000	500
Generations	50	50	800	50
Tournament size	7	7	7	7
Success	fit.=0	fit.=0	fit.=0	fit.$\leq$0.001
Observations	Timesteps=600 Santa Fe trail		Even parity	No ERC $y = x^4 + x^3 + x^2 + x$ $x \in [-1, 1]$

TABLE II

Estimation of the difficulty to find a solution. It reports the number of runs (n), number of successful runs (k), estimation of success rate $\hat{p}$ and Wilson [17] confidence intervals of the success rate with $\alpha = 0.95$, lower(L_p) and upper (U_p) values. Intervals were computed using R's `binom` package.

	Artificial ant	**6-Multiplexer**	**11-Multiplexer**	**4-Parity**	**5-Parity**	**Regression**
n	100,000	100,000	1,000	400	5,000	100,000
k	13.168	95.629	333	299	305	29,462
$\hat{p}$	0.132	0.956	0.333	0.747	0.061	0.295
L_p	0.1296	0.9550	0.3045	0.7027	0.0547	0.2918
U_p	0.1338	0.9575	0.3628	0.7876	0.0680	0.2975

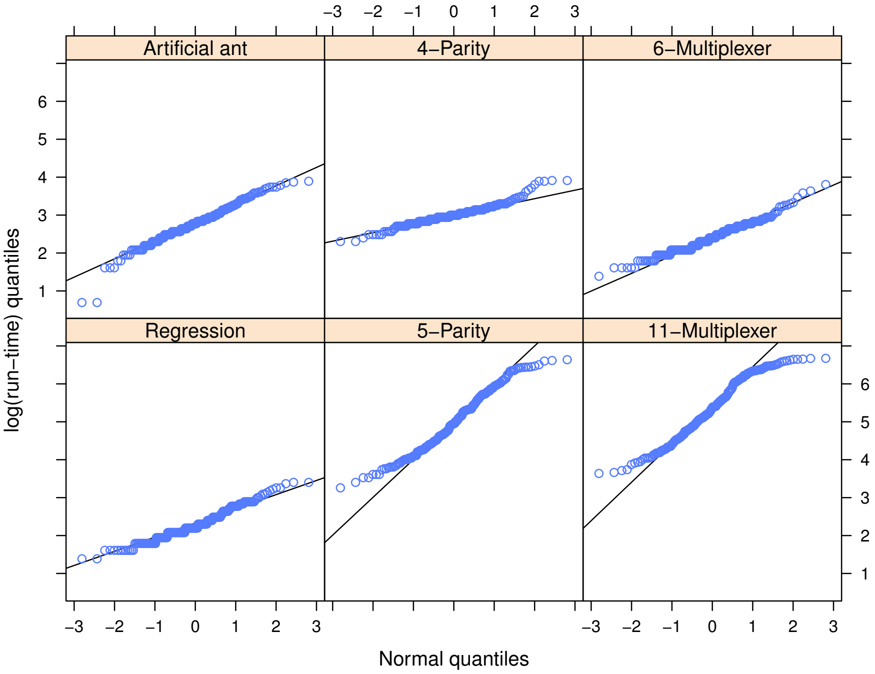

Fig. 1. Quantile plots of the logarithm of the expected run-time, measured in generations, against samples drawn from normal distributions.

algorithm [3]. So, we eliminated the selective pressure to force an extreme behavior and in this way test the lognormality of the run-time.

We carried out a new experiment using the same problem instances and parameter settings, but reducing the tournament size to one to eliminate the selective pressure. As a consequence, the search is random and therefore the difficulty of finding a solution went up dramatically. We only got successful runs in two out of the six problem instances, even when the limit of generations was increased to $1,000$. Table III shows the estimate of the success rate with Wilson intervals, number of trials and successful runs.

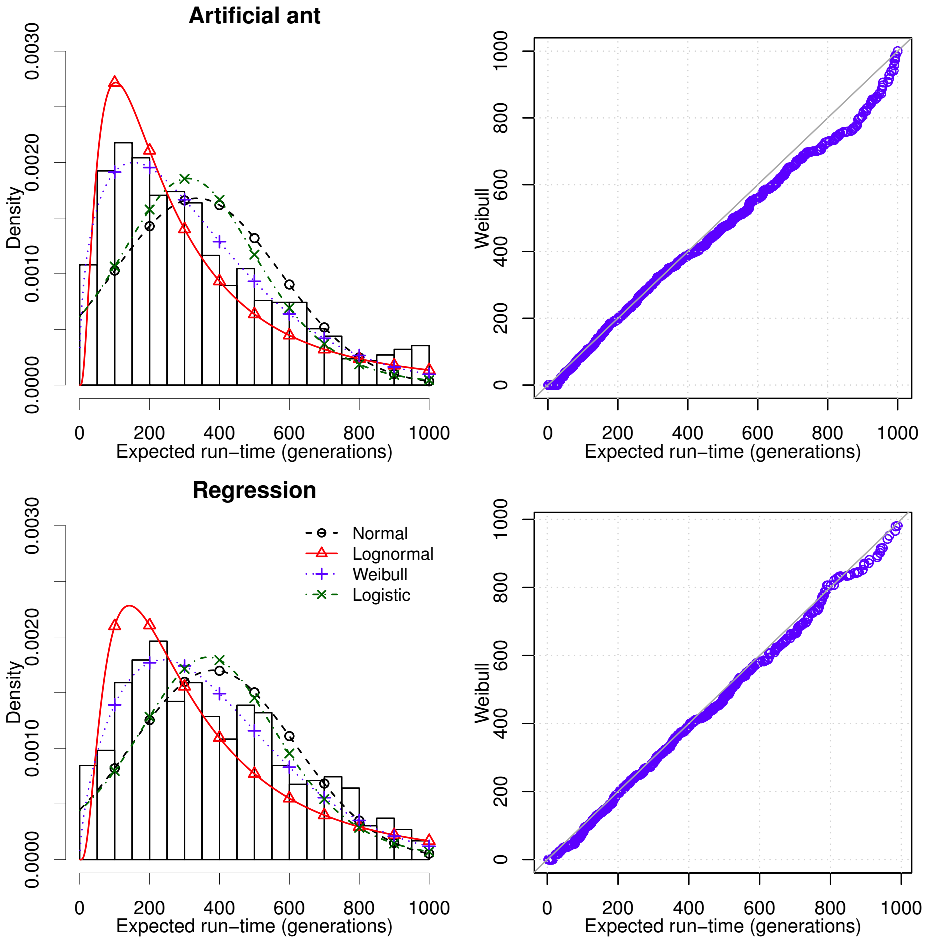

Fig. 2. Histogram of the measured run-time of two problem instances solved by GP without selective pressure (left) and quantile plots comparing data and samples drawn from a Weibull distribution (right).

TABLE III
SOME EXPERIMENTAL PARAMETERS AND ESTIMATION OF THE SUCCESS RATE IN ABSENCE OF SELECTIVE PRESSURE. IT REPORTS THE NUMBER OF RUNS (n), NUMBER OF SUCCESSFUL RUNS (k), ESTIMATION OF THE SUCCESS RATE $\hat{p}$ AND WILSON INTERVALS WITH $\alpha = 0.05$, LOWER(L_p) AND UPPER (U_p) VALUES.

	Artificial ant	**Regression**
n	5,000	5,000
k	1,185	591
$\hat{p}$	0.237	0.118
L_p	0.225	0.109
U_p	0.249	0.127

The histograms of the run-times are depicted in Fig 2 (left). Since there is no selective pressure, the population is not pushed to good regions of the search space, making the search random (however, with memory due to code bloat). As a result, the efficiency of the algorithm for finding a solution has been reduced notably, and indeed only two out of the six problems instances found enough solutions to be significant. Hence, only two problems are reported in this section, the artificial ant and the regression. In addition, the expected run-time has been dramatically increased, problems that required at most 50 generations with tournament size 7, without selective pressure require a large number of generations to find the solution, if it is found.

Perhaps the most interesting fact shown in the Fig 2 (left) is the shape of the histogram. It overlaps the probability density distributions fitted using maximum-likelihood to the histograms. Surprisingly, the lognormal distribution fails modelling the expected run-time; it contains a pronounced peak and a rapid decay, but experimental expected run-times show a smoother shape. In fact, a new distribution able to fit data pretty well appear into scene - the Weibul distribution. This observation is strongly supported by the quantile plots depicted in Fig. 2 (right); the straight line in the plots suggest that the expected run-time follows a Weibull distribution.

To avoid subjectivity in analysis, we performed rigorous Kolgomorov-Smirnov (K-S) and Anderson-Darling (A-D) goodness of fit tests. Table IV summarizes the results reporting the Weibull parameters (k, λ) estimated using maximum-likelihood and the tests statistics, D and A. The tests were

TABLE IV
KOLMOGOROV-SMIRNOV (K-S) AND ANDERSON-DARLING (A-D) GOODNESS OF FIT TESTS. THEY TEST WHETHER THE EXPECTED RUN-TIME WITH TOURNAMENT SIZE ONE FOLLOWS A WEIBULL DISTRIBUTION. THE TABLE REPORTS THE ESTIMATED PARAMETERS OF THE DISTRIBUTION (K, λ), THE K-S STATISTIC (D) AND THE A-D STATISTIC (A). THE ASTERISK (*) DENOTES THAT A TEST PASSED WITH $\alpha = 0.05$.

	$\hat{k}$	$\hat{\lambda}$	**D**	**A**
Artificial ant	1.43	355	**0.039***	**0.267***
Regression	1.42	393	**0.034***	**0.281***

conducted on 150 random samples. Tests that could not reject the null hypothesis are marked in bold letters and asterisk. K-S and A-D tests did not found evidence to reject that the Weibull distribution fits the expected run-times with $\alpha = 0.05$.

The Weibull distribution has some interesting properties that it is worth to analyze briefly. The interpretation made by Hoos and Stützle in [15] in the context of Stochastic Local Search might be applied here with some care. They suggested that the Weibull distribution in hard problems models the initial search phase, which is not present in easy problems. This conjecture cannot be directly applied to GP because of the obvious differences in the algorithms, but can be adapted.

The Weibull distribution asymptotically approximates the exponential distribution. In fact, Weibull is a generalization of the exponential distribution. Given enough time, if the algorithm without selective pressure does not find a solution, its expected run-time would become an exponential random variable. And this is interesting, because the exponential distribution is the only continuous memoryless distribution. This is an observation with practical consequences, given that the algorithm has no memory, it cannot benefit from restarts [11]. This observation makes sense in a scenario without selective pressure, where population is selected at random.

The Weibull distribution suggests an exponential -memoryless- behavior in large expected run-times, but this interpretation does not hold in low (non-exponential) run-times. In other words, when the run begins, it has memory but later it tends to be memoryless. The presence of memory cannot be explained by the selection, but there is one GP characteristic that introduces memory in the algorithm even in absence of selective pressure: The size of the trees changes along the run even when there is no selective pressure. This is an issue that probably deserves some future research.

V. RUN-TIME DISTRIBUTION WITH LOW SELECTIVE PRESSURE

In this section we examinate the run-time with the tournament size set to two. The lack of selective pressure determines the expected run-time distribution, at least, in two problems. We found that under usual circumstances the expected run-time follows a lognormal distribution, but it can change removing the selective pressure. It is worth to question what happens in an intermediate scenario with selective pressure, but much lower than in the initial experiment. Hence, the tournament size was set to two.

We used the same parameter settings shown in Table I, with two differences: The tournament size values two and the number of generations was modulated according the difficulty of finding a solution. The number of runs were 500 in almost all the problems, with the exception of the 11-multiplexer due to the computational cost. Table V summarizes the number of runs, successful runs and success rate estimates.

We verified whether the expected run-time with low selective pressure fit better a lognormal or Weibull distribution. To this end, Fig. 3 plots the histogram of the expected run-times overlapping the lognormal and Weibull distributions. As in previous experiments, we fit the parameters with maximum-likelihood. Interestingly, the figure clearly shows that in this case the lognormal distribution fits data better than the Weibull distribution in the six problem instances.

Fig. 3 suggests that, in general, the lognormal distribution fits well the expected run-time with tournament size two. There are problems where the fit is almost perfect (6-multiplexer), while in others the fit is worse (11-multiplexer and 4-parity). In any case, the lognormal distribution fits our data better than the Weibull. Therefore, in presence of selective pressure, even if it is small, the lognormal distribution is a better alternative to model the expected run-time than the Weibull in the six problem instances covered in this study.

Finally, we compare the resulting run-time distributions obtained with different tournament sizes. Fig. 4 contains the kernel plots of the algorithms of the expected run-times. Using the algorithms instead of the raw run-times eases the comparison and let the visual identification of lognormal distributions, which must appear as normal.

The plot in Fig. 4 reveals some interesting facts. High selective pressure have a positive effect from the run-time perspective, using a tournament size of seven reduces the run-time mean in all the studied cases. However, this observation should be taken with care; we should not conclude that high selective pressure enhance the search process as a general rule.

The shape of the distributions in Fig. 4 suggests that the run-time distribution without selective pressure has a different nature. With some exceptions, almost the only difference between the distributions with tournament size two and seven is the mean, both generate similar variances and shapes. However, when the tournament size equals one, the resulting distributions clearly have negative skews, which is incompatible with a normal nature. This gives credence to the fact that, at least in two problems, the lack of selective pressure affects the run-time distribution.

VI. CONCLUSIONS AND FUTURE WORK

This paper is an attempt to increase our understanding of GP by using run-time analysis. In particular, we have studied how the lack of selective pressure affects the time that the algorithm consumes to find a solution -i.e., the expected run-time. We adopted an experimental perspective, trying to find a statistical distribution able to model the expected run-time.

In common parameter settings, the lognormal distribution fits quite well the expected run-time in the six GP problems

TABLE V

Some experimental parameters and estimate of the success rate to find a solution with low selective pressure (tournament size two). It reports the number of runs (n), number of successful runs (k), estimation of the success rate $\hat{p}$ and Wilson intervals with $\alpha = 0.95$, lower(L_p) and upper (U_p) values.

	Artificial ant	6-Multiplexer	11-Multiplexer	4-Parity	5-Parity	Regression
n	500	500	200	500	500	500
k	51	450	63	258	262	314
$\hat{p}$	0.102	0.9	0.315	0.516	0.524	0.628
L_p	0.078	0.871	0.255	0.472	0.480	0.585
U_p	0.132	0.923	0.382	0.560	0.567	0.669

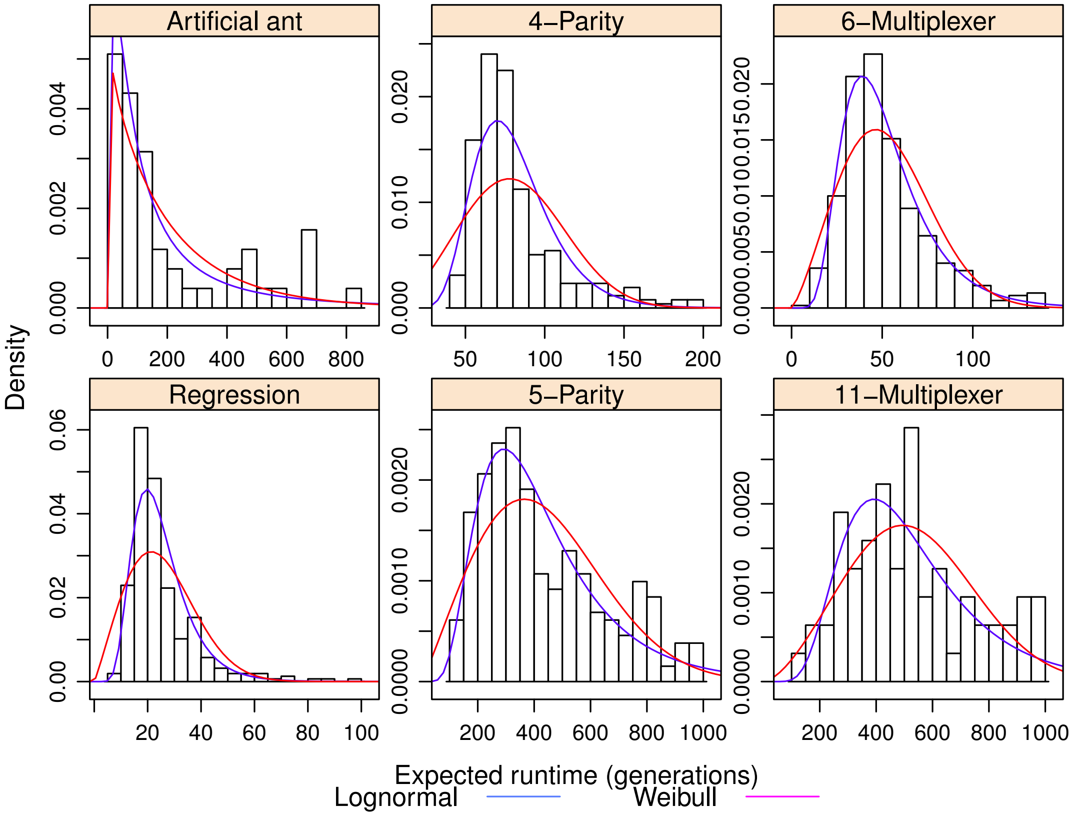

Fig. 3. Histogram of the expected runtime generated with a tournament size of 2. Lognormal (blue) and Weibull (red) distributions fitted with maximum-likelihood are overlapped.

we have studied. More interestingly, if the selective pressure is eliminated by means of setting the tournament size to one, the run-time distribution fits well a Weibull distribution. We should stress that this result applies to the two problems where we found solutions, and therefore any generalization should be done with care.

In any case, experiments shown in this paper, and experiments reported in related literature have shown a complex picture, where some statistical distributions are involved interacting with the parameters settings. Several questions remain open; probably the most important one is to understand why the run-time distribution is affected by the lack of selective

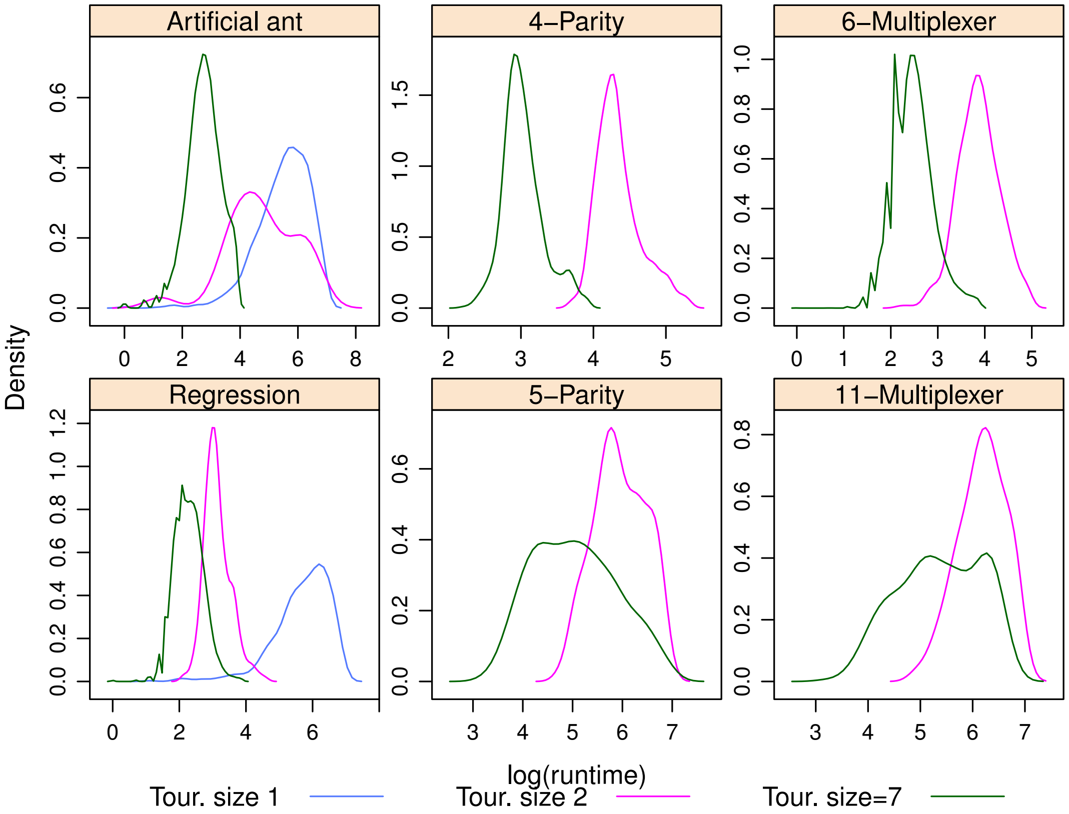

Fig. 4. Kernel density estimates of the logarithm of the expected run-times with different tournament sizes: one (no selective pressure) in blue, two in red and seven in green. Run-time is measured in generations.

pressure. Curiosly, the distributions that use to appear in run-time analysis, exponential, lognormal and Weibull distributions are the three key distributions in Reliability Theory to model the survival time of components, in contrast, for instance, to human creations modelling [19]. It is known that the failure rate determines the distribution of the life time. Ironically, identifying failure with success in GP could be a first step to explain the expected run-time distributions. Other directions to explore would be to study the connection between the selective pressure and the hardness of the problem.

Acknowledgements

This work has been partly supported by Spanish Ministry of Science and Education under project ABANT (TIN2010-19872).

References

[1] C. C. Ribeiro, I. Rosseti, and R. Vallejos, "On the Use of Run Time Distributions to Evaluate and Compare Stochastic Local Search Algorithms," in *Proceedings of the Second International Workshop on Engineering Stochastic Local Search Algorithms*, ser. SLS '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 16–30.

[2] J. E. Rowe and D. Sudholt, "The choice of the offspring population size in the (1,λ) EA," in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference - GECCO '12*. New York, New York, USA: ACM Press, Jul. 2012, p. 1349. [Online]. Available: http://dl.acm.org/citation.cfm?id=2330163.2330350

[3] P. K. Lehre and X. Yao, "On the Impact of Mutation-Selection Balance on the Runtime of Evolutionary Algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 2, pp. 225–241, Apr. 2012. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=\&arnumber=5910379\&contentType=Journals+\&+Magazines\&sortType=asc_p_Sequence\&filter=AND(p_IS_Number:6176231)

[4] A. E. Eiben and M. Jelasity, "A critical note on experimental research methodology in ec," in *Proceedings of the 2002 Congress on Evolutionary Computation (CEC2002)*. IEEE, 2002, pp. 582–587.

[5] T. Feo, M. Resende, and S. Smith, "A greedy randomized adaptive search procedure for maximum independent set," *Operations Research*, pp. 860–878, 1994.

[6] H. H. Hoos, "Stochastic local search - methods, models, applications," Ph.D. dissertation, Technische Universitat Darmst́adt, Germany, 1998.

[7] H. Hoos and T. Stützle, "Towards a characterisation of the behaviour of stochastic local search algorithms for SAT," *Artificial Intelligence*, vol. 112, no. 1-2, pp. 213–232, 1999.

[8] D. F. Barrero, B. Castaño, M. D. R-Moreno, and D. Camacho, "Statistical Distribution of Generation-to-Success in GP: Application to Model Accumulated Success Probability," in *Proceedings of the 14th European Conference on Genetic Programming EuroGP 2011*, ser. LNCS, vol. 6621. Turin, Italy: Springer Verlag, 2011, pp. 155–166.

[9] D. Frost, I. Rish, and L. Vila, "Summarizing CSP hardness with continuous probability distributions," in *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence*. AAAI Press, 1997, pp. 327–333.

[10] T. Stützle and H. Hoos, "Analyzing the run-time behaviour of iterated local search for the TSP," in *III Metaheuristics International Conference*. Kluwer Academic Publishers, 1999.

[11] M. Chiarandini and T. Stützle, "Experimental evaluation of course timetabling algorithms," Intellectics Group, Computer Science Department, Darmstadt University of Technology, Darmstadt, Germany, Tech. Rep. AIDA-02-05, April 2002.

[12] R. M. Aiex, M. G. C. Resende, and C. C. Ribeiro, "TTT plots: a perl program to create time-to-target plots," *Optimization Letters*, vol. 1, no. 4, pp. 355–366, Oct. 2006. [Online]. Available: http://link.springer.com/article/10.1007/s11590-006-0031-4

[13] H. Hoos and T. Stützle, "Evaluating Las Vegas algorithms – pitfalls and remedies," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI-98)*. Morgan Kaufmann Publishers, 1998, pp. 238–245.

[14] ——, "Characterizing the run-time behavior of stochastic local search," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 1998.

[15] H. Hoos and T. Stützle, "Local search algorithms for SAT: An empirical evaluation," *Journal of Automated Reasoning*, vol. 24, no. 4, pp. 421–481, 2000.

[16] J. McDermott, K. De Jong, U.-M. O'Reilly, D. R. White, S. Luke, L. Manzoni, M. Castelli, L. Vanneschi, W. Jaskowski, K. Krawiec, and R. Harper, "Genetic programming needs better benchmarks," in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference - GECCO '12*. New York, New York, USA: ACM Press, Jul. 2012, pp. 791–799.

[17] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," *Journal of the American Statistical Association*, no. 22, pp. 309–316, 1927.

[18] D. F. Barrero, D. Camacho, and M. D. R-Moreno, "Confidence Intervals of Success Rates in Evolutionary Computation," in *GECCO '10: Proceedings of the 12th annual conference on Genetic and Evolutionary Computation*. Portland, Oregon, USA: ACM, 2010, pp. 975–976.

[19] I. Herraiz, D. Rodriguez, and R. Harrison, "On the statistical distribution of object-oriented system properties," in *Third International Workshop on Emerging Trends in Software Metrics (WETSoM'2012),*, June 2012, pp. 56–62.