Genetic Programming and Evolvable Machines manuscript No.
(will be inserted by the editor)

A Study on Koza's Performance Measures

David F. Barrero · Bonifacio Castaño ·
María D. R-Moreno · David Camacho

Received: date / Accepted: date

Abstract John R. Koza defined several metrics to measure the performance of an Evolutionary Algorithm that have been widely used by the Genetic Programming community. Despite the importance of these metrics, and the doubts that they have generated in many authors, their reliability has attracted little research attention, and is still not well understood. The lack of knowledge about these metrics has likely contributed to the decline in their usage in the last years. This paper is an attempt to increase the knowledge about these measures, exploring in which circumstances they are more reliable, providing some clues to improve how they are used, and eventually making their use more justifiable. Specifically, we investigate the amount of uncertainty associated with the measures, taking an analytical and empirical approach and reaching theoretical boundaries to the error. Additionally, a new method to calculate Koza's performance measures is presented. It is shown that these metrics, under common experimental configurations, have an unacceptable error, which can be arbitrary large in certain conditions.

Keywords Genetic Programming, computational effort, performance measures, experimental methods, measurement error

David F. Barrero and María D. R-Moreno
Departamento de Automática, Universidad de Alcalá
E-mail: david@aut.uah.es, E-mail: mdolores@aut.uah.es

Bonifacio Castaño
Departamento de Matemáticas, Universidad de Alcalá
E-mail: bonifacio.castano@uah.es

David Camacho
Departamento de Informática, Universidad Autonónoma de Madrid
E-mail: david.camacho@uam.es

1 Introduction

Two widely known efficiency measures in Genetic Programming (GP) are the number of individuals to be processed to achieve at least one success with a certain probability, and the total computational effort. Both were defined by John R. Koza in [17] and are closely related to each other. Computational effort is indeed directly obtained as the minimum value of the number of individuals to be processed.

The accuracy and reliability of these measures have not been the object of intense investigation. Angeline first observed that the computational effort [2] is actually a random variable, and concluded that the stochastic nature of the computational effort should be handled with proper statistical tools. Some time after, [16] calculated the computational effort using confidence intervals (CIs) instead of just punctual estimation, achieving a remarkable conclusion: when success probability is low, CIs of the computational effort are almost as large as the computational effort. In order words, the variability of computational effort is similar to its magnitude, and thus, in that case is not reliable.

To the authors' knowledge, the only systematic attempt made to understand why the computational effort presents the variability observed by Keijzer was done by Christensen and Oppacher [10]. They identified three sources of variability and provided empirical data that provided some light regarding the circumstances that reduce the reliability of computational effort. More research in this area was done by [28, 27], who studied how to apply CIs to the computation of the computational effort, and by [24], who investigated the statistical properties of computational effort in steady-stade algorithms.

Our work is aligned with the research done by Christensen and Oppacher [10]. It is a systematic attempt to take a step forward to identify and characterize the computational effort. We first performed an empirical study in [6], where we reported some experimental evidence that introduced concerns about the accuracy of the computational effort. This paper is an attempt to provide a theoretical explanation of those results with an analytical model of the error associated with data, to provide with Koza's performance measures.

The main contributions of this paper are: 1) a novel method to compute performance measures; 2) a statistical model of the maximum error associated with the measurement of Koza's performance measures; 3) based on the previous result, a description of the circumstances that make those measures more (or less) reliable; 4) additional empirical evidence regarding the reliability of Koza's measures. The conclusion that can be deduced from this work is that in common experimental settings the variability of Koza's performance measures is rather high. In some circumstances, this error is asymptotically (and needlessly) high, in particular when the success probability and number of runs are low and the variability of the run-time is high. Hence, it is not suitable to use Koza's performance measures in those contexts.

The paper is structured as follows. It begins with a description of the performance measures under study, specifically, the computational effort and the number of individuals to be processed. This section includes a detailed discussion of the mathematical properties of the measures. Section 3 is an exploratory analysis that investigates the origin of randomness in the measurement of computational effort. We identify two sources of variability, the ceiling operator and the estimation error. The first one is studied in the same section. The second source of variability is much more complex, so it is analyzed in the following sections. Sections 4 and 5 deal with the effects of the estimation error in the estimation of the number of individuals to be processed and the computational effort in section 5. Section 6 provides empirical support to the analytical models developed in the previous sections. Finally, some conclusions and future work are outlined.

2 Koza's performance measures

This section describes in detail Koza's performance measures with an emphasis on the computational effort measure.

2.1 Some initial definitions

The *Success Rate* (SR) is the probability of getting a success when the algorithm is run for an infinite number of generations. The exact meaning of success depends on the problem and the objectives of the practitioner, so, depending on the context, we consider that a run has yielded a success if it satisfies a certain success predicate set by the experimenter. From the point of view of the SR, the exact form of this predicate is irrelevant as long as it clearly classifies the outcome of the run as success or failure.

Given an experiment with $y(M,i)$ successful runs in generation i, each run composed by a population of M individuals,we can define the *cumulative success probability* $P(M,i)$ as

$$P(M,i) = \frac{1}{n}\sum_{j=1}^{i} y(M,j) \tag{1}$$

Expressing $P(M,i)$ as a function of the *cumulative number of successes*, $k(M,i) = \sum_{j=1}^{i} y(M,j)$ is usually more convenient, yielding that $P(M,i) = k(M,i)/n$. We should point out that $P(M,i)$ is an observed accumulated probability, but for language abuse, it is usually omitted. It is also an estimate, so, to be strict, when $P(M,i)$ comes from an experiment, we shold denote it as $\hat{P}(M,i)$.

We have previously defined SR as the accumulated success probability at the end of the experiment, and thus $SR = \lim_{i\to\infty} P(M,i)$. However, in the general case, SR cannot be known, and it has to be estimated. The algorithm is run for a fixed number of G generations, so $k(M,i) = k(M,i+1),\ \forall i > G$, and $P(M,i)$ remains constant, so the estimation of SR is

$$\widehat{SR} = \hat{P}(M,G) = \frac{k(M,G)}{n} \tag{2}$$

The definition we have made of SR implicitly assumes that there is no guarantee that the algorithm will explore all of the search space, and therefore it might not find a solution. It seems reasonable that, under certain conditions, for instance an evolutionary algorithm (EA) with some types of mutation, given infinite time the algorithm will be able to find a solution [25]. This topic is open to theoretical discussion, and we simply assume that the algorithm might not find a solution in infinite generations.

A few words should be dedicated to the notation. We used the original notation proposed by Koza, who expressed the cumulative success probability as a function of M and i. The original intention of Koza was to emphasize the dependence of the probability on the population size and the generation. From a strict mathematical point of view, the only independent variable in the previous equations is i and, with exceptions, the population size usually does not change in the execution of an EA. The notation $P(M, i)$ might induce the idea that M is an independent variable while in general it is not, it uses a fixed value set by the experimenter before the algorithm is run, as with many other parameters in GP. Hence, in our opinion, the accumulated success probability should be expressed as $P(i)$ instead of $P(M, i)$; in the following we use this simplified notation. Another issue about notation is related to the discrete nature of EAs. The notation suggests that the performance measures are defined in continuous time, although they are discrete values (generations). In this paper we consider them as continuous. Our results will not be affected by this decision and the notation will be more consistent and clear.

2.2 A first approach to Koza's performance measures

Koza, in his classical book [17], defined a performance measure called *computational effort*. The computational effort is defined as the number of individuals that the algorithm has to process to achieve at least one success with a given probability z, expressed as $I(i, z)$. It is common to express the probability z with the geek letter ε such as $z = 1 - \varepsilon$. A common value of ε used in the literature is 0.01 ($z = 0.99$).

The number of individuals to be processed to achieve a solution with probability z at generation i is given by

$$I(i, z) = MiR(i, z), \tag{3}$$

where M is the population size and $i = 1, 2, ..., G$ the generation, thus, Mi is the number of individuals processed until generation i. Therefore Mi estimates the number of individuals that have to be evaluated in i generations, while $R(i, z)$ (or simply R using Koza's notation) is the number of runs that the experiment needs to achieve, at least, one success with probability z at generation i, and it is defined as

$$R(i, z) = \left\lceil \frac{ln(1 - z)}{ln(1 - P(i))} \right\rceil \tag{4}$$

The operator $\lceil \ldots \rceil$ is the ceiling operator and returns the smallest integer not less than its argument, i.e., it rounds up the fractional part of its argument. This operation was introduced because R gives the number of times that the experiment should be run, and thus it must be an integer. However, it should be noticed that usually it only has a mathematical interpretation, and the experiment is not supposed to be repeated R times. The importance of this observation will be evident later.

Equation (4) can be deducted directly from statistics and probability theory. A Bernoulli trial [20] is defined as an experiment whose outcome can take two random values, named "success" and "failure". Many problems in Evolutionary Computation (EC), where an optimum or near-optimum solution can be identified, may be described as a Bernoulli trial because the algorithm in those domains can achieve a satisfactory solution, or not, i.e., a "success" or a "failure" with a certain probability.

By definition, the probability of getting one success after R Bernoulli trials with success probability p is described by the geometric distribution,

$$P(X = R) = (1-p)^{R-1}p, \tag{5}$$

and the probability of getting at least one success in R trials is described by the well known cumulative distribution function (CDF) of the geometric distribution,

$$P(X \leq R) = 1 - (1-p)^R \tag{6}$$

It is interesting to point out that the geometric distribution is the only discrete distribution without memory [20]. Using the CDF of the geometrical distribution in equation (6), it is straightforward to calculate the number of trials R such as $P(X \leq R) = z$. With these considerations we can express equation (6) with the notation used by Koza.

$$z = 1 - (1-p)^R \tag{7}$$

which is the same expression that Koza deduced in [17] using probabilities. Taking natural logarithms on both sides of the equation we can isolate R

$$R \ln(1-p) = \ln(1-z) \Longrightarrow R = \frac{\ln(1-z)}{\ln(1-p)} \tag{8}$$

which is the same as (4), without the ceiling function. In any case, the number of individuals that have to be processed to achieve at least one solution with probability z, takes the form

$$I(i,z) = Mi \left\lceil \frac{ln(1-z)}{ln(1-P(i))} \right\rceil \tag{9}$$

Therefore the number of processed individuals is a function of i. By definition, the computational effort, denoted by $E(z)$ (just E with Koza's notation), is the minimum value of (9)

$$E(z) = \min_i \left\{ Mi \left\lceil \frac{ln(1-z)}{ln(1-P(i))} \right\rceil \right\} \tag{10}$$

Equations (9) and (10) are rather simple, however, understanding their behavior is not trivial. Several statistical issues arise when they are studied in detail, as will be shown later.

3 Variability sources of Koza's performance measures

Previous work performed by [10] identified three sources of variability in computational effort: the ceiling operator, the estimation of the success probability, and the minimum operator. We hold a slightly different point of view about the effects of the minimum operator. First of all, we hold that, to be strict, it is necessary to clearly distinguish between $I(i, z)$ and $E(z)$, something that some studies do not do. In our opinion, the minimum operator is a deterministic non-linear operator that removes information and therefore introduces variability, but it is the difference between $I(i, z)$ and $E(z)$. In other words, the reliability of the measurement of $E(z)$ only depends on the quality of the estimation of $I(i, z)$, which does not depend on any minimum operator. So we explicitly exclude the minimum operator in the study and decompose the problem into two. Firstly, we study the reliability of $I(i, z)$, secondly its results will be applied to study the computational effort.

Consequently, we consider two sources of randomness in $I(i, z)$ and $E(z)$: the ceiling operator and the estimation of the cumulative success probability. In order to simplify the study of the effects of these uncertainty sources, we model them as independent noise sources using the following model

$$I(i, z) = \hat{I}(i, z) + \varepsilon^{I}_{ceil} + \varepsilon^{I}_{est} \tag{11}$$

and, by definition,

$$E(z) = \min_{i} \left(\hat{I}(i, z) + \varepsilon^{I}_{ceil} + \varepsilon^{I}_{est} \right) \tag{12}$$

The error terms ε^{I}_{ceil} and ε^{I}_{est} propagate to $E(z)$ as

$$E(z) = \min_{i} \left(\hat{I}(M, i, z) \right) + \varepsilon^{E}_{ceil} + \varepsilon^{E}_{est} \tag{13}$$

So we can identify an uncertainty ε^{I}_{ceil} source generated by the ceiling operator, as well as a randomness source ε^{I}_{est} associated to the estimation of $P(i)$. This paper moves towards characterize ε^{I}_{ceil} and ε^{I}_{est}, to try to understand how they affect the accuracy of Koza's performance measures. Firstly we study the accuracy of $I(i, z)$ and then we address $E(z)$.

3.1 Ceiling operator

The first variability source we have studied is the ceiling operator. Strictly speaking, the ceiling operator is not a randomness source because it is a deterministic operator, but it removes information, increases the variability of the measure and reduces its precision, as will be demonstrated later, so its effects in practical terms are the same as a biased random error.

In order to study the effect of the ceiling operator, let us define a simplified form of the computational effort, $E_c(z)$, as $E_c(z) = \min_i (I_c(i,z))$, where $I_c(i,z) = MiR_c(i,z)$ and

$$R_c(i,z) = \frac{ln(1-z)}{ln(1-P(i))} \tag{14}$$

It is clear that, assuming $\varepsilon^I_{est} = 0$, the error term introduced by the ceiling operator in the estimation of $I(i,z)$, ε^I_{ceil}, is the difference between $I(i,z)$ and $I_c(i,z)$,

$$\varepsilon^I_{ceil} = I(i,z) - I_c(i,z) = Mi(R(i,z) - R_c(i,z)) \tag{15}$$

The error depends on the the fractional part of $R(i,z)$ that is rounded by the ceiling operator, the population size, and the generation number. We know that $(R(i,z) - R_c(i,z))$ is limited by the maximum fractional part of a real number, so $(R(i,z) - R_c(i,z)) < 1$. With this consideration, it is possible to bound ε^I_{ceil}

$$\varepsilon^I_{ceil} < Mi \tag{16}$$

It follows that $\max(\varepsilon^I_{ceil}) = Mi$. This equation introduces an absolute limit to the ceiling error which is linear to i for a given population size. One way to study the importance of this error compared with $I(i,z)$ is calculating the relative ceiling error ($\varepsilon^I_{ceil}(\%)$), which is straightforward using the definition of $I_c(i,z)$ and (16).

$$\varepsilon^I_{ceil}(\%) \leq \frac{\max(\varepsilon^I_{ceil})}{I_c(i,z)} = \frac{\ln(1-P(i))}{\ln(1-z)} \tag{17}$$

and thus the relative maximum ceiling error in the measure is a function of $P(i)$ and z. The error increases with $P(i)$ and z, and it asymptotic when $P(i) \approx 1$.

Given the reported results in this section, we conclude that the error generated by the ceiling operator might be very significant. The maximum amount of error introduced by this operator depends on the population size and the generation, nevertheless, in relative terms, it only depends on the accumulated success probability and z. We do not recommend using values of z lower that $P(i)$. Fortunately, once $P(i)$ is known, the error may be bounded by equation (17), moreover, it can be completely eliminated by simply removing the ceiling operator. We are unable to find any remarkable disadvantage, so, this evidence moves us to suggest not using the ceiling operator when calculating $I(i,z)$ and $E(z)$.

3.2 Estimation error

The second source of variability we can identify comes from the estimation of $P(i)$. The true success probability is rarely known in EC, and therefore the experimenter has to estimate it [10].

In practical terms, (9) cannot be directly used, so it has to be replaced by its estimation

$$\widehat{I}(i,z) = Mi \left\lceil \frac{ln(1-z)}{ln(1-\widehat{P}(i))} \right\rceil \tag{18}$$

where $\widehat{P}(i)$ is the estimation of the accumulated success probability

$$\widehat{P}(i) = \frac{k(i)}{n} \tag{19}$$

The difference between the theoretical $P(i)$ and the experimental $\hat{P}(i)$ is the only randomness source of $I(i,z)$, the error induced by this difference is what we name *estimation error*.

Following [10], we model the estimation error ε_{est} as a noise such as $P(i) = \hat{P}(i) + \varepsilon^P_{est}$. This error, associated with the estimation of $P(i)$, induces another error in the estimation of $I(i,z)$, that we model by adding an error term ε^I_{est}, so $I_c(i,z) = \hat{I}_c(i,z) + \varepsilon^I_{est}$. In order to isolate the effects of the estimation error and avoid unnecessary complexity, we do not consider the ceiling operator in the rest of the paper. So we use $I_c(i,z)$ instead of $I(i,z)$.

With these considerations we can state that the number of individuals to be processed is given by

$$\begin{aligned} I_c(i,z) &= Mi \frac{\ln(1-z)}{\ln(1-(\hat{P}(i)+\varepsilon^P_{est}))} \\ &= Mi \frac{\ln(1-z)}{\ln(1-\hat{P}(i))} + \varepsilon^I_{est} \end{aligned} \tag{20}$$

then, the estimation error of $I_c(i,z)$ as a function of the estimation error of the cumulative success probability is the difference between $I_c(i,z)$ and $\hat{I}_c(i,z)$

$$\begin{aligned} \varepsilon^I_{est} &= I_c(i,z) - \hat{I}_c(i,z) \\ &= \frac{Mi\ln(1-z)}{\ln(1-P(i))} - \frac{Mi\ln(1-z)}{\ln(1-(P(i)+\varepsilon^P_{est}))} \end{aligned} \tag{21}$$

To ease the interpretation of (21), it is depicted in Fig. 1 with $P(i)$ and ε^P_{est} as independent variables. The rest of the parameters are set to commonly used values, $M = 500$, $\varepsilon = 0.01$ and $i = 10$. Fig. 1 shows that ε^I_{est} has an asymptotic behaviour in two planes, $P = 0$ and $\varepsilon^P_{est} = P$. The high estimation error found in the plane $P = 0$ was previously observed in [10] using the Taylor series of equation (20) and found that $I_c(i,z)$ is very sensitive to estimation errors when $P(i)$ is close to 0, which is the situation in early generations of

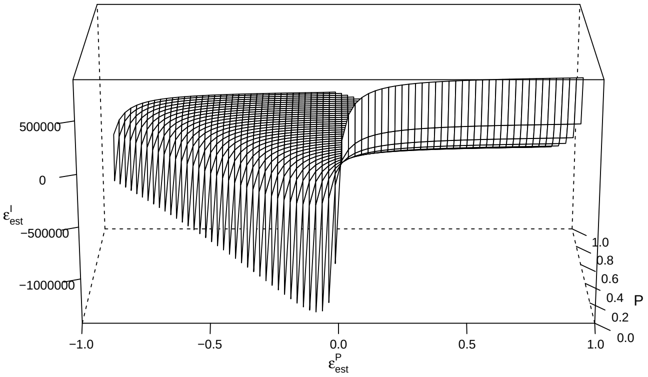

Fig. 1 Absolute error produced by the estimation of $P(i)$ as function of the success probability and the estimation error. The function is defined for $P \in (0, 1]$, $\varepsilon^P_{est} \in [-1, 1] \setminus P + \varepsilon \in (0, 1]$. M is fixed to 500, i to 10, and $\varepsilon = 0.01$.

the evolutionary process. The reason for this sensitive area can be found in the first term of (21), given $\varepsilon^P_{est} \neq 0$, and taking the limit

$$\lim_{P(i)\to 0} \left(\frac{M i \ln(1-z)}{\ln(1-P(i))} - \frac{M i \ln(1-z)}{\ln(1-(P(i)+\varepsilon^P_{est}))} \right) \tag{22}$$

yields an infinite error.

Another asymptotic error is originated by the second term of (21). When $\varepsilon^P_{est} \approx -P(i)$ the denominator tends to be 0, and then the estimation error increases its magnitude. It should be noticed that this effect is not symmetrical, it only happens for negative values of ε^P_{est}, i.e., when $P(i)$ is overestimated.

The relative estimation error is given by the ratio $\varepsilon^I_{est}/I_c(i, z)$, then, using the definition of $I_c(i, z)$ and (21)

$$\varepsilon^I_{est}(\%) = \frac{\varepsilon^I_{est}}{I_c(i,z)} \leq 1 - \frac{\ln(1-P(i))}{\ln(1-(P(i)+\varepsilon^P_{est}))} \tag{23}$$

This equation provides a way to determine whether the error is significant in relative terms. When the estimation error is small, the ratio is close to 1 and therefore the estimation error is, in proportion, close to 0%.

The relationship between ε^P_{est} and ε^I_{est} given by (23) can be better appreciated in Fig. 2. First we observe that the sign of ε^P_{est} has a strong influence on $\varepsilon^I_{est}(\%)$, an overestimation of $P(i)$ (negative ε^P_{est}) leads to more error in the calculus of $I_c(i, z)$ than an underestimation (positive ε^P_{est}) of the same magnitude. This behaviour is explained by the asymptotic effects of the relative estimation error when $P(i) + \varepsilon^P_{est} = 0$, which was seen before in Fig. 1. Secondly, we can observe that $\varepsilon^I_{est}(\%)$ also depends on the value of $P(i)$; low values of $P(i)$ are more sensitive to the estimation error than high ones, as

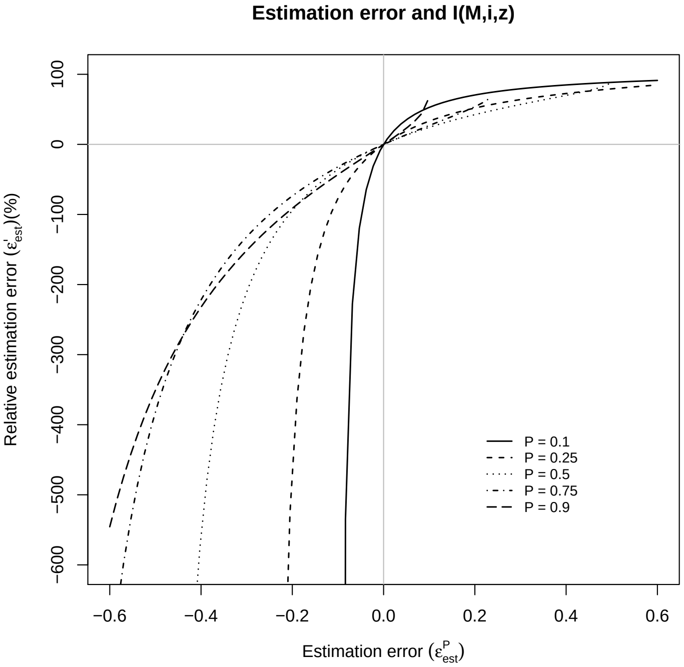

Fig. 2 Relative error as function of SR and estimation error. The function is defined for $P \in (0,1], \varepsilon^P_{est} \in [-1,1] \backslash P + \varepsilon^P_{est} \in (0,1]$.

can be seen in the slope of the curves, which are much more inclined in the former case.

As a conclusion for this section we can state that the number of processed individuals is specially sensitive to the estimation error in two cases: when the accumulated success probability is very low, close to 0, and when $\varepsilon^P_{est} \approx -P(i)$. Drawing conclusions about the relationship between the estimation error and $E(z)$ requires further analysis, which basically deals with the minimum operator. In any case, and as an almost tautological conclusion, high estimation errors will generate high errors in the computation of the number of processed individuals, and we can conjecture that this error will also be translated to the estimation of the computational effort. In this section we have related the estimation error of $P(i)$, ε^P_{est}, to the error that it introduces in ε^I_{est}, however, it is still unclear which factors determine the value of ε^P_{est}. The next section addresses this problem using binomial CIs.

4 Characterization of the estimation error with confidence intervals

The magnitude of the estimation error of $P(i)$ is a key element to explain the accuracy of Koza's performance measures. Due to the stochastic nature of ε^P_{est}, it is not possible to set a hard limit to its size, as was done with the ceiling error, nonetheless, it does not mean that there are no mathematical tools that could shed some light on this topic. The maximum likelihood estimator of the accumulated success probability is given by $\hat{P}(i) = k(i)/n$. Given a certain generation, let say i_0, the number of successful runs $k(i_0)$ is, by definition, a binomial random variable [5,22]. Then, ε^P_{est} is an error associated with the estimation of a binomial variable, and thus, it can take any value between 0 to 1. Nonetheless, it is still possible to determine a region where the estimation of the probability is likely to be contained with binomial CIs [7,23] and identify that region with ε^P_{est}.

The magnitude of the uncertainty associated with the estimation of the success probability at time i_0 can be associated with the *Confidence Interval Width*, or CIW. The CIW of an interval $[L, U]$ is defined as the difference between its upper and lower bounds [7,23], so $CIW = U - L$. Any binomial CI method may be used, for instance, a normal approximation or Wald interval [18], Clopper-Pearson or "exact" interval [11], Agresti-Coull or adjusted Wald [1], Wilson or 'score' [29], not to mention alternative Bayesian approaches [13,26]. No matter which method is used, binomial CIs can be used to characterize the magnitude of ε^P_{est}, and therefore also to characterize how this error is propagated to the estimation of $I(i, z)$ and $E(z)$.

4.1 Absolute estimation error of $I(i, z)$

There are several binomial CI methods, each one with its own properties. However, the basic properties are common for all the methods, and only a thorough analysis of the methods may find differences in their behaviour. In any case, for the purpose of this research, these differences are not significant, we are interested in the common properties of binomial CIs to characterize ε^P_{est}, not in the particularities of each method. Several authors [5,28,7,21] recommend the classical Wilson method [29]. This method combines good performance in average terms with simplicity, which makes Wilson a good choice to characterize ε^P_{est}. Specifically, we use Wilson with continuity correction, that corrects some aberrations found in the original method [8], so the form of the CI $[L, U]$ used in this study is

$$\begin{aligned} L &= \frac{k + \frac{1}{2} z^2_{\alpha/2}}{n + z^2_{\alpha/2}} - \frac{z^2_{\alpha/2}\sqrt{n}}{n + z^2_{\alpha/2}} \sqrt{p(1-p) + \frac{z^2_{\alpha/2}}{4n}} \\ U &= \frac{k + \frac{1}{2} z^2_{\alpha/2}}{n + z^2_{\alpha/2}} + \frac{z^2_{\alpha/2}\sqrt{n}}{n + z^2_{\alpha/2}} \sqrt{p(1-p) + \frac{z^2_{\alpha/2}}{4n}} \end{aligned} \tag{24}$$

where $p = k/n$ is the maximum likelihood estimator of the success probability, k is the number of successes, n is the number of runs and $z_{\alpha/2}$ is the upper-$\alpha/2$ critical point from $N(0, 1)$.

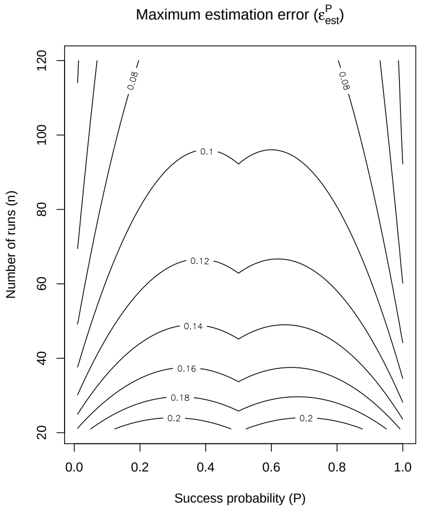

Fig. 3 Maximum estimation error ε^{P}_{est} as function of the number of runs and the success probability. ε^{P}_{est} has been calculated as the maximum of Wilson DCIW and MCIW with $\alpha = 0.05$ (see equation (25)).

The effects of the asymmetry of the estimation of a probability can be studied using the distance between $\hat{p}$ and the boundaries of the interval $[L, U]$. Inspired by [23], and as a way to measure the asymmetry of the interval, we define the *Distal Confidence Interval Width*, or DCIW, as the difference between the maximum likelihood estimator of the probability and the lower limit, so, $DCIW = \hat{p} - L$. Similarly, we define the *Mesial Confidence Interval Width*, or MCIW, as the difference $MCIW = U - \hat{p}$. In the same way, the *Confidence Interval Width* is the width of the interval, $CIW = U - L$. It is trivial to demonstrate that CIW, DCIW and MCIW satisfy the property $CIW = DCIW + MCIW$. In order to be conservative, we consider the maximum between DCIW and MCIW, so

$$\varepsilon^{P}_{est} \leq \max(DCIW, MCIW) \tag{25}$$

To better illustrate the properties of (25), it has been depicted in Fig. 3. The error ε^{P}_{est} is symmetrical with respect the axis $P = 0.5$, however its maximum is not placed on that axis, it is biased with respect to it instead. The reason can be found in the displacement of $\hat{p}$ with respect to the center of the interval. Interestingly, the effect of the asymmetry of the interval is less evident as the number of runs is increased. In any case, higher number of samples always generates tighter intervals because there is more information about the algorithm [3], more precise intervals can be built and therefore the estimation error is reduced.

We observe in Fig. 3 that for a usual experimental design with $n = 50$, the estimation error associated to $P(i)$ is, in the worst case, $\varepsilon^{P}_{est} \lesssim 0.12$. We should stress that this is not a hard limit, it is instead probabilistic with $\alpha = 0.05$ and therefore we should expect runs with higher estimation errors.

In this section we have developed some tools in order to help us to understand under which circumstances the estimation error is higher. However, the question about how these circumstances affect the estimation of $I(i,z)$ and $E(z)$ remains open.

4.2 Relative estimation error of $I_c(i,z)$

The main variability source of $I(i,z)$ is the estimation error associated with the measurement of $P(i)$. Due to the intrinsic stochastic nature of the estimation, it is necessary to use statistical methods to characterize its behaviour. Binomial statistics provide a tool to estimate ε_{est}^{P} when i is fixed as a function of the number of runs and the success probability. Since ε_{est}^{I} is a function of ε_{est}^{P}, we can use the previous result to deduce a relationship among ε_{est}^{I}, n and p.

With all these considerations, and assuming a symmetrical interval for simplicity, we can limit the effects of the estimation error in the measurement of $I(i,z)$ as the difference of $I(i,z)$ calculated in the boundaries of the interval,

$$\varepsilon_{est}^{I} \leq \frac{1}{2}\left(Mi\frac{\ln(1-z)}{\ln(1-L_i)} - Mi\frac{\ln(1-z)}{\ln(1-U_i)}\right) \tag{26}$$

where $[L_i, U_i]$ is the Wilson CI of the success probability at generation i.

In order to analyze the relative effects of the estimation error we calculate the relative maximum error $\varepsilon_{est}^{I}(\%)$ as the ratio $\frac{\varepsilon_{est}^{I}}{I(i,z)}$, with $I(i,z)$ evaluated for the probability $\tilde{p} = (k+\frac{1}{2}z_{\alpha/2}^{2})(n+z_{\alpha/2}^{2})^{-1}$. In this way $\varepsilon_{est}^{I}(\%)$ is characterized as

$$\varepsilon_{est}^{I}(\%) \leq \frac{\ln(1-\tilde{p})}{2}\left(\frac{1}{\ln(1-L_i)} - \frac{1}{\ln(1-U_i)}\right) \tag{27}$$

The surface defined by this equation for $\alpha = 0.05$ is depicted in Fig. 4. It shows that $\varepsilon_{est}^{I}(\%)$ is highly dependent on the number of runs and success probability. Using a high number of runs yields less error in the measurement of $I(i,z)$. The influence of the success probability is slightly more complicated. Low values of $P(i)$ yields poor estimations of $I(i,z)$, the same can be said when $P(i)$ is close to 1, however, in this case the effect is not so evident.

Let us consider a common experimental setup composed of 60 runs, a value within an order of magnitude commonly used in practice. Looking at Fig. 4 we find that the maximum relative estimation error is, at least, around 34% when $P(i) = 0.78$; this error does not include the error produced by the ceiling operator, and actually, it is not guaranteed that the experiment would achieve it, for instance, in case that the SR achieved by the algorithm were lower than 0.78. It shows that the estimation error is rather significant even for a relative high number of runs, 60, moreover, finding literature reporting fewer runs is not rare. We should point out that this brief discussion is quite conservative since it uses $\alpha = 0.05$.

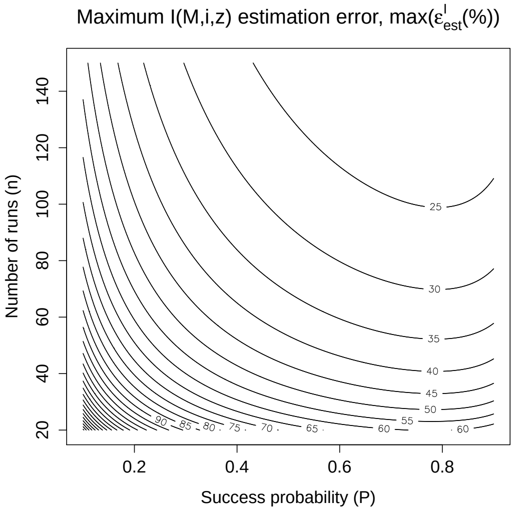

Fig. 4 Maximum relative error of $I(i, z)$ calculated using confidence intervals with $\alpha = 0.05$. X-axis represents the success probability (p) whereas y-axis represents the number of runs (n).

5 Characterization of the estimation error of $E(z)$

The main difficulty that we find in the study of $E(z)$ is that, unlike $I(i, z)$, its statistical properties do not depend on an underlying binomial random variable, but on a stochastic process. Therefore, we need a model of the time-behaviour of $I(i, z)$, which also depends on the existence of an analytical model of $P(i)$. Fortunately, this problem has been addressed in the literature and therefore we are in position to develop a model of the computational effort that can be used to estimate the error associated with its measurement.

5.1 Analytical model of $E(z)$

In [4], the authors proposed an analytical model of $P(i)$ based on the statistical modelization of the run-time to success. Let us name the model of the success probability $P^{\star}(i)$. This model decomposes $P(i)$ into two terms, one expresses the probability of finding a solution given that the algorithm has been run for G generations, and another term models the probability of finding the solution before generation i. The subsequent model yields as follows:

$$P^{\star}(i) = SR\, \Phi\left(\frac{\ln i - \mu}{\sigma}\right) \tag{28}$$

where the SR can be estimated as $\widehat{SR} = \frac{k(G)}{n}$, $\Phi(...)$ is the standard normal CDF [15], μ and σ are, respectively, the mean and variance of the lognormal distribution, and they can be estimated as

$$\hat{\mu} = \frac{\sum_{k=1}^{m} \ln g_k}{m} \tag{29}$$

and

$$\hat{\sigma} = \sqrt{\frac{\sum_{k=1}^{m} (\ln g_k - \hat{\mu})^2}{m}} \tag{30}$$

The set $\{g_1, g_2, ..., g_k\}$ denotes the generation where successful run $j \in \mathcal{N}^+ \leq k$ has found a solution.

Experiments involving some classical problems in tree-based GP showed that the lognormal distribution is a reasonable choice to model the run-time behaviour of the algorithm [4]. Additionally, there is literature in related areas supporting the idea that the lognormal distribution is a reasonable statistical distribution to model the time that the algorithm requires to find a solution. In particular, the lognormality of this random variable has been reported in areas like Ant Colonies Optimization, Simulated Annealing, Iterated Local Search, Random Restart Local Search [9,14] or backtracking algorithms in CSP problems [12].

It seems that equation (28) provides a reasonable model of the accumulated success probability in tree-based GP. Assuming this model, we can provide a new method to compute performance related to $I_c(i, z)$. Let us name this method $I_c^\star(i, z)$. Using equations (28) and (9) we trivially obtain that

$$I_c^\star(i, z) = Mi \frac{\ln(1 - z)}{\ln(1 - P^\star(i))} \tag{31}$$

Then, $I_c^\star(i, z)$ is a function of three parameters, SR, μ and σ, however this fact is not reflected in the notation, that for consistency, we maintain in this section. Once we have $I_c^\star(i, z)$, the analytical model of $E(z)$, let us name it $E_c^\star(z)$, yields as $E_c^\star(z) = \min_i \{I_c^\star(i, z)\}$. Similarly to $I_c^\star(i, z)$, $E_c^\star(z)$ is a function of three parameters, SR, μ and σ.

5.2 Analytical model of the estimation error of $E(z)$

The model described by equation (28) is a function of three variables, which are SR, μ and σ. Since the only variables of $E(z)$ are the same as those found in (28), we express $E(z)$ as $E(SR, \mu, \sigma)$. The effects of the estimation error of SR has been partially studied in [3], so we exclude this factor. The way that the estimation of μ and σ can affect the reliability of computational effort is still unknown, so, in the following we focus our investigation on the study of these two factors.

Given the analytical complexity of the model we simplify the problem using a numerical approach. A crude, but reasonable way, to estimate the relative error $\varepsilon_{est}^{E}(\%)$ numerically is just observing the effect of adding to $E(SR, \mu, \sigma)$ some noise, $E(SR, \mu + \varepsilon_\mu, \sigma + \varepsilon_\sigma)$, which introduces an error $\varepsilon_{est}^{E}(\%)$ to the estimation. So, the relative estimation error can be modeled as

$$\varepsilon_{est}^{E}(\%) = \frac{E(SR, \mu, \sigma) - E(SR, \mu + \varepsilon_\mu, \sigma + \varepsilon_\sigma)}{E(SR, \mu, \sigma)} \tag{32}$$

The problem here is which values ε_μ and ε_σ should be used. CIs provide a tool to overcome this problem. To be more specific, the CIW can be seen as an uncertainty region, a region where a measure can be located, and therefore we can use it to provide a probabilistic bound to the error associated to the measurement of $\hat{\mu}$ and $\hat{\sigma}$. If we do not consider the SR, there are two variables involved in the estimation of $E^\star$, and thus from a geometrical perspective, their estimation define an uncertainty surface whose boundaries are defined by CIs.

Determining the uncertainty region of the estimation of computational effort in the point (SR, μ, σ), for a fixed value of SR, becomes a problem of computing lognormal CIs. In particular, it is possible to estimate the magnitude of ε_μ and ε_σ with CIs using the relationship between the normal and lognormal distributions [19], which is given by the expression

$$X \sim N(\mu, \sigma) \Rightarrow e^X \sim LN(\mu_L, \sigma_L) \tag{33}$$

$$X \sim LN(\mu_L, \sigma_L) \Rightarrow \ln(X) \sim N(\mu, \sigma) \tag{34}$$

This relation is handy because it provides a way to apply all the normal statistics to lognormal distributions, including normal CIs. It is well known that the CI of the mean $[\mu^-, \mu^+]$, given an unknown σ, is

$$[\mu^-, \mu^+] = \left[\hat{\mu} - t_{\alpha/2,n-1}\frac{\hat{\sigma}}{\sqrt(n)}, \hat{\mu} + t_{\alpha/2,n-1}\frac{\hat{\sigma}}{\sqrt(n)}\right] \tag{35}$$

where $t_{\alpha/2,n-1}$ is the upper $(1-\alpha)/2$ critical value for the Students' t distribution with $(n-1)$ degrees of freedom, and n the number of samples (or runs in the context of EC). Similarly, the normal CI $[\sigma^-, \sigma^+]$ when the mean is unknown is given by

$$[\sigma^-, \sigma^+] = \left[\sqrt{\frac{(n-1)\hat{\sigma}^2}{\chi_{\alpha/2,n-1}}}, \sqrt{\frac{(n-1)\hat{\sigma}^2}{\chi_{1-\alpha/2,n-1}}}\right] \tag{36}$$

where $\chi_{\alpha/2,n-1}$ and $\chi_{1-\alpha/2,n-1}$ are the upper and lower critical values for the χ-squared distribution with $(n-1)$ degrees of freedom.

The CIs defined in (35) and (36) provide the limits of the uncertainty region $\Omega = [\mu^-, \mu^+] \times [\sigma^+, \sigma^-]$ in the domain such as $\forall(\mu', \sigma') \in \Omega$, the difference $|E(SR, \mu, \sigma) - E(SR, \mu', \sigma')|$ is defined. Then, given that Ω is a compact region, there is a point $(\mu'', \sigma'') \in \Omega$ that maximizes the difference. Hence, for each point (μ, σ), we determine its uncertainty region and estimate the relative error as

$$\varepsilon^E_{est}(\%) = \frac{E(SR, \mu, \sigma) - E(SR, \mu'', \sigma'')}{E(SR, \mu, \sigma)} \tag{37}$$

We should stress that the model of error described by (37) assumes a symmetrical error, where the effects of underestimating $E(z)$ is similar to the effects of overestimating $E(z)$. As suggested by the model $\varepsilon^I_{est}(\%)$ in (23), this

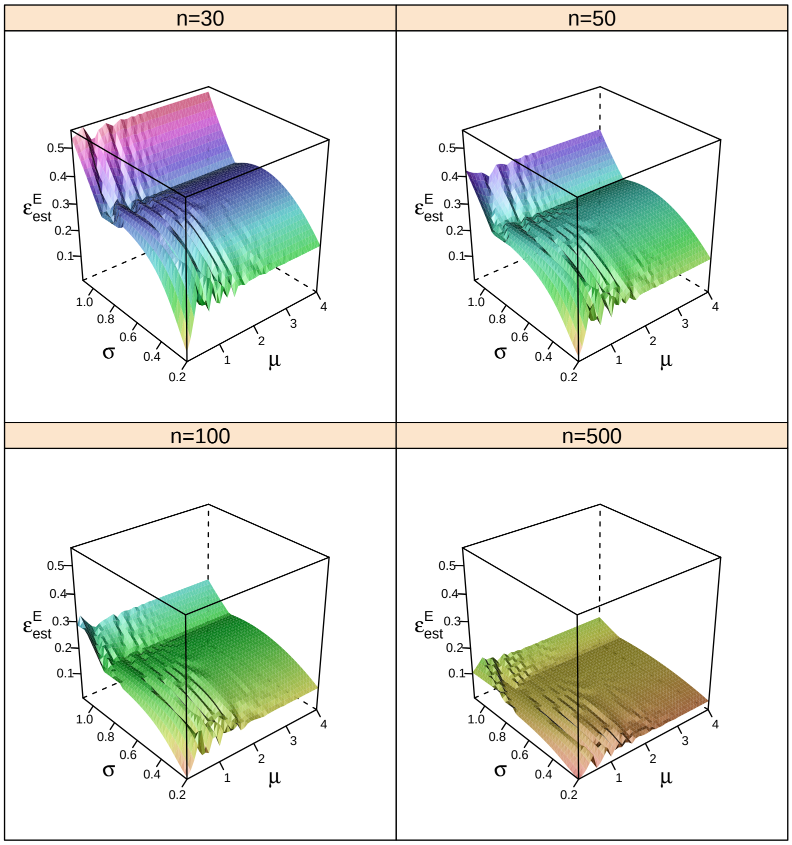

Fig. 5 Maximum expected relative estimation error of $E(z)$ as function of μ, σ and n, as modeled by (37). The number of runs takes values $n \in \{30, 50, 100, 500\}$, CIs were computed with $\alpha = 0.05$ and $E(z)$ with $\varepsilon = 0.05$. The SR is, in all the cases, 0.5. Different SR and ε values do not yield remarkable differences.

assumption does not hold true for $I(i, z)$. This fact might limit the accuracy of the error model.

The analytical form of ε^E_{est} is complex enough to dissuade its direct analysis, its representation is given in Fig. 5. The figure plots the maximum expected relative error calculated with $\alpha = 0.05$ and $SR = 0.5$ for a domain that covers the values that we have found empirically (see Table 2). SR has not been included in this study, however, we have verified that other values of SR do not yield remarkable differences.

Fig. 5 shows that the effect of μ is pretty moderate, and we only can observe a dependence of the relative error with μ in form of smooth oscillations for $0 < \mu < 2$. Interestingly, these oscillations tend to vanish as n increases. The relative error depends much more on σ in a non trivial way. When σ is low, the relative error seems to stay low, but as it is increased, the error also increases. The effect of σ on the error is influenciated by the number of runs, the higher is n, the lower is the influence of σ, however, even with large values of n, a high dispersion of the run-time to success has a negative impact on the accuracy of the measure. The proposed model forecasts, as could be expected, that the number of runs has a direct impact on the quality of the measure, higher number of runs produce better estimations of $E(z)$.

The behaviour of the maximum expected error is interesting, but its magnitude is a major concern in order to determine the reliability of $E(z)$. The error model forecasts poor accuracy unless a very high number of runs is executed, or some special -and unrealistic- conditions are satisfied. Following the model, the error tends to behave well when μ and σ are close to 0 regardless of n, which is clearly an unrealistic situation. In other scenarios, the accuracy of $E(z)$ is worse.

In a realistic experiment with 30 runs, $SR = 0.5$, $\mu = 2.5$ and $\sigma = 0.5$, the maximum error forecasted by the model proposed in (37) yields a maximum relative error around 0.33. It is hard to imagine a scenario where this error was not significant. However, this value can be reduced by increasing the number of runs. Adding 20 runs $\varepsilon_{est}^{E}(\%)$ is notably reduced to 0.25. If the experiment is composed of 100 runs the maximum estimation error values 0.16, while with $n = 500$ $\varepsilon_{est}^{E}(\%) = 0.07$. Depending on the context of the experiment, these values might be, or not, significant, however *a priori* they seem to be rather high.

6 Experimental evaluation

In order to measure Koza's computational effort error and verify to what extent our error model is realistic, we have carried out some experiments comparing the theoretical and experimental error. These experiments rely on four classical GP problems (artificial ant with the Santa Fe trail, 6-multiplexer, 4-parity and symbolic regression) run a large number of times followed by resampling to simulate runs. The main parameters for the problems under consideration are in Table 1.

In order to clarify the terminology used in this section, we define a *run* as a single execution of an algorithm and an *experiment* as a collection of $n > 0$ independent runs. Due to the random nature of EAs, many of their properties are stochastic, and thus they cannot be characterized using a single run, but instead by an experiment with several runs. We use the term *pseudoexperiment* to mean a simulated experiment carried out by resampling a dataset of runs.

Firstly, we tried to get an accurate estimation of the parameters needed by the model. It was simply done by using all the runs in the dataset to estimate SR, μ and σ. We named those statistics $(\widehat{SR}, \hat{\mu}_{best}, \hat{\sigma}_{best})$. They are shown in Table 2. It is interesting to compare the error measured in the experiment to the value that the model forecasts, which is also shown in Table 2. It can be seen that the model $(\hat{E}^{\star}_{best})$ does not perfectly match the original method

Table 1 Tableau for the problems under study: artificial ant with the Santa Fe Trail, 6-multiplexer, even 4-parity and symbolic regression without ERC. These parameters are almost the default ones in ECJ v18 with some minor modifications.

Parameter	Santa Fe Trail	6-multiplexer	4-Parity	Regression
Population	500	500	4,000	500
Generations	50	50	800	50
Terminal Set	Left, Right, Move, IfFoodAhead	A0, A1, A2, D0, D1, D2, D3, D4, D5	D0, D1, D2, D3, D4	X
Function set	Progn2, Progn3, Progn4	And, Or, Not, If	And, Or, Nand, Nor	Add, Mul, Sub, Div, Sin, Cos, Exp, Log
Success predicate	Best $fitness = 0$	Best $fitness = 0$	Best $fitness = 0$	Best $fitness \leq 0.001$
Initial depth	5	5	5	5
Max. depth	17	17	17	17
Selection	Tournament (size=7)	Tournament (size=7)	Tournament (size=7)	Tournament (size=7)
Crossover	0.9	0.9	0.9	0.9
Reproduction	0.1	0.1	0.1	0.1
Terminals	0.1	0.1	0.1	0.1
Non terminals	0.9	0.9	0.9	0.9
Observations	Timesteps=600		Even parity	No ERC $y = x^4 + x^3 + x^2 + x$ $x \in [-1, 1]$

Table 2 Best estimations of the parameters of the error model, comparison between $\hat{E}_{best}$ and $\hat{E}^{\star}_{best}$, and their relative estimation error, both, empirical ($\hat{E}_{best} - \hat{E}^{\star}_{best}/\hat{E}_{best}$) and theoretical ($\varepsilon^{E}_{est}(\%)$). Computational effort has been computed with $\varepsilon = 0.05$.

	$\widehat{SR}$	$\hat{\mu}_{best}$	$\hat{\sigma}_{best}$	$\hat{E}^{\star}_{best}$	$\hat{E}_{best}$	$\hat{E}_{best} - \hat{E}^{\star}/\hat{E}_{best}$	$\varepsilon^{E}_{est}(\%)$
Artificial ant	0.132	2.74	0.59	324,712	316,980	0.024	0.013
6-Multiplexer	0.956	2.46	0.43	14,886	14,836	0.003	0.006
4-Parity	0.7475	5.00	0.8	3,637,999	3,660,603	0.062	0.132
Regression	0.295	2.29	0.44	79,470	75,764	0.049	0.007

($\hat{E}_{best}$), however, it provides a similar result. There is an exception with the regression problem, where a quite significant difference is found. Three out of the four problems yield a theoretical relative error that doubles or halves the empirical one. Given that we do not aim to exactly model the estimation error but rather to understand it, we consider that the result is moderately satisfactory.

In order to verify the accuracy of $E(z)$ and whether $\varepsilon^{E}_{est}(\%)$ models it well, we have run 500 pseudoexperiments with $n \in \{50, 100, 200, 300, 400, 500\}$, computing $\hat{E}^{\star}$ and $\varepsilon^{E}_{est}(\%)$ using $\hat{E}_{best}$ to compare to, yielding the results shown in Fig 6. The boxplot shows empirical data while the continuous line represents the theoretical error bound computed with $\alpha = 0.01$. As one could expect, the error always decreases as n increases. The magnitude of the error is more interesting. It depends on the problem, but for $n = 50$, a common

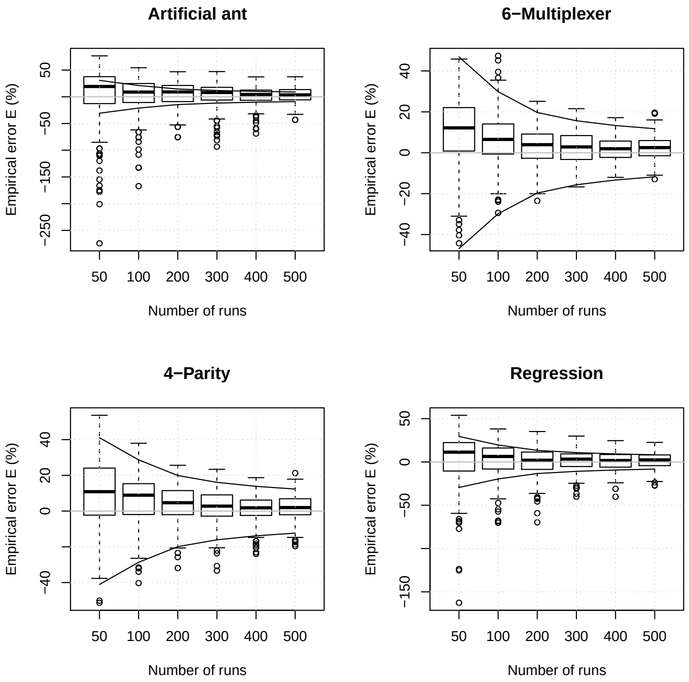
Artificial ant
6−Multiplexer
4−Parity
Regression

Fig. 6 Boxplot with the empirical relative estimation error of E compared to the boundaries of the error, in continuous lines, calculated with the error model described by equation (37). CIs used $\alpha = 0.05$ and the parameters shown in Table 2.

value, the experiment shows that half of the pseudoexperiments contains an error between 0 and 25%, which is quite significant in most contexts.

The error model is successful at limiting, to some extent, the estimation error. However, this success is not equal for all the problems, in particular the model fails in the artificial ant problem, and its accuracy is limited in the regression problem, which are the two problems with lower SR. It can be explained by the exclusion of the SR in this error model. Curiously, the model fails in the two problems with low SR. In these cases the estimation of the SR is more prone to be far from the true value, and its effect is more critical, as previously seen. This issue is not considered by the model, and it seems reasonable to assume that the deviation of the model from the experimental data is introduced by the error in the estimation of the SR.

7 Conclusions

In this paper we have tried to provide some clues to better understand Koza's performance measures, their behaviour and reliability. In short, we can identify one source of variability, the ceiling operator, and one source of randomness, the estimation of success probability. We studied their effect in $I(i, z)$. The ceiling operator introduces a maximum relative error that is arbitrary high, depending on the success probability. It is possible to remove the operator without significant drawbacks. So, we recommend, in the same line as other previous authors, not using the ceiling operator.

The only source of randomness in the measurement of $I(i, z)$ and $E(z)$ is introduced by the estimation of the success probability. An estimation of this error can be done using binomial CIs. Basically, the quality of the estimation of a success probability depends on two factors: the number of runs and the value of the probability. The worst scenario is estimating a success probability close to 0 or 1 with a low number of runs, in that case the estimations are very unreliable. This is just the scenario found in early stages of the EA, and unfortunately there is only one method to improve the reliability of the measure: increasing the number of runs.

In order to study the reliability of $E(z)$, we have developed analytically (and empirically validated) a statistical limit to the estimation error of $E(z)$. This model uses the lognormal nature that the runtime-to-success in tree-based GP appears to have. Theory and experiments showed that, for common experimental designs, the error associated to $E(z)$ is not negligible. The reason for the poor performance of computational effort can be explained by its non-linearity: small estimation errors of the success probability, under certain conditions, are amplified to a point that the measure is no longer reliable.

It seems reasonable to ask why it introduces all this complexity, and what is the advantage of using this measure. Koza justified it as a way to take into consideration not only the time required to find the solution, but also the population size, which determines the resources wasted in the search. From our point of view, the population size is another parameter -certainly a critical one- and should be reported separately. Probably the average number of evaluations used to achieve the solution, provides, at least, the same information, without any of the drawbacks previously described. Another viable alternative would be to report the success probability and the population size, in this way we avoid the non-linear effects, providing a more reliable information about the algorithm.

For all these reasons, and as a general conclusion of our work, under the light given by the evidence reported in this paper, we suggest avoiding the use of computational effort. In our opinion, it is unnecessarily complex and unreliable. Based on the Occams razor principle, we suggest using alternative measures such as the success probability or the average number of evaluations.

Acknowledgements Authors would like to thank Héctor Menéndez, Alejandro Sierra and Ricardo Aler for their reviews and valuable suggestions.

References

1. Agresti, A., Coull, B.A.: Approximate is better than 'exact' for interval estimation of binomial proportions. The American Statistician **52**, 119–126 (1998)

2. Angeline, P.J.: An investigation into the sensitivity of genetic programming to the frequency of leaf selection during subtree crossover. In: Proceedings of the First Annual Conference on Genetic Programming (GECCO 1996), pp. 21–29. MIT Press, Cambridge, MA, USA (1996)
3. Barrero, D.F., Camacho, D., R-Moreno, M.D.: Confidence Intervals of Success Rates in Evolutionary Computation. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation (GECCO 2010), pp. 975–976. ACM, Portland, Oregon, USA (2010). DOI http://doi.acm.org/10.1145/1830483.1830657
4. Barrero, D.F., no, B.C., R-Moreno, M.D., Camacho, D.: Statistical Distribution of Generation-to-Success in GP: Application to Model Accumulated Success Probability. In: S. Silva, J.A. Foster, M. Nicolau, M. Giacobini, P. Machado (eds.) Proceedings of the 14th European Conference on Genetic Programming, EuroGP 2011, *LNCS*, vol. 6621, pp. 155–166. Springer-Verlag, Turin, Italy (2011)
5. Barrero, D.F., R-Moreno, M., Camacho, D.: Improving experimental methods on success rates in evolutionary computation. Journal of Experimental & Theoretical Artificial Intelligence (To appear) (2013)
6. Barrero, D.F., R-Moreno, M.D., Castano, B., Camacho, D.: An empirical study on the accuracy of computational effort in genetic programming. In: A.E. Smith (ed.) Proceedings of the 2011 IEEE Congress on Evolutionary Computation, pp. 1169–1176. IEEE Computational Intelligence Society, IEEE Press, New Orleans, USA (2011)
7. Brown, L.D., Cai, T.T., Dasgupta, A.: Interval Estimation for a Binomial Proportion. Statistical Science **16**, 101–133 (2001)
8. Brown, L.D., Cai, T.T., Dasgupta, A.: Confidence Intervals for a Binomial Proportion and Asymptotic Expansions. Annals of Statistics **30**(1), 160–201 (2002)
9. Chiarandini, M., Stützle, T.: Experimental evaluation of course timetabling algorithms. Tech. Rep. AIDA-02-05, Intellectics Group, Computer Science Department, Darmstadt University of Technology, Darmstadt, Germany (2002)
10. Christensen, S., Oppacher, F.: An Analysis of Koza's Computational Effort Statistic for Genetic Programming. In: Proceedings of the 5th European Conference on Genetic Programming (EuroGP 2002), pp. 182–191. Springer-Verlag, London, UK (2002)
11. Clopper, C., Pearson, S.: The use of confidence or fiducial limits illustrated in the case of the binomial. Biometrika **26**, 404–413 (1934)
12. Frost, D., Rish, I., Vila, L.: 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'97/IAAI'97, pp. 327–333. AAAI Press (1997)
13. Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B.: Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science), 2 edn. Chapman and Hall/CRC (2003)
14. Hoos, H.H., Sttzle, T.: Evaluating Las Vegas algorithms – pitfalls and remedies. In: In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI-98), pp. 238–245. Morgan Kaufmann Publishers (1998)
15. Kaufmann, A., Grounchko, D., Cruon, R.: Mathematical Models for the Study of the Reliability of Systems, *Mathematics in Science and Engineering*, vol. 124. Academic Press, Inc. (1977)
16. Keijzer, M., Babovic, V., Ryan, C., O'Neill, M., Cattolico, M.: Adaptive logic programming. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001), pp. 42–49. Morgan Kaufmann, San Francisco, California, USA (2001)
17. Koza, J.: Genetic Programming: On the programming of Computers by Means of Natural Selection. MIT Press, Cambrige, MA (1992)
18. Laplace, P.S.: Théorie Analytique des probabilités. Mme. Ve Courcier, Paris, France (1812)
19. Limpert, E., Stahel, W.A., Abbt, M.: Log-normal distributions across the sciences: Keys and clues. BioScience **51**(5), 341–352 (2001)
20. Montgomery, D.C., Runger, G.C.: Applied Statistics and Probability for Engineers, 4th Edition, 4th edn. John Wiley & Sons (2006)
21. Mouret, J.B., Doncieux, S.: Encouraging behavioral diversity in evolutionary robotics: an empirical study. Evolutionary computation **20**(1), 91–133 (2012)

22. Myers, R., Hancock, E.R.: Empirical Modelling of Genetic Algorithms. Evolutionary Computation **9**(4), 461–493 (2001)
23. Newcombe, R.G.: Two-sided confidence intervals for the single proportion: comparison of seven methods. Statistics in Medicine **17**(8), 857–872 (1998)
24. Niehaus, J., Banzhaf, W.: More on Computational Effort Statistics for Genetic Programming. In: Genetic Programming, Proceedings of EuroGP'2003, *LNCS*, vol. 2610, pp. 164–172. Springer-Verlag, Essex (2003)
25. Poli, R., Vanneschi, L., Langdon, W., McPhee, N.: Theoretical results in Genetic Programming: the next ten years? Genetic Programming and Evolvable Machines **11**(3), 285–320–320 (2010)
26. Sharma, R.: Bayes approach to interval estimation of a binomial parameter. Annals of the Institute of Statistical Mathematics **27**(1), 259–267 (1975)
27. Walker, M., Edwards, H., Messom, C.: The reliability of confidence intervals for computational effort comparisons. In: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation (GECCO 2007), pp. 1716–1723. ACM, New York, NY, USA (2007)
28. Walker, M., Edwards, H., Messom, C.H.: Confidence Intervals for Computational Effort Comparisons. In: EuroGP, pp. 23–32 (2007)
29. Wilson, E.B.: Probable inference, the law of succession, and statistical inference. Journal of the American Statistical Association (22), 309–316 (1927)