

Article

# Forecasting the Project Duration Average and Standard Deviation from Deterministic Schedule Information

Pablo Ballesteros-Pérez <sup>1,\*</sup>, Alberto Cerezo-Narváez <sup>1,\*</sup>, Manuel Otero-Mateo <sup>1</sup>,  
Andrés Pastor-Fernández <sup>1</sup>, Jingxiao Zhang <sup>2,\*</sup> and Mario Vanhoucke <sup>3,4,5</sup>

<sup>1</sup> Escuela Superior de Ingeniería, Universidad de Cádiz, 11519 Puerto Real (Cádiz), Spain; manuel.otero@uca.es (M.O.-M.); andres.pastor@uca.es (A.P.-F.)

<sup>2</sup> School of Economics and Management, Chang'an University, Xi'an 710064, China

<sup>3</sup> Department of Business Informatics and Operations Management, Ghent University, 9000 Gent, Belgium; mario.vanhoucke@ugent.be

<sup>4</sup> Vlerick Business School, 9000 Gent, Belgium

<sup>5</sup> UCL School of Management, University College London, London E14 5AA, UK

\* Correspondence: pablo.ballesteros@uca.es (P.B.-P.); alberto.cerezo@uca.es (A.C.-N.); zhangjingxiao@chd.edu.cn (J.Z.); Tel.: +34-956-483-200 (P.B.-P. & A.C.-N.); +86-029-8233-4383 (J.Z.)

Received: 10 November 2019; Accepted: 13 January 2020; Published: 16 January 2020



**Abstract:** Most construction managers use deterministic scheduling techniques to plan construction projects and estimate their duration. However, deterministic techniques are known to underestimate the project duration. Alternative methods, such as Stochastic Network Analysis, have rarely been adopted in practical contexts as they are commonly computer-intensive, require extensive historical information, have limited contextual/local validity and/or require skills most practitioners have not been trained for. In this paper, we propose some mathematical expressions to approximate the average and the standard deviation of a project duration from basic deterministic schedule information. The expressions' performance is successfully tested in a 4100-network dataset with varied activity durations and activity durations variability. Calculations are quite straightforward and can be implemented manually. Furthermore, unlike the Project Evaluation and Review Technique (PERT), they allow drawing inferences about the probability of project duration in the presence of several critical and subcritical paths with minimal additional calculation.

**Keywords:** project duration; scheduling; merge event bias; construction; PERT

## 1. Introduction

Projects ending later and costing more than planned are widespread phenomena in the construction industry [1,2]. Numerous causes were identified in the construction management literature [3–5]. Among them, one cause that is persistently identified in late projects is poor planning and project control practices [6].

Paradoxically, extensive research focused for quite some time on the development of enhanced project planning tools, such as the project evaluation and review technique (PERT) [7], stochastic network analysis (SNA) [8], and artificial intelligence and statistical learning techniques [9]. Additionally, in recent years, project monitoring and control techniques also gained more research attention, including critical chain management [10], earned value management [11], and schedule risk analysis [12].

However, most of the advanced techniques above were not successfully adopted by construction management practitioners [13]. Currently, classical deterministic scheduling techniques such as the Gantt chart [14] and the critical path method (CPM) [15] still occupy a prominent position in the daily

practice of construction management [16]. Indeed, deterministic techniques have some important advantages. They are easier to learn, and their outputs are easier to understand and communicate. The amount of input information they require is generally much lower than other advanced techniques. Additionally, the calculations they involve are also significantly less computer-intensive and do not generally require advanced training to implement them.

However, deterministic scheduling techniques also suffer from serious limitations [16]. In a recent study, Ballesteros-Pérez et al. [17] broke down the core assumptions of some classical scheduling techniques and exemplified why they consistently underestimate the project duration even when accurate input information is available. Their findings were not new, but they emphasized that deterministic techniques' major source of inaccuracy stems from neglecting and/or not properly handling activity duration variability. Duration variability is defined here as the difference between the actual (final) duration and the planned (initial) duration of a project activity.

Hence, the purpose of this paper is to propose some mathematical expressions that approximate the average project duration and the project duration standard deviation when its activities have some variability (virtually all real-life projects). Our estimates are compared against the analogous PERT project duration estimates. It is shown how, at the expense of minimal additional calculation and from the very same input information, our estimates largely outperform the PERT estimates. This allows bridging the accuracy gap that separates deterministic techniques from more complex techniques.

The paper is structured as follows: Section 2 *Background* describes the activity duration variability and the merge event bias, which are the major source and consequence, respectively, of inaccuracy in all deterministic techniques. In Section 3 *Research methods*, the mathematical expressions proposed are presented and all their subcomponents are justified. Section 4 *Application example* describes how all the calculations can be easily implemented step by step. Section 5 *Analysis* and Section 6 *Results* describe how the two expressions were tested with a diverse and representative artificial network dataset. Section 7 *Discussions* comments on how the expressions can be used in real contexts to improve the reliability of project management tasks in general, and the accuracy of some complementary scheduling techniques in particular. Then, Section 8 *Conclusions* summarizes the paper's contributions, highlighting the research limitations and proposing future avenues of research.

## 2. Background

The literature on project duration estimation is vast and cannot be representatively outlined here. Some of the most relevant scheduling techniques were presented in *Introduction*, and more are commented on later in the *Discussions*. This section considers in detail why activity duration variability is the major source of inaccuracy of all deterministic scheduling techniques and what can be done to (partially) overcome that problem.

Activities in all projects, construction projects included, suffer from some degree of duration variability [18,19]. Projects are unique, which means that, even when contractors have extensive experience, subsequent projects may differ regarding clients, contract type, location, regulations, labor, equipment, subcontractors, stakeholders, and weather [20]. All these (and many other) factors constitute potential causes of uncertainty and, eventually, of activity duration variability [21].

Deterministic scheduling techniques assume that activity durations are constant. In an attempt to draw up a representative execution plan, most project schedulers use the *average activity durations*, expecting that such an approach will produce the *most likely* project duration [13]. However, such a schedule is often not reliable.

When two or more activities are performed in parallel, the *average* time it takes to complete all of them is *not* the maximum of the activity averages, but generally more. Indeed, this situation is identical to the problem of calculating the maximum of two or more random variables. Imagine two activities whose duration is modeled by a normal distribution. Half of the time, each activity ends early, and, half of the time, it ends late. However, when performing both activities in parallel, only in one out of four occasions will both activities finish early. In the other three occasions, either one of the activities

will finish late, or both will be late. Hence, the maximum time of completion of these two activities will be higher than the average of the activity durations. This phenomenon is known as the *merge event bias*, and it is the reason why deterministic scheduling techniques generally underestimate the actual project duration.

The merge event bias is nothing but a manifestation of Jensen's inequality [22]. More precisely, the maximum of several random variables (each representing the activity durations) is a convex function. This means that, *on average*, the maximum of several randomly generated activity durations will always be higher than, or at least equal to, the averages of those activity durations. Hence, the problem of obtaining the average project duration (also known as project makespan) is the same as obtaining the maximum of several random variables, and this was intensively studied in the field of stochastic network analysis (SNA).

In SNA, activity durations are modeled with different statistical distributions. One randomly generated duration value is generated for each activity in each simulation run, and then the project duration is computed by following the activities precedence network. This operation is repeated thousands of times until the sample of project durations is deemed representative. Then, the statistical project duration distribution can be represented, and its first two moments (the project duration average and the standard deviation) can be easily calculated.

However, as hinted above, there are *no analytical solutions* to find (accurate) values of the average project duration in project network topologies that involve multiple parallel critical and subcritical paths. The only technique that provides reliable results when activities have variable durations is the Monte Carlo simulation, which is one of the most common in SNA [23–25].

The main reason why there is no analytical solution in a schedule network is because there are no statistical distributions that are, at the same time, sum-stable and max-stable. When activities are in series, their durations are added (convoluted) to calculate the path durations. When activities are in parallel, the maximum of several path durations (distributions) has to be calculated instead. If the activity duration distributions of a schedule were sum-stable and max-stable, a distribution of the same family would result after calculating the project duration irrespective of whether the schedule mixed activities in series and parallel. Also, the parameters from such a distribution could be anticipated from the original activity duration distributions. However, as noted above, there are no sum-stable and max-stable distributions. The normal distribution is sum-stable, but not max-stable. The lognormal is product-stable, but neither sum- nor max-stable. Hence, in the absence of a distribution that can be used across a whole schedule involving activities in series and in parallel, approximate approaches must be used.

Precisely with the intention of allowing project managers to make probabilistic inferences about the likely project duration, the project evaluation and review technique (PERT) was proposed over 60 years ago [7]. By means of three-point activity duration estimates (the well-known optimistic, most likely, and pessimistic durations of an activity), the average project duration is assimilated to the average duration of the (longest) critical path. Then, the project duration standard deviation is also assimilated to the duration standard deviation of that same longest critical path. The problem, however, is that PERT underestimates the average project duration and overestimates the project duration standard deviation [26,27]. Indeed, the PERT average project duration coincides with the one from a deterministic schedule. The project duration standard deviation, on the other hand, assumes that only a single critical path exists.

However, despite the limitations of PERT, it is a hugely popular technique in the field of project management. From the research standpoint, practically all its assumptions and calculation procedures were broken down and analyzed in pursuit of improvements from every possible perspective, for example, the activity duration distributions, the resulting project duration distribution, the accuracy of the three-point estimates, the inclusion of activity time–cost trade-offs, the inclusion of extra scheduling operations like minima of multiple paths, and activity self-loops, to cite just a few (see Ballesteros-Pérez [13] for an extensive review).

Of particular interest, however, may be the work of Pritsker [28]. He developed the graphical evaluation and review technique (GERT) in 1966. GERT attempted to overcome many of PERT's fundamental problems, including the merge event bias. However, due to its mathematical complexity, its implementation was not possible without resorting to Monte Carlo simulation. More recently, some of GERT's principles plus other convenient features were incorporated into M-PERT [13]. This technique was proposed by Ballesteros-Pérez et al. and also allows manual calculation of the project duration average and standard deviation. Specifically, M-PERT recursively merges all project activities until there is one activity left which represents the project duration variability. For medium and big project networks, however, this technique can be come time consuming.

Hence, there is one common trait among almost all these PERT extensions. Those which (partially) overcome the merge event bias problem do so at the expense of (substantially) increasing the technique complexity and/or the time it takes to calculate the project duration average and standard deviation. That is why our aim here is to resort to a number of operations very similar to the original PERT technique, but which significantly improve its accuracy by allowing multiple critical and subcritical paths.

### 3. Materials and Methods

We assume that a project  $p$  corresponds to a network of activities connected by technological and/or resource precedence relationships. Any activity  $i$  in the schedule can have some duration variability which is modeled by a statistical distribution with mean  $\mu_i$  and standard deviation  $\sigma_i$ . This does not preclude, however, the use of other non-location-scale or asymmetrical activity duration distributions later.

As discussed earlier, in stochastic network analysis (SNA), a representative sample of project durations are obtained by Monte Carlo simulation and eventually represented by a cumulative distribution function (CDF). Here, we are not interested in the analysis of the project duration CDF, but rather in proposing some mathematical expressions for estimating its first two moments. The first moment of the project duration distribution is the project duration mean ( $\mu_p$ ). The second moment is the project duration standard deviation ( $\sigma_p$ ).

From now on, we generically refer to the distribution of all possible project durations as *real duration* (RD). This term is intentionally chosen in contrast with the *planned duration* (PD) which is generally associated with the (unique) project duration estimate coming from a deterministic schedule (a Gantt chart, for instance). PD also coincides with the PERT estimate of the project duration average which, as described earlier, is prone to underestimations when activities have duration variability. Hence, we propose the following:

$$RD \sim \text{Project Duration distribution (mean} = \mu_p, \text{st.deviation} = \sigma_p) \rightarrow \text{with } \mu_p \geq PD. \quad (1)$$

Finding approximated values of  $\mu_p$  and  $\sigma_p$  is the aim of this research.

#### 3.1. General Notation

In this subsection, we introduce the most relevant variables, subscripts, and sets.

Let  $i = 1, 2 \dots n_i \in I$  be the set of all *activities* in the project schedule.

Let  $k = 1, 2 \dots n_k \in K$  be the set of all *critical activities* in the project schedule, that is, those whose total slack is zero (noted later as  $s_i = 0$ ).

Let  $j = 1, 2 \dots n_j \in J$  be the set of all *critical and subcritical activities* in the project schedule (those whose slack  $0 \leq s_i < \tau$ , with  $\tau$  being a cut-off slack value named slack admissibility threshold defined later).

From the definitions above, it is clear that  $K$  is always a subset of  $J$ , whereas  $J$  is always a subset of  $I$ , that is,  $K \subset J \subset I$ .

Additionally,  $d_i$ ,  $d_k$ , and  $d_j$  represent the duration of activities  $i$  (all),  $k$  (critical), or  $j$  (critical and subcritical), respectively. In deterministic schedules, they are assumed as static (constant) (generally,  $d_i = \mu_i$ ), even though they may actually have some variability  $\sigma_i$ .

$s_i$ ,  $s_k$ , and  $s_j$  represent the *total slack* (or floats) of an activity  $i$  (all),  $k$  (critical), or  $j$  (critical or subcritical) obtained from a deterministic schedule. The activity *free slacks* is not used. Furthermore, it is worth noting that, although PERT does not directly use the slack values in its calculations, they must be calculated anyway; otherwise, the critical activities cannot be identified.

Now, due to its practical interest, we need to take a closer look at the activity duration standard deviations. Like the activity duration means ( $\mu_i$ ,  $\mu_k$ ,  $\mu_j$ ), the activity duration standard deviations ( $\sigma_i$ ,  $\sigma_k$ ,  $\sigma_j$ ) can be estimated in real contexts from the scheduler's past experience, from the PERT three-point estimates, or even by analogy to similar projects. With the three-point estimates, for example, the values of the first two moments of each activity duration distribution ( $\mu_i$  and  $\sigma_i$ ) can be calculated with the following simple expressions:

$$\mu_i = \frac{O_i + 4L_i + P_i}{6}, \tag{2}$$

$$\sigma_i = \frac{P_i - O_i}{6} \sqrt{\frac{5}{7} + \frac{16}{7} \cdot \frac{(L_i - O_i)(P_i - L_i)}{(P_i - O_i)^2}}, \tag{3}$$

where  $O_i$ ,  $L_i$ , and  $P_i$  represent the optimistic (minimum), most likely (the mode), and pessimistic (maximum) duration of activity  $i$ , respectively, to be elicited by the project scheduler. As can be seen, Equation (3) already includes a correction (the square root term) recently proposed by Herrerías-Velasco et al. [29]. This extra term makes the  $\sigma_i$  estimate more accurate and also unbiased.

Regarding the possibility of estimating  $\sigma_i$  values by analogy, Ballesteros-Pérez et al. [30] established that activity duration variability in construction projects is closely modeled by lognormal distributions with median zero and coefficients of variation ( $CV_i$ ) of 0.10 (for activities with low variability), 0.20 (for medium variability), and 0.3 (for high variability). The coefficient of variation is the ratio of the activity duration standard deviation by the duration mean, that is,  $CV_i = \sigma_i/\mu_i$ . However, Ballesteros-Pérez et al. [30] measured the *log* ratios of actual versus planned activity durations. If we want to calculate any activity  $i$ 's duration in the first two moments in the natural scale ( $\mu'_i$  and  $\sigma'_i$ ), we must perform the following calculations (derived from the first two moments of a lognormal distribution):

$$\mu'_i = \mu_i \cdot e^{\frac{(2.302 \cdot CV_i)^2}{2}} \tag{4}$$

$$\sigma'_i = \sqrt{(e^{(2.302 \cdot CV_i)^2} - 1) e^{(2 \cdot LN(\mu_i) + (2.302 \cdot CV_i)^2)}} \tag{5}$$

where  $CV_i = 0.10, 0.20$ , and  $0.30$  for low, medium, and high activity duration variability, respectively. The coefficient 2.302 is added to change the base from 10 (the one used in Ballesteros-Pérez et al.'s [30] study) to  $e$  (the most common with lognormal distributions). The activity duration mean  $\mu_i$  is directly expressed in (natural) time units and can be calculated from the scheduler's past experience or from Equation (2), for instance.

### 3.2. Expression for Estimating the Average Project Duration $\mu_p$

As anticipated earlier, the calculation of  $\mu_p$  involves screening out some initial activities and keeping only those activities which are either critical or subcritical. The explanations below illustrate the necessary calculations in the same exact order a practitioner should implement them.

We consider as critical or subcritical those activities whose total slack ( $s_i$ ) remains below a slack admissibility threshold ( $\tau$ ), that is,  $i \in J \leftrightarrow 0 \leq s_i < \tau$ . The slack admissibility threshold  $\tau$  is calculated from the critical activities ( $k$ ), that is, those with  $s_i = 0$ , with the following expression:

$$\tau = K_0(1 - SP)\sigma_K, \tag{6}$$



where  $\sigma_K$  is an estimate of the critical path(s) duration standard deviation. It can be calculated with the following straightforward (but approximated) expression:

$$\sigma_K = \sqrt{\frac{PD}{\sum d_k} \cdot \sum \sigma_k^2} \tag{7}$$

When there is a single critical path, the sum of all critical activity durations is equal to the (deterministic) duration of the whole project, that is,  $\sum d_k = PD$ . In this case,  $\sigma_K = \sqrt{\sum \sigma_k^2}$  and the value of  $\sigma_K$  is exact. Only in this case does  $\sigma_K$  coincide with the PERT estimate of the project duration standard deviation.

$SP$  is the serial-parallel topological indicator, which can be calculated as follows:

$$SP = \frac{m - 1}{n_i - 1}, \tag{8}$$

where  $m$  is the number of activities in the path with *more activities* (not necessarily the longest in duration, that is, not necessarily the critical path);  $n_i$  is the total number of (initial) activities in the schedule.

The  $SP$  measures how close a network is to a parallel network ( $SP = 0$  when all activities are in parallel) or a serial network ( $SP = 1$  when all activities are in series). Recent studies pointed out that most real-life project  $SP$  values range from 0.3 to 0.7 [31].

$K_0$  is an empirical coefficient which represents the number of  $\sigma_K$  multiples that define the overall cut-off value of  $\tau$ . Specifically, values of  $K_0 = 0$  would only accept critical activities (no subcritical activities, that is, no activities with  $s_i > 0$ ). Conversely, values of  $K_0 > 3$  (meaning three times  $\sigma_K$ ) would identify as subcritical activities with too big slacks. After some experimentation, it was found that values of  $K_0$  within [1,2] generally lead to higher  $R^2$  values when calculating  $\mu_p$  (Equation (9)) shown later). Consequently, most of the time, we just assume  $K_0 = 1.5$ .

The attentive reader would have noticed that the purpose of Equation (6) is just ruling out those activities with excessive slacks. This is simply because the paths to which those activities belong are more unlikely to extend the project duration and ultimately do not affect  $\mu_p$ . Furthermore, the purpose of  $SP$  is to fine-tune the effectiveness of Equation (6) by considering the network macro-structure.

Therefore, once the value of  $\tau$  is known, it is possible to know which activities are critical and subcritical (the  $J$  set). These are the activities whose total slack  $s_i$  remains below the threshold defined by  $\tau$ . Only with those  $j$  activities can the average project duration distribution ( $\mu_p$ ) then be easily obtained with

$$\mu_p = PD \cdot \left[ 1 + K_1 \sqrt{\frac{N_p}{n_j} \cdot \frac{\sum \sigma_j}{\sum d_j} \cdot LN(S \cdot N_p)} \right] \tag{9}$$

For calculating Equation (9), we need the total number of critical + subcritical activities  $n_j$ , the (deterministic) activity durations  $d_j$ , the activity duration standard deviations  $\sigma_j$ , and the activity total slacks  $s_j$  (to calculate  $S$ ), and the deterministic project duration  $PD$ . This is the same input information that the PERT technique uses. Its components are outlined below.

$K_1$  is another empirical coefficient whose recommended value is 1.12 when the activity duration distributions are symmetrical (for example, normal, uniform). If the activity duration distributions are asymmetrical (for example, lognormal, most triangular distributions), then we suggest taking a slightly higher value ( $K_1 = 1.22$ ). It must be borne in mind, however, that coefficients  $K_0$  and  $K_1$  are somehow in equilibrium with each other. If  $K_0$  in Equation (6) was higher (accept more activities as  $j$ ), then  $K_1$  would have to be smaller (to account for a potentially lower impact of those  $j$  activities on  $\mu_p$ ), and vice versa. After some experimentation, we found the values proposed here for both coefficients seem to work well in almost all cases.

$N_p$  represents an estimate of the equivalent number of parallel critical + subcritical paths (assuming they all spanned from the project start to the project end). We calculate it as follows:

$$N_p = \frac{\sum d_j}{PD}. \tag{10}$$

It is worth noting that, although the numerator and denominator of  $N_p$  could have been integrated and simplified within Equation (9), they are presented separately for a reason;  $N_p$  shows a high correlation on its own with  $\mu_p$ , despite it not being linear. Future studies may want to improve our equations by revising, for example, how  $N_p$  is mathematically included in Equation (9).

$\sum \sigma_j$  corresponds to the sum of all critical + subcritical activity duration standard deviations. We present it separately because, in cases where we were using coefficients of variation ( $CV_j$ ), we could easily replace this term with  $\sum(CV_j \cdot d_j)$ .

$LN(\cdot)$  corresponds to the natural (Euler’s) logarithm.

$S$  is the slack tightness coefficient, whose values always remain in the range  $0 < S \leq 1$ ; it is calculated as

$$S = \frac{\tau - \frac{\sum s_j}{n_j}}{\tau} = 1 - \frac{\sum s_j}{\tau \cdot n_j}. \tag{11}$$

$S$  represents the average (total) slack per activity. When  $S = 1$ , all paths have to be critical (there is not a single activity with a slack  $s_j > 0$ ). When  $S = 0$  (an extreme which is not mathematically possible), all subcritical paths would (theoretically) not be causing an impact on  $\mu_p$  (impossible project duration extension beyond  $PD$ ). Hence, like  $SP$  in Equation (6), the variable  $S$  allows fine-tuning the representativeness of  $N_p$  by considering how loose the subcritical activities are.

With all these terms, Equation (9) for calculating  $\mu_p$  was fully defined and can be easily implemented.

### 3.3. Justification of the Expression for Estimating $\mu_p$

Equation (9) encompasses several terms, each in charge of one task. Let us review them in reverse order.

The maximum of  $n$  independent and identically distributed (iid) normal random variables has no analytical expression, but it is upper bounded by  $\sqrt{2 \cdot LN(n)}$ . This is important as, in our context,  $n \approx N_p$ , the random variables represent the path durations (which converge to normal distributions as there are more activities in series by the Central Limit theorem), and their maximum coincides with the average project duration ( $\mu_p$ ).

This upper bound, however, largely overestimates  $\mu_p$  for small values of  $n$  (for instance, when  $n < 50$ , the deviations are still above 20%). Most real projects have far fewer than 50 critical and subcritical parallel paths. That is why, from the upper bound  $\sqrt{2 \cdot LN(n)}$ , we kept the  $LN(n)$  term but reformulated it as  $K_1 \cdot LN(S \cdot N_p)$ . More precisely, the contribution of  $S$  was discussed in the paper, whereas  $K_1$  was adjusted to maximize the fit for  $N_p < 5$  paths (which approximately corresponds to projects with a serial-parallel indicator  $SP > 0.2$ , that is, most real projects as justified earlier). Consequently, in cases when  $N_p > 5$ , Equation (9) slowly (but progressively) loses accuracy.

Next, the term  $\sum \sigma_j / \sum d_j$  in Equation (9) represents the average activity duration coefficient of variation, that is, the average  $CV_j$ . In particular, this term is obtained from a weighted average of all activity coefficients of variation ( $CV_j$ ) using the deterministic activity durations ( $d_j$ ) as weights.

$$\frac{\frac{\sum(d_j \cdot CV_j)}{n_j}}{\frac{\sum d_j}{n_j}} = \frac{\sum(d_j \cdot CV_j)}{\sum d_j} = \frac{\sum \sigma_j}{\sum d_j}.$$

The ratio above is eventually multiplied by  $PD$  in Equation (9) which converts it into a proper standard deviation (in time units). The *raison d'être* of this term is that, following the previous analogy, the maximum of  $n$  iid normally distributed random variables is always proportional to the duration standard deviation of the parallel paths to which those activities belong. This justifies why we need to include a term that represents the average activity duration standard deviation, or, in this case, the average activity coefficient of variation.

There is just one term left to justify from Equation (9); this is  $\sqrt{N_p/n_j}$ . This term provides an estimate of the average number of activities per critical and subcritical path. When  $n$  iid activities are in series, then  $\sqrt{\sum \sigma_j^2} = \sqrt{n \cdot \sigma_j^2}$ . However, since we are working with the coefficient of variation here instead of variances, it means we need to divide the latter expression by the average activity duration, that is,  $\frac{\sqrt{n \cdot \sigma_j^2}}{n \cdot d_j} = \frac{\sqrt{n \cdot d_j^2 \cdot CV_j^2}}{n \cdot d_j} = \frac{CV_j}{\sqrt{n}}$ . In this expression,  $CV_j$  represents the term  $\sum \sigma_j / \sum d_j$  explained earlier. Hence,  $1/\sqrt{n}$  represents the reduction of  $CV_j$  as more activities are located in the same path. In our case, the average number of critical and subcritical activities per path can be approximated by  $n \approx \frac{n_j}{N_p}$ , then  $\frac{1}{\sqrt{n}} \approx \sqrt{\frac{N_p}{n_j}}$ , exactly as in Equation (9).

### 3.4. Expression for Estimating the Project Duration Standard Deviation $\sigma_p$

Along with  $\mu_p$ , calculating an approximation of  $\sigma_p$  allows a practitioner to make probabilistic inferences about how likely it is that a project will finish by different dates. Our expression for estimating  $\sigma_p$  is

$$\sigma_p = \sqrt{\frac{\sigma_K^2 + \sigma_J^2}{2}} \cdot [1 + K_2 \cdot LN(S \cdot N_p)] \tag{12}$$

Most variables in Equation (12) are already known.  $\sigma_K$  represents an estimate of the project duration standard deviation coming from the critical activities, as in Equation (7). Indeed,  $\sigma_J$  is quite similar to  $\sigma_K$ . The only difference is that  $\sigma_J$  is calculated with all critical and subcritical activity durations, not just the critical ones. Hence,

$$\sigma_J = \sqrt{\frac{PD}{\sum d_j} \cdot \sum \sigma_j^2} \tag{13}$$

$K_2$  is another empirical coefficient whose value is suggested to be  $K_2 = -0.22$  when activity duration variability is predominantly coming from a symmetrical distribution (for example, normal distribution), and  $K_2 = 0.07$  when the activity duration variability is modeled with asymmetrical distributions (for example, lognormal).

### 3.5. Justification of the Expression for Estimating $\sigma_p$

Equation (12) takes an *average* path duration variance from  $\sigma_J$  and  $\sigma_K$ . Then, it applies a correction factor that takes into account the reduction/increase (with  $K_2$  and  $S$ ) as a function of the estimated total number of existing critical + subcritical paths (with  $N_p$ ).

As described earlier, in the PERT technique, the project duration standard deviation corresponds to the square root of the sum of all critical activity duration variances, that is,  $\sigma_K = \sqrt{\sum \sigma_k^2}$ . This is exactly like our  $\sigma_K$  variable from Equation (7), but always assuming the existence of a single critical path ( $\sum d_k = PD$ ). However, as real projects may have more than one critical or subcritical path, this would significantly bias the  $\sigma_p$  estimate. In Equation (12), by averaging the variances of  $\sigma_K$  and  $\sigma_J$ , it is more likely that our estimate of the actual project duration standard deviation will improve because the magnitude of  $\sigma_J$ , in most cases, is lower than  $\sigma_K$ .



Finally, the term  $[1 + K_2 \cdot LN(S \cdot N_p)]$  in Equation (12), constitutes a simplified linear regression of the  $\sigma_p$  reduction, as there are more paths in parallel. In this term,  $N_p$  takes care of counting the number of those paths, and  $S$  takes care of reducing their importance as a function of how large their average slack is.  $K_2$  again just centers the regression expression to maximize accuracy for cases when  $N_p < 5$ . As expected then, the estimate of  $\sigma_p$  keeps losing accuracy as  $N_p$  grows beyond five.

### 3.6. A Final Note on Calculations with Non-Symmetrical Activity Duration Distributions

With the exception of the different values for coefficients  $K_1$  and  $K_2$ , we made no distinction between how the expressions of  $\mu_p$  and  $\sigma_p$  work when activity duration distributions are asymmetrical. In those cases, the scheduler must ensure that the deterministic activity durations ( $d_i$ ) always correspond to their *activity duration average*. Indeed, this is also a requirement for symmetrical distributions as well; however, in those cases, choosing the average durations is the default option (as the median, mode, and mean coincide in all common symmetrical distributions). Hence, in the case of symmetrical activity duration distributions,  $d_i = \mu_i$  always. However, in the case of asymmetrical distributions, then  $d_i = \mu'_i$  (as in Equation (4) for lognormal distributions, for instance). Only then will it be possible to calculate  $s_i$ ,  $\sigma_i$ , and  $PD$  properly. If not done this way, the estimates of  $\mu_p$  and  $\sigma_p$  may not be representative.

## 4. Application Example

In this section, we develop a manual example with a small project schedule. The purpose is to illustrate how  $\mu_p$  and  $\sigma_p$  could be easily calculated by practitioners. All necessary auxiliary calculations and intermediate steps are also included. The results of the  $\mu_p$  and  $\sigma_p$  are eventually compared with the PERT estimates and the (more accurate) estimates obtained by Monte Carlo simulation.

Let us refer to the nine-activity schedule network represented in Figure 1. This project consists of activities with varied duration variabilities (represented by different gray tones).

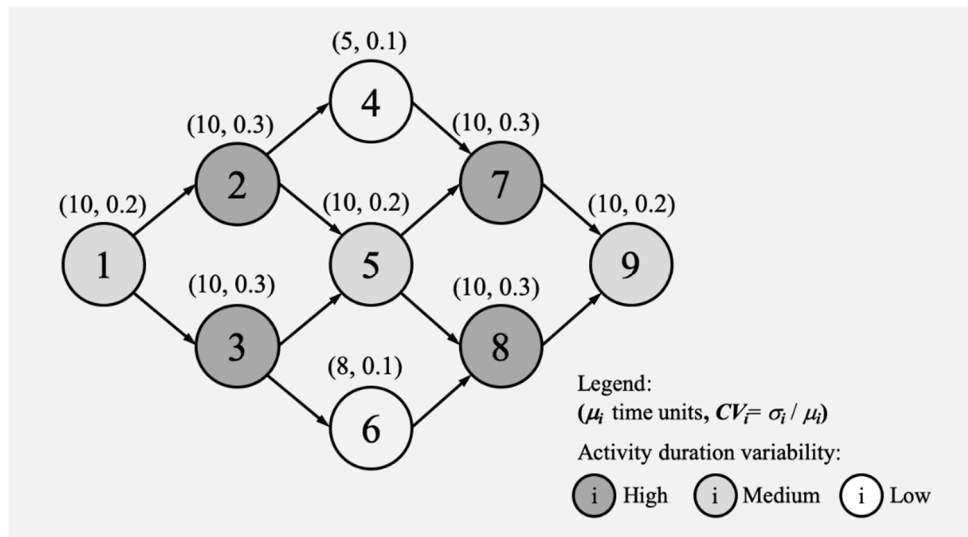


Figure 1. Example project schedule.

Initially, we assume the activity duration variability is normally distributed. This way, the deterministic activity durations are  $d_i = \mu_i$ , and the activity duration standard deviations are  $\sigma_i = \mu_i \cdot CV_i$ . Both sets of values are represented for all activities in the leftmost part of Table 1. From them, calculating the activity slacks is straightforward (indeed, all are equal to zero, apart from the slacks of activities 4 and 6, which are  $s_4 = d_5 - d_4$  and  $s_6 = d_6 - d_5$ , respectively).

**Table 1.** Project network example: deterministic schedule information. Avg.—average; Dur.—duration.

Activity	Normally Distributed Activity Duration Variability				Lognormally Distributed Activity Duration Variability			
	Avg. Dur.	Dur. Variability	(Total) Slack		Avg. Dur.	Dur. Variability	(Total) Slack	
<i>i</i>	<i>d<sub>i</sub></i>	<i>CV<sub>i</sub></i>	<i>σ<sub>i</sub></i>	<i>s<sub>i</sub></i>	<i>d'<sub>i</sub></i>	<i>CV<sub>i</sub></i>	<i>σ'<sub>i</sub></i>	<i>s<sub>i</sub></i>
1	10	0.2	2	0	11.12	0.2	5.40	0
2	10	0.3	3	0	12.69	0.3	9.92	0
3	10	0.3	3	0	12.69	0.3	9.92	0
4	5	0.1	0.5	5	5.13	0.1	1.20	5.98
5	10	0.2	2	0	11.12	0.2	5.40	0
6	8	0.1	0.8	2	8.21	0.1	1.92	2.90
7	10	0.3	3	0	12.69	0.3	9.92	0
8	10	0.3	3	0	12.69	0.3	9.92	0
9	10	0.2	2	0	11.12	0.2	5.40	0

In this case, the deterministic project duration *PD* coincides with the sum of  $d_1 + d_2, d_3 + d_5 + d_7,$  or  $d_8 + d_9 = 50$  days (or time units).

From the initial set of all nine activities (*i*), we know that the critical activities (*k*) are those with  $s_i = 0$ , that is,  $K = \{1, 2, 3, 5, 7, 8, 9\}$  (all *i* other than activities 4 and 6).

The second step involves calculating the approximated critical path duration standard deviation  $\sigma_K$  with Equation (7).

$$\sigma_K = \sqrt{\frac{PD}{\sum d_k} \cdot \sum \sigma_k^2} = \sqrt{\frac{50}{7 \cdot 10} \cdot \sum (3 \cdot 2^2 + 4 \cdot 3^2)} = 5.86 \text{ days.}$$

The longest chain of activities is five activities long; thus,  $m = 5$  and, since  $n_i = 9$  activities (all initial activities), the value of the serial-parallel indicator with Equation (8) is

$$SP = \frac{m - 1}{n_i - 1} = \frac{5 - 1}{9 - 1} = 0.5$$

As  $\sigma_K$  and *SP* are known, we can calculate the slack admissibility threshold  $\tau$  with Equation (6). By assuming  $K_0 = 1.5$ , then

$$\tau = K_0(1 - SP)\sigma_K = 1.5(1 - 0.5)5.86 = 4.39 \text{ days.}$$

The only non-critical activity whose total slack is lower than 4.39 is activity 6 ( $s_6 = 2$  days). Activity 4 with a slack  $s_4 = 5 > \tau = 4.39$  is not considered any further. Hence, the set of critical + subcritical activities is  $J = \{1, 2, 3, 5, 6, 7, 8, 9\}$ , that is, all but activity 4. The cardinality of *J* (the number of elements in the set) is then  $n_j = 8$ .

Before calculating  $\mu_p$ , it is necessary to calculate the value of a few more auxiliary variables.  $n_j = 8$  act.;  $\sum d_j = 7 \times 10 + 8 = 78$  days;  $\sum \sigma_j = 3 \times 2 + 4 \times 3 + 0.8 = 18.8$  days;  $\sum s_j = 7 \times 0 + 2 = 2$  days.  $K_1 = 1.12$ , as we are assuming symmetrical (normal) activity duration variability.

$$N_p = \frac{\sum d_j}{PD} = \frac{78}{50} = 1.56 \text{ paths, with Equation (10).}$$

$$S = 1 - \frac{\sum s_j}{\tau \cdot n_j} = 1 - \frac{2}{4.39 \cdot 8} = 0.94, \text{ with Equation (11).}$$

With all this information, we can now easily calculate  $\mu_p$  with Equation (9).

$$\mu_p = PD \cdot \left[ 1 + K_1 \sqrt{\frac{N_p}{n_j}} \cdot \frac{\sum \sigma_j}{\sum d_j} \cdot LN(S \cdot N_p) \right] = 50 \cdot \left[ 1 + 1.12 \sqrt{\frac{1.56}{8}} \cdot \frac{18.8}{78} \cdot LN(0.94 \cdot 1.56) \right] = 52.3 \text{ days.}$$

A much more accurate estimate of the average project duration with Monte Carlo simulation would be 53.4 days; that is, we miss it by 1.1 days. However, our estimate is substantially closer than the only estimate obtained ( $PD = 50$  days, which is also the PERT estimate), and it was calculated manually.

Now, imagine we want to calculate the project duration standard deviation  $\sigma_p$ . First, we need to calculate  $\sigma_j$  with Equation (13).

$$\sigma_j = \sqrt{\frac{PD}{\sum d_j} \cdot \sum \sigma_j^2} = \sqrt{\frac{50}{78} \cdot \sum (3 \cdot 2^2 + 4 \cdot 3^2 + 0.8^2)} = 5.72 \text{ days.}$$

Then, assuming  $K_2 = -0.22$  (symmetrical activity duration variability), we use Equation (12) to calculate  $\sigma_p$ .

$$\sigma_p = \sqrt{\frac{\sigma_K^2 + \sigma_j^2}{2}} \cdot [1 + K_2 \cdot LN(S \cdot N_p)] = \sqrt{\frac{5.86^2 + 5.72^2}{2}} \cdot [1 - 0.22 \cdot LN(0.94 \cdot 1.56)] = 5.24 \text{ days.}$$

The project duration standard deviation obtained by simulation yields 4.88 days. We observe a deviation of just 0.36 days in this case. The only alternative estimate would have been the PERT one, but it does not offer a solution when there are multiple critical paths. If we assume that there is only a single critical path (as all critical paths intentionally had the same standard deviations in our example), then

$$PERT \sigma_K = \sqrt{\sum_{one \ path} \sigma_k^2} = \sqrt{\sum (3 \cdot 2^2 + 2 \cdot 3^2)} = 5.47 \text{ days,}$$

which has a deviation of 0.59 days, that is, again worse than our  $\sigma_p$  estimate.

Finally, the same calculations could have been replicated but assuming activity duration variability modeled with lognormal distributions. In this case, the values of  $K_1$  and  $K_2$  would change to 1.22 and 0.07, respectively. Additionally, before starting all calculations, the activity  $d_i$  and  $\sigma_i$  values would need to be calculated with Equations (4) and (5). The results of these variables for all activities are displayed in the rightmost part of Table 1 under the  $d'_i$  and  $\sigma'_i$  columns. Then, the deterministic project duration ( $PD = 58.7$  days) and the activity slacks ( $s_i$ ) could have been calculated in the same way as normal activity duration variability. Indeed, the rest of the calculation process is exactly analogous to the previous example. The final results would have been as follows:

$\mu_p = 66.8$  days (simulation estimate = 68.7 days) better than  $PD = 58.7$  days.

$\sigma_p = 18.3$  days (simulation estimate = 18.4 days) better than PERT  $\sigma_K = 18.9$  days.

## 5. Analysis

In the previous section, we developed a manual example with the single purpose of illustrating how the expressions would be implemented by practitioners. In this section, a much wider and representative validation and performance analysis is conducted.

Our expressions depart from deterministic schedule information to estimate the first two moments of the project duration distribution ( $\mu_p$  and  $\sigma_p$ ). As in the manual example, we again employ Monte Carlo simulations to calculate the almost exact values of  $\mu_p$  and  $\sigma_p$ . Then, the simulation estimates are compared with those obtained with our expressions. However, in order to draw valid conclusions, we use a wide benchmarking set of project schedules and model all their activities representatively.

### 5.1. Artificial Project Dataset

Network topology refers to the logical layout of a network (a project schedule). It defines the way different activities (often referred to as nodes) are placed and interconnected with each other. To test the quality of our  $\mu_p$  and  $\sigma_p$  estimates, an artificial network dataset was used consisting of 4100 different networks (project schedules) with varied topologies. This dataset was developed by

the Operations Research and Scheduling Research Group at Ghent University (Belgium). It can be downloaded from <http://www.projectmanagement.ugent.be/research/data> (MT dataset in the “artificial project data” section (MT = measuring time)).

This set contains 4100 activity-on-node networks with 32 activities each. However, two activities out of the 32 are included simply to signpost the project start and end and are considered dummy activities (zero duration). The network dataset was generated with the RanGen2 algorithm. RanGen2 is a robust random network generator capable of generating a wide range of network topologies, which was validated in several studies [32,33]. Also, this very same network dataset was representatively used in other research studies on earned value management (for example, References [31,34–36]) and schedule risk analysis (for example, References [11,12,37]).

In particular, the network dataset was generated under pre-set values of four topological indicators: the serial–parallel (*SP*), the activity distribution (*AD*), the length of arcs (*LA*), and the topological float (*TF*). These four topological indicators were initially proposed by Tavares et al. [38] and later refined by Vanhoucke et al. [33] and Vanhoucke [11]. More precisely, the *SP* indicator, introduced earlier, describes how close a network is to a serial or parallel network. *AD* describes the distribution of activities in the different schedule paths. *LA* measures the distance between two activities in the project network. *TF* measures how dense a network is (the amount of topological slacks) at the topological level. The values of these indicators all range between 0% and 100%, and they are considered representative and accurate descriptors of a network topology. For the interested reader, the values of all these indicators for the 4100 networks can be found in Appendix B of the Supplementary Materials ( $\mu_p$  and  $\sigma_p$  regression results file).

### 5.2. Activity Duration Distribution

There were not many studies analyzing which statistical distribution is the most suitable to describe the distribution of the activity durations in a project schedule. Trietsch et al. [39] suggested that both the activity duration distribution and the activity duration variability seemed to be fairly well approximated by lognormal distributions. Hence, we also predominantly use the lognormal distribution in our validation analysis. However, we also resorted to other distributions (e.g., normal, uniform) to explore whether our expressions worked equally well with them. The second column of Table 2 includes the 12 different distribution combinations used.

**Table 2.** Project dataset configurations of activity durations (one per set) and activity duration variabilities (two per set). Act.—activity; Proj.—project.

Dataset	Act. Dur. ( $d_i$ ) Distribution	N° Proj.	N° act./ Proj. ( $u_i$ )	Activity Duration Variability 1	Activity Duration Variability 2
I	Lognormal ( $\mu = 2, \sigma = 1$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
II	Lognormal ( $\mu = 2, \sigma = 1$ )	4100	30	Normal with $CV_i = 0.1$ (constant)	Lognormal with $CV_i = 0.1$ (constant)
III	Lognormal ( $\mu = 2, \sigma = 1$ )	4100	30	Normal with $CV_i = 0.3$ (constant)	Lognormal with $CV_i = 0.3$ (constant)
IV	Lognormal ( $\mu = 2, \sigma = 1$ )	4100	15	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
V	Lognormal ( $\mu = 0.25, \sigma = 0.005$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
VI	Lognormal ( $\mu = 0.25, \sigma = 0.75$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
VII	Lognormal ( $\mu = 6, \sigma = 0.12$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
VIII	Lognormal ( $\mu = 6, \sigma = 1.5$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
IX	Normal ( $\mu = 25, \sigma = 7.5$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
X	Normal ( $\mu = 25, \sigma = 7.5$ )	4100	30	Normal with $CV_i = 0.1$ (constant)	Lognormal with $CV_i = 0.1$ (constant)
XI	Normal ( $\mu = 25, \sigma = 7.5$ )	4100	30	Normal with $CV_i = 0.3$ (constant)	Lognormal with $CV_i = 0.3$ (constant)
XII	Uniform ( $l = 0, u = 100$ )	4100	30	Uniform with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$
XIII	Constant ( $d_i = 10$ )	4100	30	Normal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$	Lognormal with $CV_i = \text{Uniform} (l = 0.1, u = 0.3)$

To choose the parameter values of each dataset combination presented in Table 2, a preliminary study was conducted of 101 real construction projects comprising over 6000 activities. The intention was to choose some distribution parameter values that emulated those of real projects. The 101 project schedules were retrieved from a real project dataset developed by Batselier and Vanhoucke [40] and Vanhoucke et al. [41]. These schedules contained both planned and actual (as-built) durations. From both sets of durations and from each project, activity *log* duration values were taken and then their average and standard deviation calculated. Later, these two moments were used to generate activity duration distributions that seemed as realistic as possible in our simulations. In particular,

they were used to generate the lognormally distributed (deterministic) activity duration values of the 30 non-dummy activities in the 4100 schedules. For the interested reader, the values of both moments ( $\mu$  and  $\sigma$ ) of the activity *log* durations for the 101 real projects can be found in Appendix A of the Supplementary Materials (Table S1). For further details about the real projects, the reader is referred to the individual project cards at OR-AS.be [42].

### 5.3. Activity Duration Variability

Having set the (initial) activity durations, the activity duration variability was modeled with normal distributions and lognormal distributions predominantly (although a few simulations were also run using uniform distributions, see dataset XII in Table 2). Normal distributions were used because they are the most common in academic settings and theoretical studies. Lognormal distributions were also used because, as explained earlier, this distribution seems to be the closest for modeling real construction project activities. Furthermore, building on the recent findings of Ballesteros-Pérez et al. [30] for construction projects, we assumed that the median of our distributions would equal the initial (deterministic) activity durations defined in the previous subsection. This would ensure that half of the time the activities would end soon and half of the time late. This would avoid biasing the results toward predominantly early or late projects.

Next, activity duration variability was reproduced by randomly allocating a coefficient of variation between 0.10 (low variability) and 0.3 (high variability) to each activity as measured by Ballesteros-Pérez et al. [30] (in  $\log_{10}$  scale). The median in the normal distribution is equal to the mean, which indicates that the stochastic activity duration values were generated with the following expression:

$$d_i \sim \mu_i \cdot \text{Normal}^{-1}(\text{mean} = 1, \text{st.dev.} = CV_i). \quad (14)$$

In the expression above,  $\mu_i$  represents the default (deterministic) duration of each activity (defined in the previous subsection).  $\text{Normal}^{-1}(\cdot)$  represents the quantile function of the normal distribution, that is, the one in charge of generating the normally distributed stochastic values.  $CV_i$  is the coefficient of variation randomly chosen for each activity in the range [0.1, 0.3]. It is worth noting that this range of  $CV_i$  values in natural scale produces significantly less variability than in log scale (see Equation (15)). However, values of  $CV_i > 0.3$  are very likely to generate negative activity durations, which makes no sense and should be avoided.

In the case of lognormally distributed activity duration variability, the median does not coincide with the distribution mean unless it is transformed into a log scale first. That is why the following expression was adopted to generate thousands of (stochastic) activity duration values for each activity in the Monte Carlo simulations:

$$d_i \sim \mu_i \cdot e^{\text{Normal}^{-1}(\text{mean}=0, \text{st.dev.}=2.302 \cdot CV_i)}. \quad (15)$$

As before, the coefficient 2.302 is included in Equation (15) to change the base from 10 to  $e$ . Variables  $\mu_i$ ,  $\text{Normal}^{-1}(\cdot)$ , and  $CV_i$  are analogous to those in Equation (14).

### 5.4. Parameter Configuration Summary

In the proposed simulation framework, all activities were scheduled to start as soon as possible, and activity preemption was not allowed to avoid losing (simulation) control over the impact of activity duration variability on  $\mu_p$  or  $\sigma_p$ . All activity precedence relationships were assumed to be finish–start. The latter choice, however, is not relevant, as the other three types of precedence (start–start, start–finish, and finish–finish) can be easily reformulated as a finish–start relationship (or the other way around) in any schedule (see Lu and Lam [43] for a comprehensive treatment of precedence relationship transformations).

Hence, 13 different parametrizations of the same 4100-network dataset were finally used, with different activity duration distributions, and two activity duration variabilities per set. They are all summarized in Table 2.

To summarize, the different simulation combinations differed in one or more of the following aspects:

- By the distribution used to generate the initial (deterministic) activity duration values. Most of the time lognormal distributions were used (datasets I–VIII); sometimes, normal distributions (datasets IX–XI) were used, as well as a uniform (dataset XII), and even a constant distribution (dataset XIII, which assumed all activities lasted 10 days on average). Of particular interest are datasets V to VIII which approximately represent the maximum and minimum values found in the first two (log) moments of the 101 real projects.
- By the number of activities in the schedule. In all cases other than dataset IV, the 30 non-dummy activities per network were used. However, a case with fewer activities was also considered to be sure that the network size (number of activities) did not affect the performance of our expressions.
- By distribution (normal, lognormal, and uniform) and level of activity duration variability ( $CV_i$ ). These are represented in the central and rightmost blocks of Table 2. Some assumed  $CV_i = 0.1$  always and some assumed  $CV_i = 0.3$  always, whereas most assumed uniformly distributed  $CV_i$  values between a lower bound ( $l = 0.1$ ) and an upper bound ( $u = 0.3$ ).

Overall, the 13 dataset parametrizations seem to be representative of any possible real project. If our expressions for estimating  $\mu_p$  and  $\sigma_p$  work well with these 13 datasets, then they must also work reasonably well in other contexts. For each of the 4100 projects in the 13 datasets, 10,000 simulations were run assuming normal activity duration variability and another 10,000 assuming lognormal duration variability. Results are discussed next.

### 6. Results

The detailed project results for all 13 datasets can be found in Appendix B of the Supplementary Materials. Here, we present some summary results in Tables 3 and 4.

**Table 3.** Project duration estimates ( $\mu_p/PD$ ) performance results for the 13 project datasets. MAE—mean absolute error; MSE—mean squared error; red.—reduction.

Dataset	Activity Duration Variability 1					Activity Duration Variability 2				
	R	R <sup>2</sup>	K <sub>1</sub>	MAE × 10 <sup>-3</sup> (red.)	MSE × 10 <sup>-3</sup> (red.)	R	R <sup>2</sup>	K <sub>1</sub>	MAE × 10 <sup>-3</sup> (red.)	MSE × 10 <sup>-3</sup> (red.)
I	0.91	0.83	1.12	4.4 (56%)	0.1 (88%)	0.92	0.85	1.22	19.1 (64%)	0.9 (85%)
II	0.90	0.81	1.12	1.3 (50%)	0.0 (87%)	0.92	0.85	1.22	5.0 (61%)	0.1 (82%)
III	0.92	0.84	1.12	8.0 (59%)	0.2 (94%)	0.96	0.92	1.22	27.1 (73%)	1.5 (86%)
IV	0.84	0.71	1.12	5.5 (42%)	0.1 (75%)	0.86	0.74	1.22	26.6 (46%)	1.8 (72%)
V	0.96	0.93	1.12	8.1 (71%)	0.2 (95%)	0.96	0.93	1.22	20.7 (76%)	1.3 (93%)
VI	0.91	0.83	1.12	4.1 (59%)	0.1 (92%)	0.95	0.90	1.22	16.4 (72%)	0.7 (85%)
VII	0.97	0.94	1.12	3.9 (77%)	0.1 (95%)	0.97	0.94	1.22	17.9 (78%)	0.9 (95%)
VIII	0.86	0.74	1.12	4.8 (44%)	0.1 (77%)	0.86	0.74	1.22	21.7 (50%)	1.3 (75%)
IX	0.97	0.93	1.12	3.4 (74%)	0.0 (95%)	0.97	0.94	1.22	16.1 (78%)	0.7 (94%)
X	0.97	0.93	1.12	1.1 (67%)	0.0 (92%)	0.97	0.95	1.22	4.4 (74%)	0.1 (90%)
XI	0.96	0.93	1.12	6.4 (75%)	0.2 (95%)	0.97	0.95	1.22	29.7 (78%)	2.4 (92%)
XII	0.95	0.90	1.12	4.0 (70%)	0.1 (94%)	0.96	0.93	1.22	18.0 (75%)	0.8 (92%)
XIII	0.96	0.93	1.12	8.6 (71%)	0.3 (95%)	0.96	0.93	1.22	20.9 (76%)	1.3 (92%)
Avg.	0.93	0.87	1.12	4.9 (63%)	0.1 (90%)	0.94	0.89	1.22	18.3 (69%)	1.1 (87%)



**Table 4.** Project duration standard deviation estimates ( $\sigma_p/\sigma_K$ ) performance results for the 13 project datasets.

Dataset	Activity Duration Variability 1					Activity Duration Variability 2				
	R	R <sup>2</sup>	K <sub>2</sub>	MAE × 10 <sup>-3</sup> (red.)	MSE × 10 <sup>-3</sup> (red.)	R	R <sup>2</sup>	K <sub>2</sub>	MAE × 10 <sup>-3</sup> (red.)	MSE × 10 <sup>-3</sup> (red.)
I	0.77	0.59	-0.22	28.2 (26%)	2.2 (64%)	0.76	0.58	0.07	43.6 (19%)	8.6 (49%)
II	0.62	0.39	-0.22	18.4 (8%)	1.2 (44%)	0.58	0.33	0.07	27.7 (-5%)	2.2 (28%)
III	0.73	0.53	-0.22	33.1 (36%)	2.5 (66%)	0.82	0.67	0.07	46.4 (1%)	5.8 (49%)
IV	0.70	0.48	-0.22	31.2 (12%)	2.5 (49%)	0.75	0.56	0.07	44.4 (16%)	9.7 (49%)
V	0.92	0.84	-0.22	22.6 (54%)	1.7 (80%)	0.83	0.70	0.07	21.0 (25%)	1.3 (62%)
VI	0.77	0.60	-0.22	26.6 (33%)	1.9 (66%)	0.76	0.58	0.07	40.1 (26%)	8.5 (50%)
VII	0.92	0.85	-0.22	18.5 (65%)	1.0 (89%)	0.86	0.74	0.07	20.2 (36%)	1.3 (70%)
VIII	0.57	0.33	-0.22	29.9 (9%)	3.3 (37%)	0.73	0.54	0.07	45.6 (12%)	11.1 (45%)
IX	0.88	0.78	-0.22	20.5 (57%)	1.2 (82%)	0.89	0.79	0.07	23.7 (39%)	2.4 (70%)
X	0.91	0.82	-0.22	11.1 (58%)	0.4 (85%)	-0.48	0.23	0.07	37.8 (-19%)	4.2 (-49%)
XI	0.94	0.88	-0.22	21.4 (67%)	1.3 (89%)	0.90	0.82	0.07	17.4 (49%)	0.8 (76%)
XII	0.82	0.67	-0.22	25.9 (47%)	2.0 (73%)	0.82	0.67	0.07	32.6 (29%)	4.5 (51%)
XIII	0.92	0.85	-0.22	23.7 (54%)	1.7 (80%)	0.58	0.34	0.07	27.6 (1%)	2.3 (28%)
Avg.	0.81	0.66	-0.22	23.5 (40%)	1.8 (70%)	0.68	0.58	0.07	32.5 (18%)	4.8 (44%)

Tables 3 and 4 compare the actual values (obtained by simulation) with the estimated values (with our expressions) of  $\mu_p$  and  $\sigma_p$  values in the form of ratios, that is,  $\mu_p/PD$  and  $\sigma_p/\sigma_K$ . There are two reasons for this. Firstly, in this way, projects of different durations can be compared under the same scale. Secondly, we can also compare how much better our estimates perform compared to the PERT estimates. As described earlier,  $PD$  is the deterministic project duration but also the PERT  $\mu_p$  estimate. The PERT project duration standard deviation estimate, on the other hand, cannot be produced with PERT when there is more than one critical path. That is why we assumed that the best possible PERT-like estimate would be  $\sigma_K$ , although we ourselves proposed it for the first time in Equation (7).

Let us review all the information displayed in Tables 3 and 4. The first column gives an indication of the dataset analyzed (by row). Each dataset contained 4100 projects and they were all analyzed under two different activity duration variabilities (central and rightmost blocks of Tables 3 and 4). The first block (activity duration variability 1) mostly coincides with normally distributed duration variability except for dataset XII, which was uniformly distributed. The rightmost block (activity duration variability) always coincides with lognormally distributed activity duration variability. The bottom row in both tables displays the column averages for easier data interpretation.

Inside both the central and rightmost blocks, the same information is found. Pearson’s coefficient of correlation ( $R$ ) for  $\mu_p$  is in Table 3 and that for  $\sigma_p$  is in Table 4, which shows the degree of linear correlation between the actual and estimated 4100  $\mu_p/PD$  and  $\sigma_p/\sigma_K$  values, respectively, in each dataset. The coefficient of determination ( $R^2$ ) quantifies the proportion of the (actual vs. estimated  $\mu_p/PD$  and  $\sigma_p/\sigma_K$ ) variability described by our empirical estimates.  $R^2$  is, of course, the squared version of  $R$ ; however, as it offers complementary information, it was deemed useful to include both. Finally, the last two columns in the central and rightmost blocks display the mean absolute error ( $MAE$ ) and the mean squared error ( $MSE$ ), both in  $10^{-3}$  scale, calculated as follows:

$$MAE = \frac{1}{4100} \sum_{p=1}^{4100} |Y_p - \hat{Y}_p|, \tag{16}$$

$$MSE = \frac{1}{4100} \sum_{p=1}^{4100} (Y_p - \hat{Y}_p)^2, \tag{17}$$

where Equations (16) and (17) were calculated for the 13 datasets.  $Y_p$  corresponds to the actual value of  $\mu_p$  (in Table 3) or  $\sigma_p$  (in Table 4) for each project  $p$  (all of them obtained by simulation).  $\hat{Y}_p$  corresponds

to our estimated values of  $\mu_p$  (in Table 3) or  $\sigma_p$  (in Table 4). In addition, all MAE and MSE values in Tables 3 and 4 are also accompanied by an indication of the reduction achieved in MAE and MSE in percentage terms when using our  $\mu_p$  (in Table 3) or  $\sigma_p$  (in Table 4) estimates, compared to using  $PD$  or  $\sigma_K$ , respectively. These values are included within parentheses.

Multiple readings can be made from Tables 3 and 4. For the sake of clarity, we limit ourselves to the most essential. A quick inspection of the first columns of Table 3 shows that values of  $R > 0.90$  are achieved for both duration variabilities.  $R^2$  values are also very close to 90%. Overall, this means that there is a high degree of correlation between our estimates and the actual values. Also, in Table 3, inspection of the MAE confirms that our expressions manage to reduce the variability from the PERT estimates by around two-thirds (63% and 69%). However, the reduction in the MSE is even higher (90% and 87%). This means that, when using our estimates, values can be expected to be around two-thirds closer to the actual values compared to the deterministic project duration estimates, but also that the amount of large deviations (measured by squared errors) would be significantly reduced.

Results in Table 4 are still good, but clearly not as good as those in Table 3. In this table, our project duration standard deviation estimates are not indisputably superior when compared to  $\sigma_K$  (which, it must be borne in mind, are already better estimates than PERT's). Still, in the case of normal variability (central block), we can observe  $R > 0.80$  and  $R^2$  values again around two-thirds. The absolute errors are reduced by 40% on average and the squared errors by around 70%. As  $\sigma_K$  is already better than the PERT estimate, we can expect our estimates to double (at the very least) PERT's accuracy in absolute terms concerning its project duration standard deviation estimate. In squared terms, we could expect our estimates to improve PERT's estimate by around 75% at least.

Results on the lognormal variability (rightmost block in Table 4) also improve the  $\sigma_K$  estimates, but not as much as hoped. Indeed, there are a couple of datasets (II and X) which worsen the  $\sigma_K$  estimates (negative reduction values) in the MAE, MSE, or both. These datasets coincide with the configurations with lower activity duration variability ( $CV_i = 0.10$ ). However, configurations with higher variability (datasets III and XIII, with  $CV_i = 0.3$ ) do not report much better results compared to the case with average variability. Overall, this means that the expression for estimating  $\sigma_p$ , although promising because of its decent correlation values, still needs some work. This is something we already disclosed when introducing it; however, this will probably come at the expense of complicating it numerically. This is an option for future research.

Tables 3 and 4 also contain two columns each reporting the values of coefficients  $K_1$  and  $K_2$ . During the design stage, we experimented by letting these coefficients vary freely in order to minimize the MSE in every dataset. Results of these alternative scenarios can be found in Appendix D of the Supplementary Materials (Tables S3 and S4). It can be seen, however, that the reduction from making coefficients  $K_1$  and  $K_2$  constant did not have a substantial impact on the expressions' performance.

Finally, for those visual readers, Figures 2 and 3 represent the regression graphs of the first dataset (dataset I). Dataset I can be considered the most representative of all datasets, as it approximately presents the average parameters and distribution configurations.

In Figures 2 and 3, the top graphs represent the regression results in ratio scale. On observing the  $\mu_p$  estimates (in Figure 2), it is evident that they are superior to the  $\sigma_p$  estimates (in Figure 3) as the datapoint clouds of the former follow a clearer bisector pattern. The other four graphs (the ones at the bottom of each figure), represent the correlations of estimated and actual  $\mu_p$  and  $\sigma_p$  values in the natural scale instead of ratios (not homogenized by project duration). Arguably, the effect of the project size (duration) plays an important role as it gives the impression of obtaining higher correlations than in ratio scale. Still, these graphs show that the empirical expressions proposed in this paper work quite satisfactorily.

A final comment concerns the last two conditions of a representative regression analysis: independence of errors (normally distributed residuals) and constant variance of errors across observations (homoscedasticity). Regarding the first condition, we must be aware that our expressions are tested in a benchmark network dataset. The projects in this dataset do not follow a perfectly regular

(random) pattern. For example, many of them (approximately one-third, but dependent on each dataset parameter configuration) did not produce a project duration extension at all ( $\mu_p = PD$ ). This means that, in those cases, there was a clearly dominant critical path, and our expressions anticipated it perfectly. However, although this aspect was intrinsically good (demonstrating that our expressions work as they are supposed to), it also artificially increased the residuals kurtosis (a higher than usual density of  $\mu_p/PD = 1$  and  $\sigma_p/\sigma_K = 1$  values). This artificially produced a slight departure from normality in the quantile–quantile (QQ) plots of some datasets (a sample of which is included in Appendix C of the Supplementary Materials for dataset I). Still, all things considered, the QQ plots showed that the residual normality assumption was perfectly tenable.

Regarding the homoscedasticity check, the standardized estimates versus standardized residual graphs were also drawn for all datasets (Appendix C in the Supplementary Materials shows some for dataset I). The gunshot pattern found in all of them evidenced an absence of datapoint directionality which eventually confirms that there was hardly any heteroscedasticity. Overall, all regression conditions were met, indicating that our estimates are unbiased and consistent.

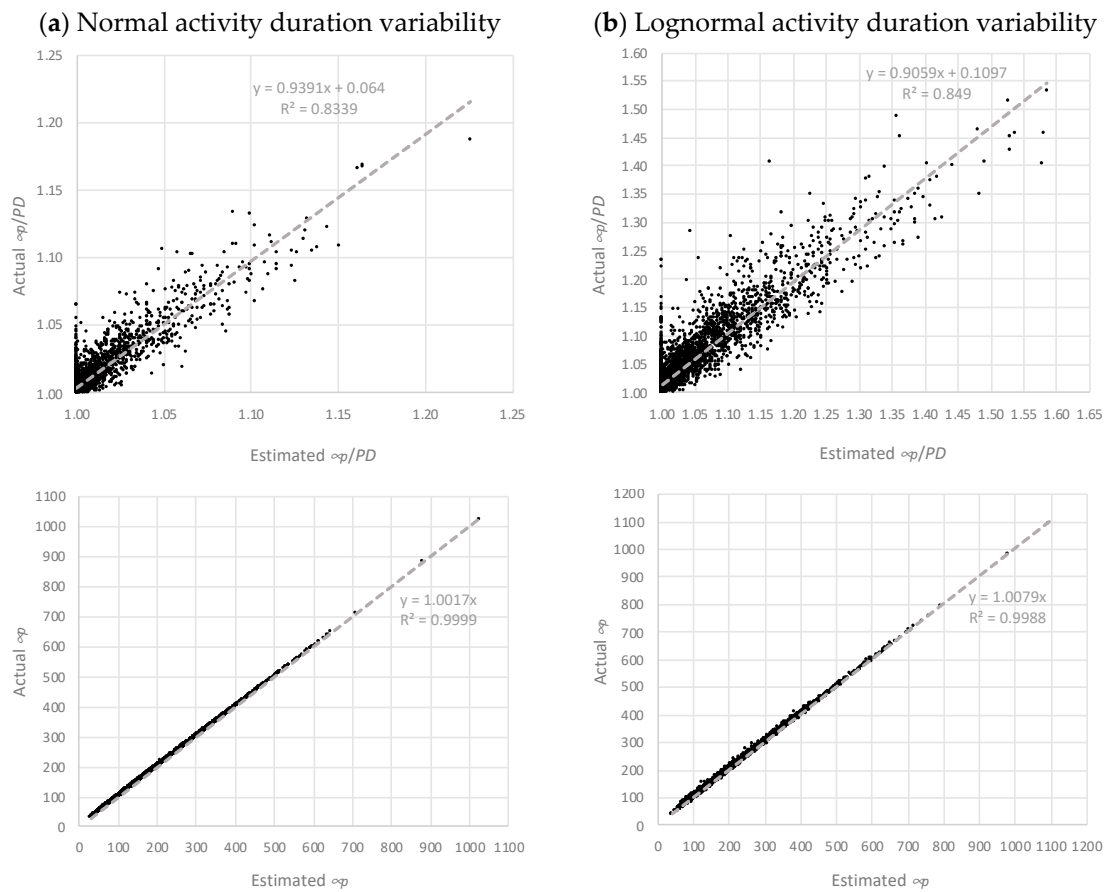


Figure 2. Dataset I  $\mu_p$  regressions with normal (a) and lognormal (b) activity duration variability.

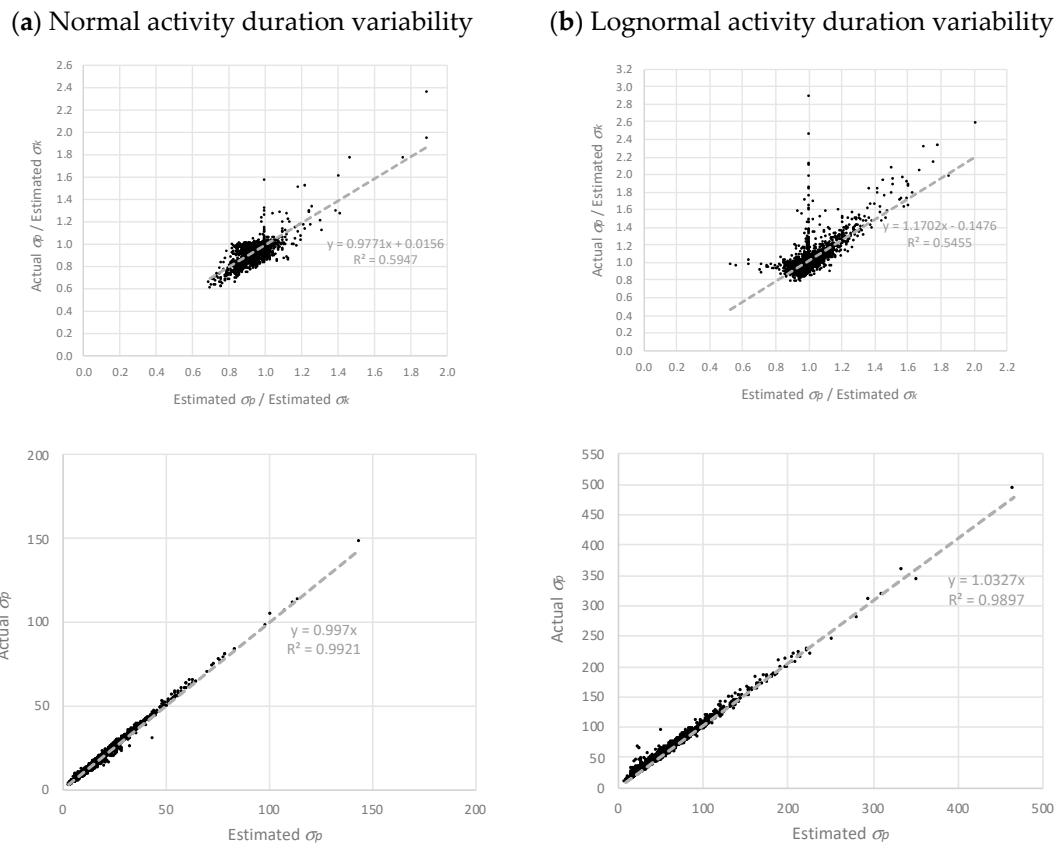


Figure 3. Dataset I  $\sigma_p$  regressions with normal (a) and lognormal (b) activity duration variability.

### 7. Discussion

In this paper, we proposed several expressions that allow calculating the overall project duration average ( $\mu_p$ ) and standard deviation ( $\sigma_p$ ) from purely deterministic schedule information. Calculations are quite straightforward and were exemplified step by step by means of a small project network in Section 4. These calculations can be implemented either manually or in a simple spreadsheet. The only requirement is that the scheduler is familiar with the critical path algorithm. Basically, identifying the critical activities of a project involves the calculation of the activities' earliest and latest start dates. The difference between earliest and latest start dates is equal to the activities' total slacks ( $s_i$ ). Eventually, by comparing the size of  $s_i$  against the slack admissibility threshold  $\tau$  from Equation (6), we can discriminate which activities are critical or subcritical. Hence, any basic spreadsheet that sequences activities departing from the activity durations and precedencies can implement the formulae proposed here. The only requirement is to follow the calculations in the same exact order used in the small project example. For those project schedulers who are not familiar with how to calculate the critical of bigger projects, it is always possible to resort to a commercial scheduling software (e.g., Microsoft Project, Oracle Primavera) to extract the values of the activity slacks ( $s_i$ ). Then, the rest of the calculations can be implemented in a spreadsheet, even manually.

However, beyond obtaining a more accurate estimate of the project duration average, there are other potentially interesting practical applications of the estimates proposed in this paper. Obviously, the first is the possibility of calculating a more accurate average project duration estimate ( $\mu_p$ ) from basic deterministic schedule information with hardly any extra effort. This improved estimate will allow project managers, for instance, to plan execution and resource allocation better with lower margins of error. Also, we anticipated that calculating both  $\mu_p$  and  $\sigma_p$  values could allow us to make probabilistic inferences about any likely project duration ( $X$ ). More precisely, we could say that  $z = (X - \mu_p) / \sigma_p$ .

Then, assuming  $X$  follows a normal distribution, the value of  $z$  could be looked up in a standard normal distribution table and an (approximated) project completion date inferred.

There are other project management tools and techniques that can also benefit from more accurate  $\mu_p$  and  $\sigma_p$  values. Earned value management (EVM), for example, is a deterministic project monitoring and control technique that also produces project duration forecasts, named  $EAC(t)$  (see Ballesteros-Pérez et al. [31] for a recent and comprehensive comparison of the most relevant  $EAC(t)$  expressions). All possible parametrizations of  $EAC(t)$  include a deterministic estimate of the planned duration  $PD$  which, after this research, could be replaced by  $\mu_p$ , enhancing their overall accuracy. Similarly, schedule risk analysis (SRA) is a proactive planning and monitoring technique which ranks project activities and determines their relative importance and exerts different levels of project control on them. SRA uses Monte Carlo simulations to calculate its metrics because, so far, there is no other way of calculating their parameters from deterministic schedule information. For example, the significance index (SI) metric uses  $E(PD)$  (the expectation of the project duration) which is our  $\mu_p$  here. Also, the schedule sensitivity index (SSI) and the criticality slack sensitivity index (CSS) both use  $\sigma_p$ . There are still more variables to calculate (for example, the probability of an activity being critical), but the steps taken in this piece of research promise to avoid the use of computer-intensive simulations when calculating these metrics in the future.

On the other hand, a first apparent limitation of our performance measurement approach is that we predominantly tested our expressions with activity durations modeled by normal and lognormal distributions, and just testimonially with other distributions. The advanced reader may wonder whether the performance results from Tables 3 and 4 might have differed significantly if other distributions were used. In this vein, Hajdu and Bokor [44] concluded that the maximum project duration deviation when using alternative activity distributions was generally well below 10%. This finding resonated with observations from other earlier studies on PERT. For example, MacCrimmon and Ryavec [45] showed in 1964 that, if triangular distributions for modeling activity durations were chosen instead of beta distributions, the probabilistic project duration would have produced almost identical results. All these results are also aligned with our  $K_1$  coefficient values in Equation (9) for calculating  $\mu_p$ . It was observed that the regression coefficient values ( $K_1$ ) that minimize the MSE were 1.12 and 1.22 for symmetrical and asymmetrical distributions, respectively. Hence, irrespective of the specific distribution modeling the activity duration variability, what matters is the order of magnitude of each activity's first two moments ( $\mu_i$  and  $\sigma_i$ ). That is why they constitute the most relevant input information in our expressions.

Another practical limitation of our expressions would be how they can be used in real projects when no activity duration variability estimates ( $\sigma_i$ ) are available. In this regard, Ballesteros-Pérez et al. [30] measured that the average activity duration variability in construction projects remains generally close to  $CV_i = 0.20$ . Hence, if only average activity duration values ( $\mu_i$ ) are known, a project scheduler could assume that  $\sigma_i = CV_i \cdot \mu_i = 0.20 \cdot \mu_i$  for all its activities in the case of symmetrical (for example, normal) duration variability. If they wanted to simulate more realistic (for example lognormal) asymmetrical activity variability, then they would need to resort to Equations (4) and (5) by replacing  $CV_i$  with 0.20 and proceeding as usual.

## 8. Conclusions

Deterministic scheduling techniques, such as the Gantt chart and the critical path method, are still the most common in practical settings these days. However, these techniques neglect activity duration variability which exacerbates a phenomenon known as the *merge event bias*. This bias is shared with the PERT duration estimates and is generally translated into average project durations being underestimated and standard deviations being overestimated. More recently, many advanced non-deterministic scheduling techniques were proposed. However, they are significantly more complex and highly computer-intensive, they require extensive precalibration information, and/or practitioners are not adequately trained to implement them. As a result, their adoption in practice is quite limited.

In this paper, we proposed a series of empirical expressions that allow construction project schedulers to come up with better estimates of the average project duration ( $\mu_p$ ) and the project duration standard deviation ( $\sigma_p$ ). These estimates clearly outperform the analogous classical PERT estimates with the need for neither extra (deterministic) scheduling information nor a substantial additional calculation effort. For testing the superiority of our estimates, a wide and representative benchmark dataset containing 4100 networks (schedules) was employed. The activity durations and their variabilities were modeled with different statistical distributions and different parametrizations.

In this study, we commented on some apparent limitations of our estimates and how to overcome them. Finally, we also extensively discussed how the  $\mu_p$  and  $\sigma_p$  estimates can be used in real construction contexts and/or in combination with other techniques (such as earned value management, and schedule risk analysis) to enhance accuracy or reduce calculation effort.

Future avenues of research are plentiful. For example, the current estimates work well with fewer than five critical and subcritical parallel paths ( $N_p < 5$ ). Future expressions may want to use higher-degree polynomials of the variable  $LN(N_p)$  to improve the accuracy of our  $\mu_p$  and  $\sigma_p$  estimates. Similarly, the accuracy of the  $\sigma_p$  estimate, when lognormal activity duration variability is assumed, could clearly be improved. Future research may want to consider extending the array of estimates that can be (deterministically) calculated. A fine example would be approximating the probability of an activity being critical (the criticality index) with deterministic calculations. This would greatly benefit the field of schedule risk analysis by allowing many metrics to be calculated with computer simulation.

**Supplementary Materials:** The following are available online at <http://www.mdpi.com/2076-3417/10/2/654/s1>: Supplementary A: Real projects dataset characteristics summary and project activity durations log means and log standard deviations; Supplementary B: 4100-project schedule networks and datasets regression results; Supplementary C: Dataset I  $\mu_p$  and  $\sigma_p$  regression QQ plots and homoscedasticity graphs; Supplementary D: Summary of all datasets  $\mu_p$  and  $\sigma_p$  regression results with free  $K_1$  and  $K_2$ .

**Author Contributions:** Conceptualization, P.B.-P.; methodology, P.B.-P. and A.C.-N.; validation, P.B.-P. and M.V.; formal analysis, P.B.-P.; investigation, P.B.-P., M.O.-M. and J.Z.; resources, M.V.; data curation, M.V.; writing—original draft preparation, P.B.-P., A.C.-N. and M.V.; writing—review and editing, P.B.-P., A.C.-N. and M.V.; supervision, M.O.-M. and A.P.-F.; funding acquisition, P.B.-P., A.C.-N. and A.P.-F. All authors have read and agreed to the published version of the manuscript.

**Funding:** The first author acknowledges the Spanish Ministry of Science, Innovation, and Universities for his Ramon y Cajal contract (RYC-2017-22222) co-funded by the European Social Fund. The first two authors also acknowledge the help received by the research project PIN-0053-2019 funded by the Fundación Pública Andaluza Progreso y Salud (Junta de Andalucía, Spain). The first four authors also acknowledge the help received by the research group TEP-955 from the PAIDI (Junta de Andalucía, Spain). Finally, the fifth author, acknowledges the support from the National Natural Science Foundation of China (No. 71301013), the National Social Science Fund Post-financing projects (No.19FJYB017), the List of Key Science and Technology Projects in China's Transportation Industry in 2018-International Science and Technology Cooperation Project (No.2018-GH-006), and the Humanity and Social Science Program Foundation of the Ministry of Education of China (No. 17YJA790091).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Ansar, A.; Flyvbjerg, B.; Budzier, A.; Lunn, D. Does infrastructure investment lead to economic growth or economic fragility? Evidence from China. *Oxf. Rev. Econ. Policy* **2016**, *32*, 360–390. [[CrossRef](#)]
2. Flyvbjerg, B. *Over Budget, over Time, over and over Again*; Oxford University Press: Oxford, UK, 2011.
3. Hamzah, N.; Khoiry, M.A.; Arshad, I.; Tawil, N.M.; Ani, A.C. Cause of Construction Delay—Theoretical Framework. *Procedia Eng.* **2011**, *20*, 490–495. [[CrossRef](#)]
4. Keane, P.J.; Caletka, A.F. *Delay Analysis in Construction Contracts*; Wiley-Blackwell: Oxford, UK, 2008; ISBN 978-1405156547.
5. Mahamid, I.; Bruland, A.; Dmaid, N. Causes of Delay in Road Construction Projects. *J. Manag. Eng.* **2012**, *28*, 300–310. [[CrossRef](#)]
6. Zidane, Y.J.T.; Andersen, B. The top 10 universal delay factors in construction projects. *Int. J. Manag. Proj. Bus.* **2018**, *11*, 650–672. [[CrossRef](#)]



7. Malcolm, D.G.; Roseboom, J.H.; Clark, C.E.; Fazar, W. Application of a Technique for Research and Development Program Evaluation. *Oper. Res.* **1959**, *7*, 646–669. [[CrossRef](#)]
8. Pontrandolfo, P. Project duration in stochastic networks by the PERT-path technique. *Int. J. Proj. Manag.* **2000**, *18*, 215–222. [[CrossRef](#)]
9. Acebes, F.; Pereda, M.; Poza, D.; Pajares, J.; Galán, J.M. Stochastic earned value analysis using Monte Carlo simulation and statistical learning techniques. *Int. J. Proj. Manag.* **2015**, *33*, 1597–1609. [[CrossRef](#)]
10. Goldratt, E.M. *Critical Chain: A Business Novel*; North River Press: Great Barrington, MA, USA, 1997; ISBN 0566080389.
11. Vanhoucke, M. *Measuring Time—Improving Project Performance Using Earned Value Management*; Springer: Berlin, Germany, 2010; ISBN 978-1-4419-1014-1.
12. Ballesteros-Pérez, P.; Cerezo-Narváez, A.; Otero-Mateo, M.; Pastor-Fernández, A.; Vanhoucke, M. Performance comparison of activity sensitivity metrics in schedule risk analysis. *Autom. Constr.* **2019**, *106*, 102906. [[CrossRef](#)]
13. Ballesteros-Pérez, P. M-PERT: Manual project duration estimation technique for teaching scheduling basics. *J. Constr. Eng. Manag.* **2017**, *143*, 04017063. [[CrossRef](#)]
14. Taylor, F.W. Shop Management. *ASME Trans.* **1903**, *24*, 1337–1480.
15. Kelley, J.; Walker, M. Critical-Path Planning and Scheduling. In Proceedings of the Eastern Joint Computer Conference, Boston, MA, USA, 1–3 December 1959.
16. Wilson, J.M. Gantt charts: A centenary appreciation. *Eur. J. Oper. Res.* **2003**, *149*, 430–437. [[CrossRef](#)]
17. Ballesteros-Pérez, P.; Larsen, G.D.; González-Cruz, M.C. Do projects really end late? On the shortcomings of the classical scheduling techniques. *J. Technol. Sci. Educ.* **2018**, *8*, 86–102. [[CrossRef](#)]
18. Ballesteros-Pérez, P.; González-Cruz, M.C.; Cañavate-Grimal, A. Mathematical relationships between scoring parameters in capped tendering. *Int. J. Proj. Manag.* **2012**, *30*, 850–862. [[CrossRef](#)]
19. Ballesteros-Pérez, P.; del Campo-Hitschfeld, M.L.; Mora-Melia, D.; Domínguez, D. Modeling bidding competitiveness and position performance in multi-attribute construction auctions. *Oper. Res. Perspect.* **2015**, *2*, 24–35. [[CrossRef](#)]
20. Ballesteros-Pérez, P.; González-Cruz, M.C.; Fernández-Diego, M.; Pellicer, E. Estimating future bidding performance of competitor bidders in capped tenders. *J. Civ. Eng. Manag.* **2014**, *20*, 702–713. [[CrossRef](#)]
21. Chudley, R.; Greeno, R. *Building Construction Handbook*, 11th ed.; Routledge: Abingdon, UK, 2016; ISBN 113890709X.
22. Jensen, J.L.W.V. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Math.* **1906**, *30*, 175–193. [[CrossRef](#)]
23. Hajdu, M. Effects of the application of activity calendars on the distribution of project duration in PERT networks. *Autom. Constr.* **2013**, *35*, 397–404. [[CrossRef](#)]
24. Nelson, R.G.; Azaon, A.; Aref, S. The use of a GERT based method to model concurrent product development processes. *Eur. J. Oper. Res.* **2016**, *250*, 566–578. [[CrossRef](#)]
25. Khamooshi, H.; Cioffi, D.F. Uncertainty in Task Duration and Cost Estimates: Fusion of Probabilistic Forecasts and Deterministic Scheduling. *J. Constr. Eng. Manag.* **2013**, *139*, 488–497. [[CrossRef](#)]
26. Clark, C.E. The Greatest of a Finite Set of Random Variables. *Oper. Res.* **1961**, *9*, 145–162. [[CrossRef](#)]
27. Clark, C.E. Letter to the Editor—The PERT Model for the Distribution of an Activity Time. *Oper. Res.* **1962**, *10*, 405–406. [[CrossRef](#)]
28. Pritsker, A.A.B. *GERT: Graphical Evaluation and Review Technique (Rand Corporation. Memorandum RM-4973-NASA)*; Rand Corp.: Santa Monica, CA, USA, 1966.
29. Herrerías-Velasco, J.M.; Herrerías-Pleguezuelo, R.; Van Dorp, J.R. Revisiting the PERT mean and variance. *Eur. J. Oper. Res.* **2011**, *210*, 448–451. [[CrossRef](#)]
30. Ballesteros-Pérez, P.; Sanz-Ablanedo, E.; Soetanto, R.; González-Cruz, M.C.; Larsen, G.D.; Cerezo-Narváez, A. Duration and cost variability of construction activities: An empirical study. *J. Constr. Eng. Manag.* **2020**, *146*, 04019093. [[CrossRef](#)]
31. Ballesteros-Pérez, P.; Sanz-Ablanedo, E.; Mora-Melià, D.; González-Cruz, M.C.; Fuentes-Bargues, J.L.; Pellicer, E. Earned Schedule min-max: Two new EVM metrics for monitoring and controlling projects. *Autom. Constr.* **2019**, *103*, 279–290. [[CrossRef](#)]
32. Demeulemeester, E.; Vanhoucke, M.; Herroelen, W. RanGen: A random network generator for activity-on-the-node networks. *J. Sched.* **2003**, *6*, 17–38. [[CrossRef](#)]

33. Vanhoucke, M.; Coelho, J.; Debels, D.; Maenhout, B.; Tavares, L.V. An evaluation of the adequacy of project network generators with systematically sampled networks. *Eur. J. Oper. Res.* **2008**, *187*, 511–524. [[CrossRef](#)]
34. Vanhoucke, M. On the dynamic use of project performance and schedule risk information during project tracking. *Omega* **2011**, *39*, 416–426. [[CrossRef](#)]
35. Wauters, M.; Vanhoucke, M. Study of the stability of earned value management forecasting. *J. Constr. Eng. Manag.* **2014**, *141*, 04014086. [[CrossRef](#)]
36. Colin, J.; Vanhoucke, M. Setting tolerance limits for statistical project control using earned value management. *Omega* **2014**, *49*, 107–122. [[CrossRef](#)]
37. Elshaer, R. Impact of sensitivity information on the prediction of project's duration using earned schedule method. *Int. J. Proj. Manag.* **2013**, *31*, 579–588. [[CrossRef](#)]
38. Tavares, L.V.; Ferreira, J.A.; Coelho, J.S. The risk of delay of a project in terms of the morphology of its network. *Eur. J. Oper. Res.* **1999**, *119*, 510–537. [[CrossRef](#)]
39. Trietsch, D.; Mazmanyan, L.; Gevorgyan, L.; Baker, K.R. Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation. *Eur. J. Oper. Res.* **2012**, *216*, 386–396. [[CrossRef](#)]
40. Batselier, J.; Vanhoucke, M. Construction and evaluation framework for a real-life project database. *Int. J. Proj. Manag.* **2015**, *33*, 697–710. [[CrossRef](#)]
41. Vanhoucke, M.; Coelho, J.; Batselier, J. An Overview of Project Data for Integrated Project Management and Control. *J. Mod. Proj. Manag.* **2016**, *3*, 6–21.
42. Operations Research & Scheduling Research Group. OR-AS. Be Real Project Data. Available online: <http://www.Projectmanagement.Ugent.Be/?Q=research/Data/Realdata> (accessed on 4 November 2019).
43. Lu, M.; Lam, H.-C. Transform Schemes Applied on Non-Finish-to-Start Logical Relationships in Project Network Diagrams. *J. Constr. Eng. Manag.* **2009**, *135*, 863–873. [[CrossRef](#)]
44. Hajdu, M.; Bokor, O. The Effects of Different Activity Distributions on Project Duration in PERT Networks. *Procedia Soc. Behav. Sci.* **2014**, *119*, 766–775. [[CrossRef](#)]
45. MacCrimmon, K.R.; Ryavec, C.A. An Analytical Study of the PERT Assumptions. *Oper. Res.* **1964**, *12*, 16–37. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).