Performance comparison of activity sensitivity metrics in Schedule Risk Analysis

Article *in* Automation in Construction · July 2019

DOI: 10.1016/j.autcon.2019.102906

P. Ballesteros-Pérez
Universitat Politècnica de València

Alberto Cerezo-Narváez
Universidad de Cádiz

Manuel Otero-Mateo
Universidad de Cádiz

A. Pastor
Universidad de Cádiz

Performance comparison of activity sensitivity metrics in

Schedule Risk Analysis

Pablo Ballesteros-Pérez[1*]; Alberto Cerezo-Narváez[2]; Manuel Otero-Mateo[3];

Andrés Pastor-Fernández[4]; Mario Vanhoucke[5]

[1,2,3,4] *Dpto. de Ingeniería Mecánica y Diseño Industrial, Escuela Superior de Ingeniería.*

Universidad de Cádiz, Avda. Universidad de Cádiz 10, Puerto Real, 11519 Cádiz (Spain)

[1*] Senior researcher, +34 956 483 200, pablo.ballesteros@uca.es (corresponding author)

[2] Assistant Professor, +34 956 483 311, alberto.cerezo@uca.es

[3] Assistant Professor, +34 956 483 200, manuel.otero@uca.es

[4] Associate Professor, +34 956 483 211, andres.pastor@uca.es

[5] *Ghent University, Tweekerkenstraat 2, 9000 Gent (Belgium)*

Vlerick Business School, Reep 1, 9000 Gent (Belgium)

UCL School of Management, University College London, 1 Canada Square, London E14 5AA (UK)

[5] Professor, +32 926 435 69, mario.vanhoucke@ugent.be

Performance comparison of activity sensitivity metrics in

Schedule Risk Analysis

Highlights

- A performance comparison of all relevant SRA metrics to date is presented.
- A new SRA metric named Criticality-Slack-Sensitivity index (*CSS*) is proposed.
- *CSS* is the top-performing metric in one-off calculation mode.
- The Schedule Sensitivity Index (*SSI*) is the best metric in iterative calculation.
- The duration of construction projects can be significantly shortened using SRA.

Performance comparison of activity sensitivity metrics in Schedule Risk Analysis

Abstract

In Schedule Risk Analysis (SRA), activity sensitivity metrics measure the importance of activities in a project schedule. Highly sensitive activities are those more likely to increase project duration variability and/or cause project duration extensions. Several activity sensitivity metrics have been proposed over the years, but a comparison of all of them has never been made. This has made it difficult to know which metrics perform better and under what circumstances.

In this paper, an extensive comparison of all relevant SRA activity sensitivity metrics is performed using a set of 4100 artificial projects. Unlike previous studies, the comparison framework is decoupled from corrective actions (e.g. activity crashing) which allows the merits of each metric to be assessed individually. Additionally, a new metric that performs better for overall sensitivity ranking is proposed. Results show that most sensitivity metrics do not perform well unless they are applied iteratively (the sensitivity of the remaining scheduled activities has to be recalculated whenever the duration variability of at least one activity has been restricted). However, if applied iteratively, most metrics can enhance project monitoring and control, while significantly shortening project duration.

Keywords: scheduling; schedule risk analysis; activity sensitivity; project delays; project control.

1. Introduction

Schedule risk analysis (SRA) is a simulation technique that allows project managers to identify the critical schedule components that may have the biggest impact on project objectives [1]. SRA is a prominent project monitoring and control technique, maybe only surpassed in popularity by the Earned Value Management (EVM), a technique with which SRA can be combined resulting in what is known as *Dynamic project scheduling* [2].

However, whereas EVM is predominantly a 'reactive' technique (EVM measures actual time and/or cost deviations with respect to a baseline), SRA is a proactive technique. SRA identifies which schedule components (normally a subset of activities) are key to delivering the project on time and/or budget even before those components have actually started. Identifying those key components allows the project manager to focus on the project tasks that really matter, that is, to be more efficient.

Economy and productivity have always been crucial aspects of construction management [3], but even so, projects ending late and/or suffering from cost overruns are still a widespread phenomenon [4]. Maybe not that surprisingly though, poor planning, monitoring and control practices usually stand out as the most relevant factors causing project overruns [5]. In this context, every tool that shows promise in alleviating this problem is worth considering.

Over the years, SRA has produced a series of metrics that allow the activities that are more 'important' than others to be discriminated mathematically. Scientific literature frequently refers to these as activity 'sensitivity' measures or just 'SRA metrics' [6]. Basically, these metrics rank activities by giving them a number (generally ranging from 0 to 1) reflecting their relative importance. Once these values are known, the project manager can also set a numerical threshold, which can be dynamically adjusted later, if necessary. All activities whose metric value exceeds the threshold should be monitored more closely during execution. By 'monitoring' researchers generally mean that it should be ensured that those activity durations do not exceed their planned durations [7], otherwise the project duration and/or cost will surely be negatively impacted.

Previous comparisons of these metrics have also simulated the effect of some corrective actions that are taken when highly sensitive activities suffer from time overruns [6]. This approach has some advantages when reflecting on how these metrics can be applied in real contexts. However, it also has a critical disadvantage: the effects of SRA are mixed up with the outputs of those corrective actions. This makes it very difficult to distinguish which are the real benefits of implementing SRA on its own, and also, under which circumstances each SRA metric performs best.

In the same vein, previous studies have combined SRA with some scheduling compression techniques (e.g. activity crashing, activity fast-tracking or activity substitution) [8]. Scheduling compression techniques undoubtedly have an important place in project planning, monitoring and control. However, they serve a very different purpose: to shorten the schedule either beforehand and/or during the project execution stage. Project managers always try to find a balance between a sufficiently short project duration and the increased risk, money or resources that such a schedule configuration involves. Once a suitable balance is found (because there is no more money, resources, or just because it becomes too risky or technically impossible to shorten it any more), SRA can be implemented to measure the activity sensitivity and ensure that the project is delivered as planned.

Consequently, the aim of SRA is to *ensure that the actual activity durations are as close as possible to their planned durations*. Shorter durations, while not harmful, are much less likely than delays. Hence, the purpose of SRA is to identify those activities whose potential duration variability needs to be reduced (constrained). How these durations are constrained in practice is outside the scope of this paper. However, it generally involves tighter and more frequent activity progress control as well as some pre-specified back-up plans if any highly sensitive activities suffer delays.

As a result, the real foe in project control is duration variability. There have been many studies pointing out exactly the same. Among the most recent, Ballesteros-Pérez et al. [9] demonstrated how classical (deterministic) scheduling techniques generally underestimate project duration and cost, by neglecting activity duration variability. Later, Ballesteros-Pérez et al. [10], on measuring the ratios of actual vs. planned activity durations on a wide set of construction projects, determined that the coefficient of variation is around 60%. These authors also proved that this (unconstrained) activity duration variability is enough to increase the duration of construction projects by an average of 20% and the project cost by at least 7%. All these facts make clear that it is necessary to start paying more attention to activity duration variability and the tools that can handle it effectively. Among these tools, SRA is arguably the most effective.

Finally, there have been many other pieces of research dealing with SRA at project-level. Project-level aspects are generally dependent on the project type and/or some contextual information that is not generally easy to generalise and/or model mathematically [11]. Hence, the scope of this paper is restricted to activity-level sensitivity metrics only. Project level aspects, while undoubtedly important, will be left for future research.

Therefore, in this paper, a performance comparison of all activity-based sensitivity metrics published to date is carried out. A metric's performance is understood as its capacity to restrict project duration variability and shorten average project duration. The comparison framework adopted will also show how effective SRA metrics are, regardless of other scheduling compression techniques. Other aspects considered will be: how the performance of the SRA metrics increases as more activities have their durations constrained (an effort is made to make their actual duration equal to the planned duration), as well as the influence of network topology (what the network looks like in a project schedule). In order to obtain realistic and representative results, a varied set of 4100 project network schedules is used in this study. Activity duration variability in these network schedules will be modelled to resemble that of actual construction projects. This will further enhance the representativeness of the results obtained.

The paper will be structured as follows. In the *Literature review* section, the three subsections will go over the mathematical notation, the details of the SRA metrics compared, and the results from previous (partial) performance comparison studies. A new SRA metric will also be proposed and mathematically defined in this section. In the *Materials and methods* section, the artificial project dataset as well as the activity duration modelling will be described. Then, the performance framework for measuring the effectiveness of all SRA metrics will be outlined. The *Results* section will summarise all the metrics performance results under different scenarios (different levels of project control and network topologies) as well as how the SRA metrics are calculated (one-off vs iteratively). The *Discussion* section will propose complementary SRA approaches and justify how current metrics still show a significant potential for improvement. Finally, the *Conclusions* will summarise the research analysis and contributions, state the limitations and suggest some research continuations.

2. Literature review

2.1. Existing activity-based SRA metrics

The eight metrics whose performance is compared are described in this section. The most relevant variables and mathematical expressions will be defined contextually, instead of at the outset. This approach will make it easier to remember their meanings later. Nevertheless, for easier reference, a comprehensive description of all variables and abbreviations is listed and explained in alphabetical order as *Supplemental online material*.

SRA metrics can only be calculated by (Monte Carlo) simulation. The following notation will be common to all metrics and is defined here:

i refers to each activity in a construction schedule (project) with i=1, 2, … *n*.

n is the total number of activities in a construction schedule.

j refers to each Monte Carlo simulation run when calculating the SRA metrics and project durations with j=1, 2, … *N*.

N is the total number of Monte Carlo simulations performed.

k refers to the number of activities whose duration variability (σ_i) will be *constrained* with k=0, 1, 2, … *n*.

σ_i represents the standard deviation of activity *i*'s durations in the *N* simulation runs. When an activity is *constrained*, it will mean that its possible activity (stochastic) duration will always be forced to equal its planned duration. In this context, when σ_i is said to be constrained, what is really meant is that σ_i=0, and activity *i*'s duration will become a deterministic variable (constant throughout the simulation runs and equal to its planned/baseline duration).

With this preliminary notation, all the SRA metrics to be compared are presented in Table 1. To avoid information cluttering, the simulation-based estimators of all metrics have been presented as *Supplemental online material*.

<Insert Table 1 here>

The last row of Table 1 contains the Criticality-Slack-Sensitivity index (*CSS*), a new metric proposed in this study. The *CSS* constitutes an improvement of the *SSI* and *MOI* metrics by adding a third term considering the difference between activity *i*'s slack when all activity durations are stochastic ($E(s_i)$) versus deterministic (s'_i). The expression of the *CSS* (extracted from Table 1) is:

$$CSS_i = SSI_i \cdot \frac{E(s_i) - s'_i}{E(PD)} = CI_i \cdot \frac{E(s_i) - s'_i}{E(PD)} \cdot \frac{\sigma_{d_i}}{\sigma_{PD}} \qquad (1)$$

In particular, this new index presents three terms, each responsible for one task. The *CI* term attributes more importance to those activities that are more frequently critical. The difference between the stochastic and deterministic slacks indirectly measures the average impact of the merge event bias in activity *i*, that is, how much the variability of all project activities allows activity *i* to shift. If this term is zero, this can be because either activity *i* is always critical or is never critical. In the first case, both $E(s_i)$ and s'_i equal 0. In the second case, $E(s_i)=s'_i$. However, in neither case will the activity contribute to minimising the merge event bias, that is, to reducing the Project duration average (it might reduce the project duration variability though, but only if CI_i=1). Finally, the third term (the ratio of duration standard deviations) reflects the proportion of project duration variability that can be controlled by the activity *i* itself (not by other activities). This term has been inherited from the *SSI*.

Apart from the SRA metrics described in Table 1, other alternative approaches can be found in the literature. For example, it has been suggested by a few researchers that combining some of the SRA metrics described in Table 1 could enhance their performance [15]. In this regard, it was reported by Liu and Wang [11], that Yangbin Ou, in an internal dissertation in 2003 (that could not be accessed by the authors of this paper), proposed and tested a composite metric named Activity Compound Criticality Index (*ACCI*). This metric apparently corresponded to the product of *CI* and *CRI(r)*. As a precautionary action, the performance of pairs of all the SRA metrics described above was tested by the authors of this paper. For the sake of clarity and brevity, those auxiliary experiments have not been included in the results. However, it is worth highlighting that no combination showed a higher performance than that of the top performing metric out of the two metrics being multiplied.

Furthermore, there have been a few examples of other metrics whose calculation involves some kind of (subjective) judgemental input. Among these, probably the most relevant is the Activity Critical Comprehensive Index (*ACCI* [1]) proposed by Cui et al. [16]. This *ACCI* consists of three additive terms that measure the relative importance that the project manager wants to attribute to the average project (1) duration, (2) variance and (3) activity criticality, with respect to the (1) duration, (2) variance and (3) criticality of the longest path it belongs to, respectively. The three terms are quite simplistic and the idea is that the scheduler decides which one he/she wants to prioritise. However, the three terms are actually encompassed (in one way or another) in the expression of the previous eight metrics. For this reason, these composite additive metrics have not been explored further.

Finally, differing significantly from previous approaches, but also requiring some subjective input, other authors have measured activity importance by measuring how each activity could contribute to an increase in project duration variability. In this vein, Cho and Yum [17] developed a Taguchi tolerance-based design technique that could be implemented manually, but whose calculation actually takes quite a lot longer than performing Monte Carlo simulations. The analysis performed here will also analyse project variability by choosing activities according to their ranked SRA metric values. However, this study will mostly focus on how SRA metrics can shorten project duration (not just its variability). Hence, these alternative metrics will no longer be considered.

2.2. Previous SRA performance comparison studies

While there have been many studies discussing the advantages and limitations of some SRA metrics (e.g. [6,13,15]), the purpose of this paper is not to recount the latter.

[1] This metric has the same abbreviation as the Activity Compound Criticality Index, but they actually have nothing in common.

Numerical comparison of activity-based sensitivity metrics, on the other hand, have been in short supply. To date, only Vanhoucke [1,2,7] and Madadi and Iranmanesh [14] have attempted to measure the performance of SRA metrics by resorting to relatively large and representative network datasets.

The studies by Vanhoucke [1,2,7] were the first to perform a thorough comparison of the first six metrics (the *MOI* and *CSS* were not included). Vanhoucke measured the performance of these metrics by comparing how different threshold metric values (during a simulated project execution) allowed activities requiring some type of intervention (normally to be shortened to bring the project back on track) to be flagged. In particular, Vanhoucke resorted to an index named Unit Contribution (UC). The UC was defined as the decrease in the number of time units (e.g. days) of the project duration divided by the decrease in the total number of time units of all controlled activities resulting from the corrective actions adopted. For instance, whenever a highly sensitive activity (considered as such by having exceeded a threshold value) experienced a duration overrun, its activity duration was halved. Then, the resulting decrease in total project duration (with respect to average project duration) was measured. This approach was satisfactory to show that SRA metrics are indeed useful and that some seem to perform better than others. The author also created the network dataset that will be used later in this study and proved numerically how network topology significantly conditions the effectiveness of SRA. However, Vanhoucke's studies had the following limitations.

First, the SRA metrics were used as action thresholds, which does not provide any insight into whether the metric values are proportional or just roughly rank the activities' sensitivity.

Second, the studies left unanswered how often SRA metrics need to be recalculated (once at the beginning? Once at every tracking period? Once after any activity suffers a deviation (no matter how small) from the project baseline?). These questions will be answered later in this study.

Third, Vanhoucke's performance analyses always involved activity crashing (whenever a highly sensitive activity was delayed and had to be brought back on track). As described earlier, this approach mixed the contributions of crashing with SRA, making it difficult to distinguish what was the result of what. Here, the effectiveness of the SRA metrics will be analysed separately from any corrective action.

Fourth, the number of simulations from Vanhoucke's studies generally were of 100 runs per simulated project. With this number of simulations, a relative error of around 10% is to be expected (the relative error is measured as the standard deviation of the Monte Carlo estimates with respect to their actual value). In this study, 10,000 simulations are used per SRA metric and project. Since the error of Monte Carlo estimates is proportional to $1/\sqrt{N}$ [18], this will reduce errors to approximately a tenth.

Fifth, Vanhoucke's studies did not include the *MOI* metric, nor the *CSS* metric as the latter has been proposed here for the first time. Moreover, the results from this study later do not entirely agree with Vanhoucke's performance results, which also merits closer inspection.

Two years later, Madadi and Iranmanesh [14] resorted to a different but smaller network dataset and compared the performance of the *MOI* against the *CI*, *SI* and *CRI*(r) metrics (neither the *CRI*(ρ), *CRI*(τ), *SSI*, nor *CSS* were included). However, they also (indirectly) measured the effect of constraining some activities on reducing both the project duration mean and its variability.

According to the studies briefly described above, the top performing metrics were the *SSI* (for Vanhoucke) and the *MOI* (for Madadi and Iranmanesh). The performance analysis in this paper will compare both for the first time, while also considering the (direct) effect of constraining activity duration on both the average project duration and its variability.

3. Materials and methods

Hulett [19] was one of the first to set clear directions on how SRA should be implemented. He defined four sequential steps which are briefly outlined here:

1. Define the baseline schedule, which will act as the point of reference for subsequent simulation runs.
2. Define activity duration (and cost) uncertainty by means of defining the statistical distributions that model those activity durations (and costs) for each activity.
3. Run (Monte Carlo) simulations. In each run, the activities (and in consequence, the project) will have different durations and costs (and probably a different critical path).
4. Sensitivity output. With the information stored from the previous step through many simulation runs, it is then possible to calculate the activity sensitivity metrics.

In this study, these four steps were repeated for each artificial schedule network (4100 projects), for each SRA metric (8 plus one that ranks the activities randomly), and for two calculation modes (one-off and iteratively). *One-off* means that all SRA metrics were calculated just once at the beginning of each project. *Iteratively* means that all metrics were recalculated as activity duration variabilities were constrained one by one, that is up to 30 times (activities) per project. The *iterative* mode of calculation was included to assess whether the remaining activity SRA metric values become unreliable when the project manager attempts to keep the duration of some activities as planned (on time). Finally, as stated earlier, the number of simulation runs from step 3 was always 10,000.

3.1. Simulated projects dataset

The artificial projects dataset consisted of 4,100 activity-on-node networks with 30 activities each. Each network had two dummy activities (with zero duration) signalling the project start and end. This dataset, along with other instances of artificial projects (see [20] for an overview), was developed by the Ghent University Operations Research & Scheduling Research Group and can be downloaded here: http://www.projectmanagement.ugent.be/research/data/RanGen (MT set). From each network (project), a file can be found containing all the predecessors activity information.

The project dataset was generated with the RanGen2 algorithm. RanGen2 is a robust random network generator validated in several studies [21,22] and capable of generating a wide range of different network topologies. The same set of projects has been used in many recent research studies on SRA (e.g. [23,24]) and EVM (e.g. [2,25,26]).

In particular, the project 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*). The *SP* indicator describes how close a network is to a serial or parallel network. The *AD* describes the distribution of activities in the different network paths. The *LA* measures the distance between two activities in the project network. The *TF* measures the slack or float activities have at a topological level, that is, how dense the network is. All indicators range from 0% to 100%. These four topological indicators were initially proposed by Vanhoucke et al. [22] and slightly refined in Vanhoucke [23]. They are considered representative and accurate descriptors of a network topology. For the interested reader, the values of all four indicators can be found for the 4100 network instances as *Supplemental online material*.

The different project networks were generated by setting specific staggered values of the *SP* indicator from *SP*=0 (all project activities are in parallel) to SP=100% (all activities are in series). While the *SP* was set, the other indicators (*AD*, *LA* and *TF*) could vary freely when searching for new random network configurations. Namely, the series of *SP* values used were 7%, 17%, 28%, 38%, 48%, 59%, 69%, 79%, and 90%. Extremes (0% and 100%)

were not included in the analyses as they are not considered representative of real construction projects. Also, rounded *SP* values (e.g. 10%, 20%, 30%...) were not possible due to the fixed number of activities per project (30).

Concerning the stochastic generation of activity durations, many statistical distributions have been used in the past (Uniform, Beta, Normal, Triangular, etc.) [1,6,27]. Log-Normal distributions were used here for the following reasons. Log-Normal distributions, by definition, cannot produce negative durations, but still allow for values located far from the distribution average. Log-Normal distributions are quite simple (they depend on two parameters only) but recent empirical studies have shown that they still satisfactorily model construction activity duration variability [28,29]. With this in mind, the Log-Normally-distributed activity durations were generated using this expression:

$$d_i \sim \mu_i \cdot 10^{Normal\left(mean=0,\, CV_i\right)} \qquad (2)$$

Where μ_i represents each activity *i*'s average duration average (the planned/baseline duration) and CV_i the coefficient of variation of a Normal distribution with zero mean (and a standard deviation of 1). The second term of expression 2 (the 10^Normal distribution) generates log-normally-distributed multipliers. This Normal distribution had a zero mean because this way the average stochastic activity durations coincided (on average) with their planned/baseline durations, as $\mu_i \cdot 10^0 = \mu_i \cdot 1 = \mu_i$. This kept the projects from ending systematically sooner or later. A base of 10 was used here because, along with Euler's number, it is the most common logarithmic base.

In the proposed simulation framework, all activities are scheduled to start as soon as possible and activity preemption is not allowed; the latter to avoid randomly affecting activity duration variability. Additionally, μ_i and CV_i values were generated beforehand for the 30 activities of the 4100 networks (30·4100=123,000 pairs of different μ_i and CV_i values). Only with this approach was it possible to ensure that each activity had exactly the same average duration and variability when different SRA metrics were compared in different simulation runs. In particular, the activity planned/baseline duration μ_i values were arbitrarily and stochastically generated following a Normal distribution with mean 100 (e.g. days) and standard deviation 20 (e.g. days). CV_i values, on the other hand, were generated following a Uniform distribution ranging between 0.10 and 0.30. The latter range was adopted so that the activity duration variability emulated the same levels of variability as those observed in a sample of 101 projects by Ballesteros-Pérez et al. [10]. More precisely, those authors identified that most activity duration variability lies in the range (in Log-scale with base 10) between 0.1 (low variability) and 0.3 (high variability), with the average being 0.2.

Therefore, unlike μ_i whose values hardly make any difference to the results, the CV_i values chosen greatly influence the results' representativeness. As CV_i values were carefully chosen to resemble those of real construction projects, it is expected that the results here will also provide a realistic order of magnitude of how the duration of real construction projects can be shortened by constraining the duration variability of the most sensitive activities. For those readers interested in knowing how the same SRA metrics would perform under scenarios with strictly lower and higher activity duration variability (CV_i=0.10 and 0.30, respectively), the same simulation results can be found as *Supplemental online material*. Moving forward, only the case of CV_i varying uniformly between 0.10 and 0.30 has been presented.

Finally, it may be worth noting that in this study the activity cost dimension has been intentionally neglected in the sensitivity analyses. This is a common trait in most SRA studies as, generally, the cost dimension is much simpler than the time dimension. Whereas activity order matters in time analysis, the cost is merely an additive variable. The project cost generally resembles a Normal distribution whose average and standard deviation can be closely approximated by the sum of averages and cost variances of all activity cost distributions [30]. This conjecture was recently confirmed by Batselier and Vanhoucke [31] in an empirical study involving 52 projects. This means that the network topology does not influence project cost either, unless there is a significant correlation between project duration and project cost. However, while this correlation seems to exist at activity level, it does not seem significant at project level [10]. As a result, the cost dimension in this and future SRA metrics comparisons can be safely neglected without any loss of representativeness.

3.2.Performance measurement framework

For each simulation run, all activities had their SRA metrics calculated (once in the *one-off* calculation mode, or multiple times in the *iterative* mode). Then, activities were ranked by decreasing order of the value of each SRA metric. This means that, by decreasing order of one SRA metric at a time, k activities out of the total n activities per schedule, had their duration variability constrained (their stochastic durations were forced to remain constant and equal to the planned durations, that is, to equal to μ_i). Experiments were repeated testing all SRA metrics and the whole range of k activities from 0 (no activities with duration variability constrained) up to 30 (all activities in the schedule had their duration variability constrained).

With regards to metric performance measurements, two variables were registered for each simulation run: the project duration median percentile reduction (Δm_k) and the project duration standard deviation reduction ($\Delta\sigma_k$). Results for both variables and for all 4100 projects can be found as *Supplemental online material*. Results in the paper will only report their average Δm_k and $\Delta\sigma_k$ values. Also, due to its particular relevance, the calculation procedure of Δm_k is represented in Figure 1.

<Insert Figure 1 here>

Figure 1 shows two (probabilistic) project duration curves, both of which are obtained with N (10,000 here) Monte Carlo simulation runs. The one on the right represents the project when all its activity durations can vary freely (no activity durations have been constrained). The curve on the left represents the project duration when k activities have had their duration variability constrained (that is $d_i = \mu_i$). Hence, Δm_k measures the difference of the project duration median (measured in probability, that is, as a reduction in percentiles) between two scenarios: a project with k activities constrained and the original project duration curve (when no activity durations had been constrained yet). This can be formulated mathematically as:

$$\Delta m_k = 0.5 - \mathrm{Prob}^{k=0}(m_k) \quad \text{with } k = 0, 1, 2 \ldots n \tag{3}$$

Hence, Δm_k represents the (negative) increment between two probability values. The median is chosen as a more reliable indicator because, unlike the Project Duration average, it is always associated with the same probability value (50^{th} percentile = 0.5).

The second performance variable is quite simple and represents the project duration standard deviation reduction (in the N simulation runs) when k activities have been constrained (respect to the unconstrained schedule with k=0). It is defined as:

$$\Delta \sigma_k = 1 - \frac{\sigma_k}{\sigma_{k=0}} \quad \text{with } k = 0, 1, 2 \ldots n \tag{4}$$

Finally, and following a similar convention, from now on project control (*PC*) will be referred to as k/n, that is, the percentage of activities whose duration variability has been constrained. This variable will be useful in providing an estimate of the project manager's control effort when a monitoring the project execution.

4. Results

4.1.SRA metrics performance by project control level

The average performance results measured by Δm_k and $\Delta\sigma_k$ are presented in this section. Detailed results by project can be found in the *Supplemental online material*.

Figure 2 shows the first set of the performance results when all SRA metrics are calculated once off and by varying level of project control (*PC*). It may be worth remembering that *one-off* means that all SRA metrics were calculated just once at the beginning of each project. That is, they were used to rank all activities by decreasing order of importance at the outset, and they were never recalculated later as some activities had their durations constrained. This is relevant, as when a single activity is duration-constrained, the sensitivities of all remaining activities may also change.

<Insert Figure 2 here>

Figure 2 shows in two tables the Δm_k values (top) and $\Delta\sigma_k$ values (bottom). The percentage of activities that have had their duration variability constrained from *PC*=0% (none) up to 100% (all) is displayed in columns. All the SRA metrics compared are shown in rows, plus a random allocation of constrained activities at the top of each table. This row represents a baseline comparison that all SRA metrics should outperform (in this case by achieving lower values).

Values of Δm_k range from 0% (no project duration difference at all with the initial project duration median percentile) up to 35.8%. However, the latter value (35.8%) is conditioned by the level of activity variability that it was chosen for the activity durations when trying to resemble that of real construction projects. Also, obviously, when *PC*=0% or 100%, all metrics perform exactly the same because no or all activities are constrained. On average then, 35.8% is the maximum percentile a median project duration may reach when all project activities are kept perfectly on time.

On the other hand, project duration variability reduction (measured by the decrease in the project duration standard deviation), can range between 0% (no constrained activities) up to 100% (all activity durations have been constrained). The latter would correspond to a deterministic schedule.

Regarding metric performance, the effectiveness of these metrics increases with *PC* effort, as expected. However, their performance ceiling will be better assessed when the *iterative* calculations from Figure 3 are analysed later.

In Figure 2, the top performing metric, both in Δm_k and $\Delta \sigma_k$ values, is the new *CSS*. This metric, when calculated once off, achieves the maximum (average) project duration median and variability reduction with respect to the totally unconstrained schedule. The values of this metric are highlighted in bold. The second-best metric is the *CRI*(τ). This comes as a surprise because in Vanhoucke's [1,2,7] studies this metric was always among the worst performers. This is not actually seen to be the case for the one-off calculation mode. After the *CSS* and the *CRI*(τ), the remaining six metrics perform similarly, both for Δm_k and the $\Delta \sigma_k$.

Overall, maybe with the exception of *CSS* and *CRI*(τ), results suggest that all metrics become virtually blind after starting to constrain some activity durations. "Blind" here means that they become virtually useless, as they are no longer are capable of determining which activities are the most sensitive. This is essential in real-life project monitoring and control, as the scheduler needs to know that any change, even if this means ensuring that activities last as planned, will impact the sensitivities of the remaining (unconstrained) activities.

Figure 3 presents the same performance results, but with SRA metrics being recalculated as activity durations are being constrained (*iterative* calculation mode). In this figure, for the sake of clarity, only the first 9 activities with constrained durations (up to *PC*=9/30=30%) have been shown. This calculation approach is more computationally demanding than the one-off calculations (because SRA metrics calculations need to be repeated as many times as the activities are constrained). However, by looking at the corresponding values between Figures 2 and 3 (columns with *PC*=10, 20 and 30% mostly), it is evident that the performance of all metrics has improved significantly. All metrics (even the worst performing, Random excluded) achieve higher Δm_k and $\Delta\sigma_k$ values than those in Figure 2 with one-off calculations.

<Insert Figure 3 here>

There are also some changes in the ranking. The metrics that performed better before (*CSS* and *CRI*(τ)) are just average performers now. The best iterative metric is the *SSI*, closely followed by *CRI*(r) and *CRI*(ρ). The *CI* and *SI* are the worst performers. In this instance, and apart from the *CSS* and *MOI* results which were not included originally, all results fully agree with Vanhoucke's [1,2,7], but not with Madadi and Iranmanesh's [14]. The only explanation possible is that Vanhoucke must have resorted to *iterative* calculations, whereas Madadi and Iranmanesh must have used *one-off* calculations.

Additionally, from Figure 3 it is possible to approximate the performance *ceilings* of all SRA metrics. Let us take a closer look, for example, at the *PC*=30% column. This column represents the project duration median probability reduction (top table) and project duration variability reduction (bottom table) that a project manager could achieve if a tight control was kept on 30% of the project activities. This level of control seems representative and feasible in real projects. In this column, the *SSI* achieved Δm_k=27.1% and $\Delta\sigma_k$=52.3%. The same values in one-off calculations for the *CSS* were Δm_k=19.6% and $\Delta\sigma_k$=28.1%. And the equivalent values for *Random allocation* are Δm_k=13.1% and $\Delta\sigma_k$=14.7%. Overall, this shows that the top performing metric in one-off calculation (the *CSS*) is approximately halfway between not being effective at all (represented by the random allocation results) and the best performance possible (those from the *SSI* in iterative mode).

At this point it is also convenient to remember that combinations of pairs of the eight metrics above were also tested (but not reported here). However, none clearly outperformed the current ones. This might mean that, despite each metric having different sensitivity detection mechanisms, the *SSI* seems to represent the current performance ceiling. That is, with the activity variability levels adopted, no better values of Δm_k and $\Delta \sigma_k$ could be found. The obvious next research step then would be to try to propose an SRA metric that could perform like the iterative *SSI* but when applied one-off. This is something the new *CSS* has not fully achieved, but it seems to be on the right track.

4.2.SRA metrics performance by Serial-Parallel and Project Control level

The same 4100 networks analysed can be reorganised by their Serial-Parallel (*SP*) indicator and by Project Control (*PC*) level. As described earlier, the *SP* describes how close a network is to a perfectly serial (*SP*=100%) or parallel network (*SP*=0%). *SP* is calculated as the number of activities in the path with the highest number of activities (which may not be the longest in duration) minus 1, divided by the total number of activities in the schedule minus 1, that is *n*-1 (dummy activities are not considered). The *SP* is probably one of the simplest, yet most relevant topological indicators. Vanhoucke [7] proved in his studies that the *SP* seems to greatly condition the tracking efficiency of both the EVM and SRA techniques. This author also conducted an exploratory study of several construction projects and reported that most construction projects boast *SP* indicators ranging between 12% and 78%. Our simulated projects dataset indeed covers a slightly wider range of *SP* values (from 7% up to 90%).

Finally, in Vanhoucke's [7] performance comparison, probably the most comprehensive to date, approximately *PC*=27% of all activities were controlled. Results for up to *PC*=30% are shown in the following tables. However, further *PC* values can be found as *Supplemental online material*.

As a result, Figures 4 and 5 show the SRA metrics performance results (only Δm_k values are shown on this occasion) assuming calculations are *one-off* (Figure 4) and *iterative* (Figure 5).

<Insert Figure 4 here>

<Insert Figure 5 here>

Quick inspection of both figures confirms again that as the level of *PC* control increases, Δm_k values also increase. Similarly, it is evident that the SRA metrics lose effectiveness (SRA metrics have lower Δm_k values) as the *SP* increases, that is, as they become closer to serial networks. Additionally, the *CSS* and the *CRI*(τ) are still the top performing metrics in the one-off calculations, but the *CSS* gains more advantage as the *PC* increases. In the iterative calculations, the *SSI* still seems to perform better, but is closely followed by *CRI*(r) and *CRI*(ρ).

Finally, a comparison of equivalent *PC* levels in both figures against random allocation confirms again that the top performing *one-off* metrics still have significant room for improvement (defined by the top performing *iterative* metrics, the *SSI* for instance).

5. Discussion

An enhanced SRA metric should reach Δm_k and $\Delta \sigma_k$ values similar to the ones observed for the *SSI* in Figures 3 and 5 (iterative calculations), but be achieved with one-off calculations. However, finding a mathematical expression that achieves this has proven to be quite elusive. There are two main reasons for this.

First, whenever an activity suffers a deviation from the baseline schedule, the whole schedule suffers some degree of change too. One-off numerical results shown earlier clearly support this as it is evident that SRA metrics become very unreliable as highly sensitive activities have their duration constrained.

Second, decreasing activity duration variability does not always lead to a decrease in project duration variability. Sometimes the latter can remain unaffected or, as proved by Gutierrez and Paul [32], it can also lead to an increase in project duration variability. Similarly, Elmaghraby et al. [33] experimentally showed that decreasing some activity average durations could cause project duration extensions on some (rare) occasions. This means that finding an activity or schedule (mathematical) attribute that is 'always' proportional to a project duration extension is certainly not straightforward.

Overall, both reasons demonstrate that finding a better SRA metric for one-off calculations is an extremely challenging task. Maybe because of this, the effectiveness of SRA has not been exempt from criticism. Some researchers have found that focusing solely on activity sensitivity may lead to erroneous assessment of activity importance [19,34]. To overcome this limitation, some researchers suggested shifting the point of attention to the potential risks that produce activity and/or project duration variability (risk-driven approach) [35]. This approach can bring some advantages (it is indeed more accurate if all risks are known beforehand). However, in real contexts, most risks are difficult to anticipate, let alone (probabilistically) estimate their impact before the project is executed. Fortunately, both approaches (activity sensitivity and risk-driven approach) are not exclusive, but rather complementary.

Other researchers have pointed out the obvious limitation of trying to measure activity importance based on a single figure (an index ranging from 0 to 1). The most notable works in this regard were the ones by Kuchta and Dorota [36] and Bowman [37–39]. Kuchta and Dorota [36] proposed a method to assess the criticality of activities with a fuzzy approach. Bowman, on the other hand, proposed a way of drawing the sensitivity curves of all activities. These sensitivity curves actually correspond to the representation of the Criticality Index (CI) as a function of activity duration. In his papers, Bowman devised a quite ingenious way of drawing those curves by resorting to a single set of Monte Carlo simulation runs. However, while useful, both Kuchta and Dorota's and Bowman's approaches did not allow to rank or prioritise the activities unless their impact was considered. Also, as the durations of some activities are constrained, the fuzzy calculations and sensitivity curves need to be updated too, and that is not possible unless more simulations are run. This means that their approaches suffer from exactly the same limitations as the SRA metrics compared above when calculated once off.

Finally, despite falling outside the scope of this paper, an interesting discussion would concern the extent to which activity duration variability should be constrained when monitoring and controlling the progress of real projects. Two recent works have recently focused on this front, but with very different approaches. First, Hu et al. [40] measured how the incorporation of activity sensitivity measurements into (Critical Chain) buffer management could lead to better project schedule risk management. Second, Martens and Vanhoucke [41] proved empirically that integrating some project-specific information (mostly resource availability) into the construction of control tolerance limits brings significantly higher monitoring efficiency. This means that, eventually, SRA cannot be implemented irrespective of resource availability aspects. These, and hopefully future studies, will enhance the relevance of the topic addressed in the present study. It is also clear that multiple avenues of research remain to be explored.

6. Conclusions

Schedule Risk Analysis (SRA) is a simulation technique that allows activity sensitivity to be measured with the intention of identifying those activities that require closer control during project execution. If an activity is highly sensitive, it is more likely that, if this activity is delayed, the whole project will also be delayed. Since 1963, several activity sensitivity metrics have been proposed, but previous performance comparison analyses have never involved all of them. Furthermore, previous comparisons have always involved corrective actions (mostly activity crashing), which made it difficult to quantify to what extent SRA was effective on its own.

In this paper, by resorting to a representative set of 4100 simulated projects, a systematic comparison of the most relevant activity-based SRA metrics published to date has been performed. The comparison is based on two performance indicators: the project duration median percentile reduction and the project duration standard deviation reduction. Both performance indicators quantify the (duration and variability) reduction achieved by a project whose activities are (partially or completely) duration-constrained versus the same project when its activity duration can vary freely (remain all unconstrained). Results have been derived and analysed by Project Control (*PC*) level and by staggered values of the Serial-Parallel (*SP*) indicator. Furthermore, performance measurements have involved two SRA metrics calculation modes: one-off and iterative. In the one-off calculation mode, the metrics are calculated just once (at the outset). In the iterative calculation mode, the metrics are recalculated whenever the schedule suffers any changes (e.g. after at least one activity has its duration constrained). Finally, for representativeness purposes, activity duration variability has been set to resemble that of real construction projects.

Results have confirmed that when the metrics are calculated once off, the top performing metric is the newly proposed Criticality-Slack-Sensitivity index (*CSS*) followed by the Cruciality Index with Kendall's tau ($CRI(\tau)$). These results seem to contradict previous performance studies. However, the performance of all metrics is approximately doubled when they are calculated iteratively, that is, when metrics are recalculated as activities have their duration variability constrained. This latter approach is, however, much more computationally demanding. Under the iterative calculation assumption, the top performing metric is the Schedule Sensitivity Index (*SSI*), closely followed by the *CRI*(*r*) and the *CRI*(ρ).

At a numerical level, results show that in construction projects, a 35.8% project duration median percentile reduction can be achieved against the original project duration when all activities are constrained (*PC*=100%). In a more representative case when only 30% of all activities are constrained (*PC*=30%), the median percentile can reach 19.6% (in one-off) or 27.1% (in iterative calculation mode). Both lead to significant project duration reductions. These are clear examples of the great benefits that project control can bring to construction managers.

Finally, whereas SRA metric performance, in the case of iterative calculation seems to have reached its full potential, there still seems to be room for improvement in more effective one-off metrics. Reasons for these statements have also been discussed.

A limitation of this study is obviously the lack of empirical validation. This is almost unavoidable, as real projects are only carried out once. Activity duration variability, unless provided by very experienced project schedulers, is very difficult to anticipate too. However, without such estimates, it is nearly impossible to calculate the SRA metrics for validation purposes from real project data. Similarly, it is also impossible to know which combination of constrained activities would have led to a shorter project duration in the presence of a single outcome: the as-built result (equivalent to a single simulation run). Overcoming these limitations may be an unsurmountable task. Therefore, in the absence of empirical validation, the comprehensive simulation approach taken here can hopefully provide strong evidence on the potential benefits of using SRA in real construction projects.

Acknowledgements

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 four authors also acknowledge the help received by the research group TEP-955 from the PAIDI (Junta de Andalucía, Spain).

References

[1] M. Vanhoucke, Measuring the efficiency of project control using fictitious and empirical project data, International Journal of Project Management. 30 (2012) 252–263. doi:10.1016/j.ijproman.2011.05.006.

[2] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega. 39 (2011) 416–426. doi:10.1016/j.omega.2010.09.006.

[3] A. Ansar, B. Flyvbjerg, A. Budzier, D. Lunn, Does infrastructure investment lead to economic growth or economic fragility? Evidence from China, Oxford Review of Economic Policy. 32 (2016) 360–390. doi:10.1093/oxrep/grw022.

[4] B. Flyvbjerg, Over Budget, Over Time, Over and Over Again, Oxford University Press, 2011. doi:10.1093/oxfordhb/9780199563142.003.0014.

[5] S.A. Assaf, S. Al-Hejji, Causes of delay in large construction projects, International Journal of Project Management. 24 (2006) 349–357. doi:10.1016/j.ijproman.2005.11.010.

[6] Vanhoucke, On the use of schedule risk analysis for project management, Journal of Modern Project Management. 2 (2015) 108–117. Stable URL: http://hdl.handle.net/1854/LU-8509541.

[7] M. Vanhoucke, Using activity sensitivity and network topology information to monitor project time performance, Omega. 38 (2010) 359–370. doi:10.1016/J.OMEGA.2009.10.001.

[8] P. Ballesteros-Pérez, Modelling the boundaries of project fast-tracking, Automation in Construction. 84 (2017) 231–241. doi:10.1016/j.autcon.2017.09.006.

[9] P. Ballesteros-Pérez, G.D. Larsen, M.C. González-Cruz, Do projects really end late? On the shortcomings of the classical scheduling techniques, Journal of Technology and Science Education. 8 (2018) 86–102. doi:10.3926/jotse.303.

[10] P. Ballesteros-Pérez, E. Sanz-Ablanedo, R. Soetanto, M.C. González-Cruz, G.D. Larsen, On the duration and cost variability of construction activities: an empirical study, Journal of Construction Engineering and Management. *In press* (2019). doi:10.1061/(ASCE)CO.1943-7862.0001739.

[11] Y. Liu, Z.F. Wang, Analysis of Project Shedule Risk Indexes in PERT Network Using Monte Carlo Simulation, in: Advanced Materials Research, Trans Tech Publications, 2013: pp. 2205–2211. doi:10.4028/www.scientific.net/AMR.760-762.2205.

[12] R.M. Van Slyke, Monte Carlo methods and the PERT problem, Operations Research. 11 (1963) 839–860. doi:10.1287/opre.11.5.839.

[13] T.M. Williams, Criticality in Stochastic networks, The Journal of the Operational Research Society. 43 (1992) 353–357. Stable URL: https://www.jstor.org/stable/2583158.

[14] M. Madadi, H. Iranmanesh, A management oriented approach to reduce a project duration and its risk (variability), European Journal of Operational Research. 219 (2012) 751–761. doi:10.1016/j.ejor.2012.01.006.

[15] S.E. Elmaghraby, On criticality and sensitivity in activity networks, European Journal of Operational Research. 127 (2000) 220–238. doi:10.1016/S0377-2217(99)00483-X.

[16] W. Cui, J. Qin, C. Yue, Criticality Measurement in PERT Networks, in: 2006 IEEE International Conference on Systems, Man and Cybernetics, IEEE, 2006: pp. 703–706. doi:10.1109/ICSMC.2006.384468.

[17] J.G. Cho, B.J. Yum, An uncertainty importance measure of activities in PERT networks, International Journal of Production Research. 35 (1997) 2737–2758. doi:10.1080/002075497194426.

[18] E. Koehler, E. Brown, S.J.-P.A. Haneuse, On the Assessment of Monte Carlo Error in Simulation-Based Statistical Analyses, The American Statistician. 63 (2009) 155–162. doi:10.1198/tast.2009.0030.

[19] D. Hulett, Schedule risk analysis simplified, Project Management Network. 10 (1996) 23–30. Stable URL: https://www.pmi.org/learning/library/schedule-risk-analysis-simplified-10573.

[20] M. Vanhoucke, J. Coelho, J. Batselier, An Overview of Project Data Management and Control, The Journal of Modern Project Management. 7 (2016) 6–21. doi:10.3963/JMPM.V3I3.158.

[21] E. Demeulemeester, M. Vanhoucke, W. Herroelen, RanGen: A random network generator for activity-on-the-node networks, Journal of Scheduling. 6 (2003) 17–38. doi:10.1023/A:1022283403119.

[22] M. Vanhoucke, J. Coelho, D. Debels, B. Maenhout, L. V. Tavares, An evaluation of the adequacy of project network generators with systematically sampled networks, European Journal of Operational Research. 187 (2008) 511–524. doi:10.1016/J.EJOR.2007.03.032.

[23] M. Vanhoucke, Measuring Time - Improving Project Performance Using Earned Value Management, Springer, International Series in Operations Research & Management Science, 2010. doi:10.1007/978-1-4419-1014-1.

[24] R. Elshaer, Impact of sensitivity information on the prediction of project's duration using earned schedule method, International Journal of Project Management. 31 (2013) 579–588. doi:10.1016/J.IJPROMAN.2012.10.006.

[25] M. Wauters, M. Vanhoucke, Study of the stability of earned value management forecasting, Journal of Construction Engineering and Management. 141 (2014) 1–10. doi:10.1061/(ASCE)CO.1943-7862.0000947.

[26] J. Colin, M. Vanhoucke, Setting tolerance limits for statistical project control using earned value management, Omega. 49 (2014) 107–122. doi:10.1016/J.OMEGA.2014.06.001.

[27] M. Vanhoucke, Project Management with Dynamic Scheduling, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. doi:10.1007/978-3-642-40438-2.

[28] D. Trietsch, L. Mazmanyan, L. Gevorgyan, K.R. Baker, Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation, European Journal of Operational Research. 216 (2012) 386–396. doi:10.1016/j.ejor.2011.07.054.

[29] J. Colin, M. Vanhoucke, Empirical Perspective on Activity Durations for Project-Management Simulation Studies, Journal of Construction Engineering and Management. 142 (2016) 04015047. doi:10.1061/(ASCE)CO.1943-7862.0001022.

[30] P. Ballesteros-Pérez, M-PERT: Manual Project-Duration Estimation Technique for Teaching Scheduling Basics, Journal of Construction Engineering and Management. 143 (2017). doi:10.1061/(ASCE)CO.1943-7862.0001358.

[31] J. Batselier, M. Vanhoucke, Construction and evaluation framework for a real-life project database, International Journal of Project Management. 33 (2015) 697–710. doi:10.1016/J.IJPROMAN.2014.09.004.

[32] G. Gutierrez, A. Paul, Analysis of the Effects of Uncertainty, Risk-Pooling, and Subcontracting Mechanisms on Project Performance, Operations Research. 48 (2000) 927–938. doi:10.1287/opre.48.6.927.12398.

[33] S.E. Elmaghraby, Y. Fathi, M.R. Taner, On the sensitivity of project variability to activity mean duration, International Journal of Production Economics. 62 (1999) 219–232. doi:10.1016/S0925-5273(98)00241-2.

[34] X. Xu, J. Wang, C.Z. Li, W. Huang, N. Xia, Schedule risk analysis of infrastructure projects: A hybrid dynamic approach, Automation in Construction. 95 (2018) 20–34. doi:10.1016/j.autcon.2018.07.026.

[35] S. Creemers, E. Demeulemeester, S. Van de Vonder, A new approach for quantitative risk analysis, Annals of Operations Research. 213 (2014) 27–65. doi:10.1007/s10479-013-1355-y.

[36] D. Kuchta, Use of fuzzy numbers in project risk (criticality) assessment, International Journal of Project Management. 19 (2001) 305–310. doi:10.1016/S0263-7863(00)00022-3.

[37] R.A. Bowman, Sensitivity curves for effective project management, Naval Research Logistics. 50 (2003) 481–497. doi:10.1002/nav.10064.

[38] R.A. Bowman, Developing activity duration specification limits for effective project control, European Journal of Operational Research. 174 (2006) 1191–1204. doi:10.1016/j.ejor.2005.03.017.

[39] R.A. Bowman, Efficient sensitivity analysis of PERT network performance measures to significant changes in activity time parameters, Journal of the Operational Research Society. 58 (2007) 1354–1360. doi:10.1057/palgrave.jors.2602297.

[40] X. Hu, N. Cui, E. Demeulemeester, L. Bie, Incorporation of activity sensitivity measures into buffer management to manage project schedule risk, European Journal of Operational Research. 249 (2016) 717–727. doi:10.1016/j.ejor.2015.08.066.

[41] A. Martens, M. Vanhoucke, An empirical validation of the performance of project control tolerance limits, Automation in Construction. 89 (2018) 71–85. doi:10.1016/J.AUTCON.2018.01.002.

Metric and source	*Brief description*	*Generic expression*
Criticality Index (***CI***) (Van Slyke, 1963) [12]	The *CI* was the first metric for measuring activity sensitivity. It basically measures the probability of an activity *i* falling in the critical path.	$CI_i = \text{Prob}(i \text{ is critical}) = \text{Prob}(s_i{=}0) = E(s_i = 0)$ Where CI_i is activity *i*'s Criticality Index; s_i is activity *i*'s slack (also known as total float); and $E(\cdot)$ is the expectation (average).
Significance Index (***SI***) (Williams, 1992) [13]	This metric was an attempt to overcome the limitations of the *CI*. The *SI* incorporates an estimate of the potential impact that a delay in activity *i* may cause in the whole project.	$SI_i = E\left(\frac{d_i}{d_i + s_i} \cdot \frac{PD^j}{E(PD)}\right)$ Where d_i is activity *i*'s duration; *PD* is the Project Duration; and PD^j is the Project duration at simulation run *j*.
Cruciality Index based on Pearson product-moment (***CRI(r)***) (Williams, 1992) [13]	This and the next two Cruciality Indices try to evaluate the activity importance by measuring the correlation between the activity duration and the project duration. *CRI(r)* corresponds to the linear correlation version	$CRI(r)_i = \text{correl}(d_i, PD) = \frac{\text{covar}(d_i, PD)}{\sigma_i^2 \cdot \sigma_p^2}$ Where correl(*x*,*y*) denotes the linear correlation between *x* and *y*; covar(*x*,*y*) the covariance between *x* and *y*; σ_i^2 is activity *i*'s duration variance; and σ_p^2 is the Project Duration variance.
Cruciality Index based on Spearman's rank (***CRI(ρ)***) (Williams, 1992) [13]	This metric tries to anticipate the potential non-linearities that the correlation between the activity duration and the project duration may have.	$CRI(\rho)_i = correl(\text{rank } d_i, \text{rank } PD)$ *CRI(ρ)* actually measures the (squared) ranking differences between the activity durations and the project durations. For further mathematical details go to *the supplemental online material.*
Cruciality Index based on Kendall's rank (***CRI(τ)***) (Williams, 1992) [13]	This metric measures the correlation by counting the proportion of concordant and discordant pairs of the same two variables (d_i and *PD*)	$CRI(\tau)_i = \text{Prob}\left\{(d_i^\ell - d_i^j)(PD^\ell - PD^j) > 0\right\} - \text{Prob}\left\{(d_i^\ell - d_i^j)(PD^\ell - PD^j) < 0\right\}$ Where ℓ is an auxiliary index defined as $\ell = j+1, j+2, \ldots N$. For further mathematical details go to *the supplemental online material.*
Schedule Sensitivity Index (***SSI***) (Vanhoucke, 2010a)[7]	Vanhoucke [7] indicated that the Project Management Body of Knowledge (PMBoK) suggested assessing activity sensitivity by multiplying the activity probability of being critical (the *CI*) and its impact (measured by the relative importance of the activity duration variability).	$SSI_i = CI_i \cdot \frac{\sigma_i}{\sigma_p}$ Where σ_i is activity *i*'s duration standard deviation; and σ_p is the Project Duration standard deviation.
Management-Oriented Index (***MOI***) (Madadi and Iranmanesh, 2012)[14]	This metric was the first to combine activity information with topological network information.	$MOI_i = \frac{\sigma_i}{\sigma_{\max}} \cdot \frac{1}{1 + E(s_i) - \frac{n_{successors\ i}}{n}}$ Where σ_{max} is the highest standard deviation among the σ_i values of all activities, that is, σ_{max}=max σ_i with i=1,2,… n; $E(s_i)$ is the expectation (average) of activity *i*'s slack (in all simulation runs) and $n_{successors\ i}$ is the total number of (direct and transitive) successors of activity *i*.
Criticality-Slack-Sensitivity index (***CSS***) (this paper)	This new index is a refinement of the previous *SSI* and *MOI* metrics, and it is proposed in this paper for the first time.	$CSS_i = SSI_i \cdot \frac{E(s_i) - s'_i}{E(PD)} = CI_i \cdot \frac{E(s_i) - s'_i}{E(PD)} \cdot \frac{\sigma_{d_i}}{\sigma_{PD}}$ Where s'_i is activity *i*'s slack in the deterministic schedule, that is, when all activities in the schedule last their planned (baseline) durations. $E(s_i)$, CI_i and SSI_i have been defined above.

Table 1. Summary of all SRA activity-based metrics compared

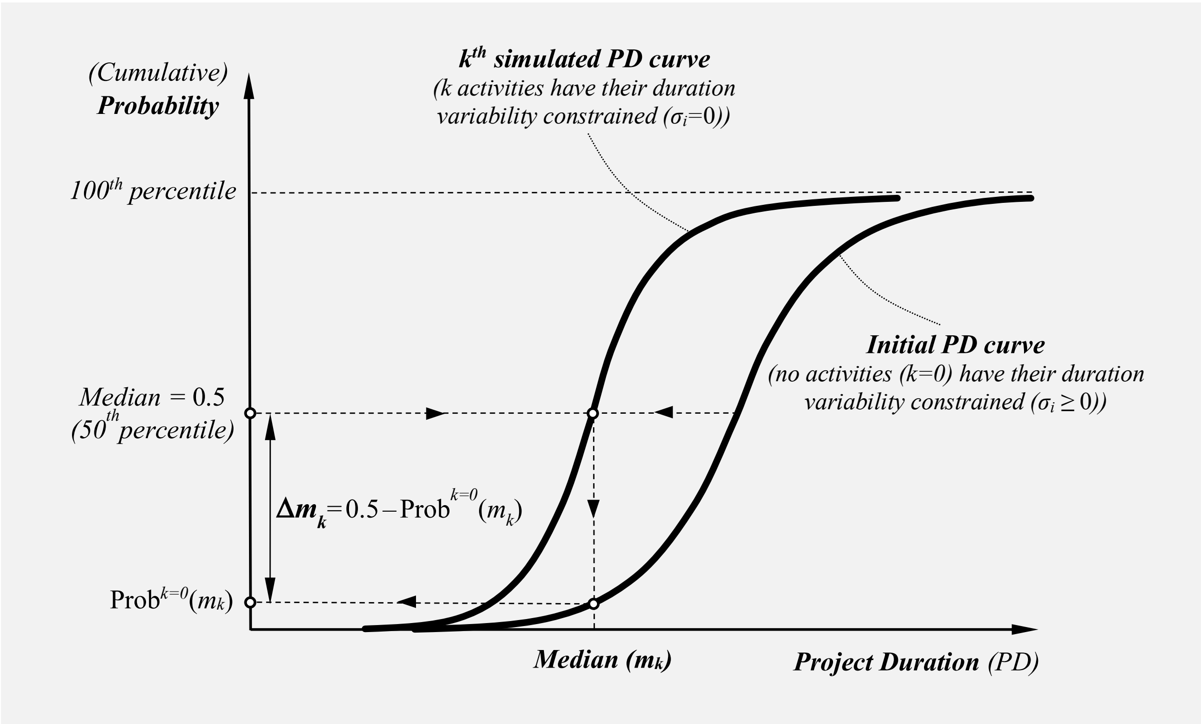

Figure 1. Tracking efficiency measurement approach (Δm_k calculation)

$\Delta \boldsymbol{m}_k$ values

PC (%)	**0%**	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Random	**0.0%**	4.3%	8.7%	13.1%	17.5%	21.8%	25.7%	29.3%	32.4%	34.9%	**35.8%**
CI	**0.0%**	5.6%	10.9%	15.7%	20.4%	24.7%	28.2%	31.3%	33.7%	35.3%	**35.8%**
SI	**0.0%**	5.6%	10.8%	15.8%	20.4%	24.7%	28.2%	31.3%	33.7%	35.3%	**35.8%**
CRI(r)	**0.0%**	5.2%	10.5%	15.4%	20.0%	24.3%	27.9%	31.0%	33.5%	35.2%	**35.8%**
CRI(ρ)	**0.0%**	5.3%	10.5%	15.4%	20.0%	24.3%	27.9%	30.9%	33.4%	35.2%	**35.8%**
CRI(τ)	**0.0%**	6.8%	12.4%	17.5%	22.1%	26.1%	29.4%	32.1%	34.1%	35.5%	**35.8%**
SSI	**0.0%**	5.3%	10.6%	15.6%	20.2%	24.5%	28.1%	31.2%	33.6%	35.3%	**35.8%**
MOI	**0.0%**	5.6%	10.9%	15.8%	20.3%	24.5%	28.0%	31.0%	33.5%	35.2%	**35.8%**
CSS	**0.0%**	**7.2%**	**14.0%**	**19.6%**	**24.5%**	**28.4%**	**31.4%**	**33.6%**	**35.1%**	**35.8%**	**35.8%**

$\Delta \boldsymbol{\sigma}_k$ values

PC (%)	**0%**	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**	**100%**
Random	**0.0%**	4.6%	9.4%	14.7%	20.6%	27.2%	34.7%	43.4%	54.2%	69.2%	**100.0%**
CI	**0.0%**	6.9%	13.9%	20.7%	28.0%	36.9%	44.7%	53.5%	64.3%	76.7%	**100.0%**
SI	**0.0%**	6.9%	13.9%	20.8%	28.1%	36.9%	44.9%	54.0%	64.7%	77.6%	**100.0%**
CRI(r)	**0.0%**	6.2%	13.2%	20.0%	27.2%	35.7%	43.2%	51.8%	62.6%	75.4%	**100.0%**
CRI(ρ)	**0.0%**	6.3%	13.3%	19.9%	27.2%	35.5%	43.0%	51.3%	61.9%	74.6%	**100.0%**
CRI(τ)	**0.0%**	8.7%	16.4%	24.2%	32.6%	42.5%	51.7%	61.3%	71.9%	84.3%	**100.0%**
SSI	**0.0%**	6.5%	13.5%	20.3%	27.5%	36.3%	44.4%	53.2%	64.1%	76.8%	**100.0%**
MOI	**0.0%**	7.0%	14.1%	20.7%	27.9%	36.1%	43.4%	51.6%	62.1%	74.9%	**100.0%**
CSS	**0.0%**	**9.4%**	**19.2%**	**28.1%**	**36.8%**	**46.0%**	**55.0%**	**64.3%**	**74.9%**	**87.4%**	**100.0%**

Figure 2. Project Duration median percentile reduction (Δm_k) and Project Duration standard deviation reduction ($\Delta \sigma_k$) as a function of the Project Control (*PC*) effort for all SRA metrics in *one-off* calculation mode (top performing values highlighted in bold text)

$\Delta \boldsymbol{m}_k$ values

PC (%)	**0%**	**3.3%**	**6.7%**	**10%**	**13.3%**	**16.7%**	**20%**	**23.3%**	**26.7%**	**30%**	**100%**
Random	**0.0%**	1.5%	2.8%	4.3%	5.8%	7.2%	8.7%	10.1%	11.7%	13.1%	**35.8%**
CI	**0.0%**	2.0%	4.7%	7.3%	10.1%	12.8%	15.5%	17.9%	20.0%	22.0%	**35.8%**
SI	**0.0%**	2.0%	4.7%	7.4%	10.0%	12.7%	15.4%	17.7%	19.8%	21.7%	**35.8%**
CRI(r)	**0.0%**	1.7%	6.5%	10.7%	14.3%	17.5%	20.3%	22.7%	24.8%	26.6%	**35.8%**
CRI(ρ)	**0.0%**	1.9%	6.5%	10.6%	14.2%	17.4%	20.3%	22.8%	24.8%	26.6%	**35.8%**
CRI(τ)	**0.0%**	1.4%	3.7%	6.0%	8.0%	10.0%	11.8%	13.6%	15.4%	17.1%	**35.8%**
SSI	**0.0%**	**4.8%**	**8.8%**	**12.5%**	**15.7%**	**18.7%**	**21.3%**	**23.5%**	**25.4%**	**27.1%**	**35.8%**
MOI	**0.0%**	2.0%	5.7%	9.0%	12.2%	15.1%	17.8%	20.2%	22.3%	24.2%	**35.8%**
CSS	**0.0%**	4.1%	7.5%	10.6%	13.3%	15.9%	18.2%	20.2%	21.8%	23.4%	**35.8%**

$\Delta \boldsymbol{\sigma}_k$ values

PC (%)	**0%**	**3.3%**	**6.7%**	**10%**	**13.3%**	**16.7%**	**20%**	**23.3%**	**26.7%**	**30%**	**100%**
Random	**0.0%**	1.4%	2.9%	4.6%	6.2%	7.7%	9.4%	11.2%	12.9%	14.7%	**100.0%**
CI	**0.0%**	2.4%	6.5%	10.7%	14.9%	19.1%	23.7%	27.9%	31.8%	35.8%	**100.0%**
SI	**0.0%**	2.3%	6.6%	10.5%	14.7%	19.0%	23.4%	27.4%	31.0%	34.7%	**100.0%**
CRI(r)	**0.0%**	2.0%	11.7%	19.5%	26.3%	32.3%	37.9%	43.0%	47.7%	52.1%	**100.0%**
CRI(ρ)	**0.0%**	1.9%	11.5%	19.1%	25.6%	31.3%	36.4%	41.1%	45.7%	49.9%	**100.0%**
CRI(τ)	**0.0%**	1.1%	4.4%	7.4%	9.9%	12.5%	15.0%	17.6%	20.4%	23.1%	**100.0%**
SSI	**0.0%**	**9.5%**	**17.0%**	**23.5%**	**29.2%**	**34.4%**	**39.2%**	**43.9%**	**48.3%**	**52.3%**	**100.0%**
MOI	**0.0%**	2.4%	9.3%	15.1%	20.5%	25.6%	30.3%	35.0%	39.4%	43.8%	**100.0%**
CSS	**0.0%**	6.0%	10.8%	15.0%	18.9%	22.3%	25.7%	28.3%	30.9%	33.2%	**100.0%**

Figure 3. Project Duration median percentile reduction (Δm_k) and Project Duration standard deviation reduction ($\Delta \sigma_k$) as a function of the Project Control (*PC*) effort for all SRA metrics in *iterative* calculation mode (top performing values highlighted in bold text)

PC = 10%

SP	**0.07**	**0.17**	**0.28**	**0.38**	**0.48**	**0.59**	**0.69**	**0.79**	**0.90**
Random	4.5%	4.3%	3.9%	4.5%	4.2%	3.9%	4.0%	4.5%	4.2%
CI	7.3%	6.5%	5.0%	5.0%	5.4%	4.6%	4.3%	4.8%	4.4%
SI	7.2%	6.5%	5.1%	4.8%	5.4%	4.9%	4.3%	4.8%	4.7%
CRI(r)	6.1%	5.9%	5.0%	4.5%	5.0%	4.8%	4.5%	4.7%	4.3%
CRI(ρ)	6.5%	5.9%	4.8%	4.6%	5.1%	4.9%	4.6%	4.8%	4.5%
CRI(τ)	9.5%	8.5%	7.2%	**7.1%**	6.3%	5.4%	**5.2%**	**5.1%**	**4.9%**
SSI	6.6%	6.1%	5.0%	4.5%	5.0%	4.9%	4.3%	4.7%	4.6%
MOI	6.5%	6.3%	4.8%	5.5%	5.5%	5.3%	4.9%	4.9%	4.4%
CSS	**11.2%**	**9.7%**	**8.1%**	6.9%	**6.4%**	**6.0%**	**5.2%**	5.0%	4.4%

PC = 20%

SP	**0.07**	**0.17**	**0.28**	**0.38**	**0.48**	**0.59**	**0.69**	**0.79**	**0.90**
Random	9.7%	9.0%	8.4%	7.4%	8.5%	7.8%	7.8%	9.1%	8.9%
CI	13.8%	12.3%	9.5%	9.8%	10.6%	9.7%	9.0%	9.6%	9.0%
SI	13.3%	12.2%	9.1%	9.7%	10.6%	9.5%	8.8%	9.6%	9.2%
CRI(r)	12.1%	11.9%	8.9%	8.9%	10.2%	9.5%	9.1%	9.5%	8.9%
CRI(ρ)	12.6%	12.1%	8.9%	9.2%	10.2%	9.2%	9.0%	9.5%	8.8%
CRI(τ)	17.2%	14.6%	12.4%	11.7%	11.8%	11.3%	10.0%	**10.1%**	**9.6%**
SSI	12.6%	12.1%	9.1%	9.3%	10.2%	9.3%	8.9%	9.6%	9.0%
MOI	12.4%	12.2%	9.1%	9.9%	10.7%	10.4%	9.1%	9.8%	9.5%
CSS	**19.6%**	**18.2%**	**15.9%**	**14.2%**	**12.7%**	**11.5%**	**10.7%**	10.0%	**9.6%**

PC = 30%

SP	**0.07**	**0.17**	**0.28**	**0.38**	**0.48**	**0.59**	**0.69**	**0.79**	**0.90**
Random	15.2%	13.5%	12.3%	11.6%	12.6%	11.5%	12.3%	13.4%	13.5%
CI	20.5%	17.2%	13.8%	13.8%	15.7%	14.3%	13.3%	14.2%	13.9%
SI	20.4%	17.4%	13.8%	13.7%	15.6%	14.2%	13.7%	14.3%	13.6%
CRI(r)	18.8%	16.9%	14.0%	13.4%	15.3%	13.6%	13.3%	14.2%	13.2%
CRI(ρ)	19.3%	16.8%	13.8%	13.4%	15.3%	13.5%	13.3%	14.1%	13.4%
CRI(τ)	24.0%	19.9%	16.5%	15.8%	17.1%	16.0%	14.8%	**14.9%**	14.0%
SSI	20.1%	17.1%	14.1%	13.4%	15.4%	14.0%	13.2%	14.3%	13.0%
MOI	18.8%	17.1%	14.1%	13.7%	15.8%	14.4%	13.7%	14.6%	14.1%
CSS	**27.0%**	**23.9%**	**22.4%**	**20.4%**	**18.7%**	**16.5%**	**16.0%**	**14.9%**	**14.5%**

Figure 4. Project Duration median percentile reduction (Δm_k) values as a function of the Serial-Parallel (SP) indicator and three Project Control (*PC*) levels (10, 20 and 30%) for all SRA metrics in *one-off* calculation mode (top performing values highlighted in bold text)

PC = 10%

SP	**0.07**	**0.17**	**0.28**	**0.38**	**0.48**	**0.59**	**0.69**	**0.79**	**0.90**
Random	4.5%	4.3%	3.9%	4.5%	4.2%	3.9%	4.0%	4.5%	4.2%
CI	13.3%	10.1%	7.9%	6.5%	6.2%	5.8%	5.2%	5.1%	4.8%
SI	13.1%	10.1%	7.5%	7.2%	6.4%	6.0%	5.4%	5.0%	4.9%
CRI(r)	15.5%	13.3%	11.1%	10.2%	9.8%	8.9%	8.5%	8.3%	8.0%
CRI(ρ)	15.9%	13.1%	11.3%	9.8%	9.6%	8.8%	8.9%	8.5%	8.4%
CRI(τ)	7.9%	7.5%	6.2%	6.0%	5.4%	5.0%	4.5%	4.9%	4.7%
SSI	**19.6%**	**15.5%**	**13.5%**	**12.2%**	**11.2%**	**10.6%**	**10.4%**	**9.8%**	**9.7%**
MOI	16.2%	12.2%	9.3%	8.3%	7.8%	6.8%	6.6%	6.4%	6.0%
CSS	19.2%	15.1%	11.9%	9.9%	9.1%	7.5%	6.2%	6.6%	5.6%

PC = 20%

SP	**0.07**	**0.17**	**0.28**	**0.38**	**0.48**	**0.59**	**0.69**	**0.79**	**0.90**
Random	9.7%	9.0%	8.4%	7.4%	8.5%	7.8%	7.8%	9.1%	8.9%
CI	27.6%	22.1%	16.9%	13.9%	12.8%	11.7%	10.3%	10.1%	9.4%
SI	27.3%	21.8%	16.9%	14.3%	12.9%	11.7%	10.6%	10.2%	9.5%
CRI(r)	28.2%	24.4%	21.3%	19.7%	19.1%	17.4%	16.4%	16.5%	15.6%
CRI(ρ)	30.5%	24.9%	21.0%	19.4%	18.5%	17.1%	17.0%	16.3%	15.8%
CRI(τ)	15.6%	14.0%	11.8%	11.6%	11.2%	10.1%	9.8%	9.9%	9.6%
SSI	**31.6%**	**26.2%**	**21.8%**	**20.5%**	**19.4%**	**18.0%**	**17.7%**	**17.2%**	**17.2%**
MOI	30.4%	24.2%	19.2%	16.7%	15.1%	13.9%	13.4%	12.7%	11.7%
CSS	31.4%	25.6%	20.9%	17.5%	15.7%	12.7%	11.3%	11.7%	10.2%

PC = 30%

SP	**0.07**	**0.17**	**0.28**	**0.38**	**0.48**	**0.59**	**0.69**	**0.79**	**0.90**
Random	15.2%	13.5%	12.3%	11.6%	12.6%	11.5%	12.3%	13.4%	13.5%
CI	37.3%	29.5%	25.6%	21.4%	19.3%	17.4%	16.3%	15.0%	14.3%
SI	36.6%	28.7%	25.7%	21.2%	19.2%	17.4%	16.2%	15.1%	14.2%
CRI(r)	36.5%	30.5%	27.0%	25.5%	**25.5%**	23.7%	22.7%	22.8%	21.6%
CRI(ρ)	38.8%	31.2%	27.6%	25.5%	24.9%	23.3%	22.9%	22.6%	21.6%
CRI(τ)	23.2%	19.5%	16.1%	15.6%	16.6%	15.5%	14.4%	14.7%	14.1%
SSI	**39.6%**	**31.6%**	**28.0%**	**26.0%**	25.2%	**23.9%**	**23.6%**	**23.0%**	**22.9%**
MOI	38.8%	30.6%	26.9%	23.6%	21.6%	19.5%	19.4%	18.6%	17.7%
CSS	39.1%	30.8%	27.1%	23.4%	21.2%	17.4%	15.3%	16.2%	14.4%

Figure 5. Project Duration median percentile reduction (Δm_k) values as a function of the Serial-Parallel (SP) indicator and three Project Control (*PC*) levels (10, 20 and 30%) for all SRA metrics in *iterative* calculation mode (top performing values highlighted in bold text)

Supplemental Online material

4100-project simulation detailed results

All project simulation results discussed in the paper can be accessed here: http://bit.ly/2JPHhnm . This link corresponds to a 33 MB MS Excel file. Please, be patient when downloading and opening it.

Additionally, there is also some additional simulation data comparing the same project networks under the assumption of low, medium and high activity duration variability. These results can be accessed here: http://bit.ly/2uAUERz . This link corresponds to a 118 MB Zip file containing multiple MS Excel spreadsheets. Please, be patient when downloading and opening it.

Abbreviations list

AD	Activity Distribution (topological) indicator.
CI	Criticality index
CI_i	Activity *i*'s Criticality Index
correl(x,y)	Linear correlation between x and y
covar(x,y)	Covariance between x and y.
CRI(r)	Cruciality Index based on Pearson product-moment.
CRI(ρ)	Cruciality Index based on Spearman's rank.
CRI(τ)	Cruciality Index based on Kendall's rank.
CSS	Criticality-Slack-Sensitivity index.
Δm_k	Project Duration median percentile reduction respect to the initial PD curve when k activities have their duration variability constrained (σ_i=0).
$\Delta\sigma_k$	Project Duration standard deviation reduction respect to the initial PD curve when k activities have their duration variability constrained (σ_i=0).

δ_i^j	Ranking difference between d_i^j and PD^j at simulation run j, that is $\delta_i^j = rank\left(d_i^j\right) - rank\left(PD^j\right)$
d_i	Activity i's duration.
d_i^j	Activity i's duration at simulation run j.
$E(\cdot)$	Expectation (average) of $(\cdot)$.
i	Activity (in a given schedule network) identifier index.
j	Monte Carlo simulation run identifier index.
k	Total number of activities whose duration variability (σ_i) has been constrained (forced to σ_i=0) for project control purposes.
LA	Length of Arcs (topological) indicator.
ℓ	Auxiliary index defined as $\ell = j+1, j+2, \ldots N$.
μ_i	Activity i's duration average (planned duration).
m_k	Project duration median when k activities have their duration variability constrained.
MOI	Management-Oriented index.
N	Total number of Monte Carlo simulations performed in a construction schedule.
n	Total number of activities in a construction schedule.
Normal$(\cdot)$	Normal probability distribution.
$n_{successors\ i}$	Total number of (direct and transitive) successors of activity i.
PC	Level of Project Control (expressed as k/n, that is, % of activities with constrained duration variability respect to the total number of activities)
PD	Project Duration.
PD^j	Project duration at simulation run j.
$\text{Prob}^k(\cdot)$	Probability in density curve k of $(\cdot)$.
σ_i^2	Activity i's duration variance
σ_p^2	Project Duration variance.
σ_i	Activity i's duration standard deviation
σ_{max}	Highest standard deviation among the σ_i values of all scheduled activities in a project, that is σ_{max}=max σ_i with i=1,2,… n
σ_p	Project Duration standard deviation.
s_i	Activity i's slack (also known as total float)

s_i^j Activity i's slack at simulation run j.

s'_i Activity i's slack (total float) when all activities in the schedule last their avg. durations.

SI Significance index.

SP Serial-Parallel (topological) indicator. It measures how close a network resembles a perfectly parallel network (SP=0) or a series network (SP=1)

SSI Schedule Sensitivity index.

TF Topological Float (topological) indicator.

Appendix B: Activity sensitivity metrics simulation-based estimators

When resorting to Monte Carlo simulation, expressions in Table 1 on the paper can be computed using the following simulation-based estimators. All abbreviations and variables can be found in the previous *abbreviations list*.

$$CI_i = \frac{\sum_{j=1}^{N} \mathbf{1}\left(s_i^j = 0\right)}{N} \quad \text{with } \mathbf{1}(\cdot)=1 \text{ if } s_i^j = 0 \text{ and } \mathbf{1}(\cdot)=0 \text{ if } s_i^j > 0 \tag{S1}$$

$$\widehat{SI}_i = \frac{1}{N}\sum_{j=1}^{N}\left(\frac{d_i^j}{d_i^j + s_i^j} \cdot \frac{PD^j}{\frac{1}{N}\sum_{j=1}^{N} PD^j}\right) \tag{S2}$$

$$CRI(r)_i = \frac{\sum_{j=1}^{N}\left(d_i^j - \frac{1}{N}\sum_{j=1}^{N} d_i^j\right)\left(PD^j - \frac{1}{N}\sum_{j=1}^{N} PD^j\right)}{\sqrt{\sum_{j=1}^{N}\left(d_i^j - \frac{1}{N}\sum_{j=1}^{N} d_i^j\right)^2}\sqrt{\sum_{j=1}^{N}\left(PD^j - \frac{1}{N}\sum_{j=1}^{N} PD^j\right)^2}} \tag{S3}$$

$$CRI(\rho)_i = 1 - \frac{6\sum_{j=1}^{N}\left(\delta_i^j\right)^2}{N\left(N^2 - 1\right)} \tag{S4}$$

$$CRI(\tau)_i = \left[\frac{4\sum_{j=1}^{N-1}\sum_{\ell=j+1}^{N} \mathbf{1}\left\{\left(d_i^\ell - d_i^j\right)\left(PD^\ell - PD^j\right) > 0\right\}}{N\left(N-1\right)}\right] \text{-1} \tag{S5}$$

$$SSI_i = CI_i \cdot \sqrt{\frac{\sum_{j=1}^{N}\left(d_i^j - \frac{1}{N}\sum_{j=1}^{N} d_i^j\right)^2}{\sum_{j=1}^{N}\left(PD^j - \frac{1}{N}\sum_{j=1}^{N} PD^j\right)^2}} \tag{S6}$$

$$MOI_i = \sqrt{\frac{\sum_{j=1}^{N}\left(d_i^j - \frac{1}{N}\sum_{j=1}^{N} d_i^j\right)^2}{\max\limits_{i=1,2,\dots n}\left\{\sum_{j=1}^{N}\left(d_i^j - \frac{1}{N}\sum_{j=1}^{N} d_i^j\right)^2\right\}}} \cdot \frac{1}{1 + \frac{1}{N}\sum_{j=1}^{N} s_i^j - \frac{n_{sucessors\ i}}{n}} \tag{S7}$$

$$CSS_i = SSI_i \cdot \frac{\frac{1}{N}\sum_{j=1}^{N} s_i^j - s'_i}{\frac{1}{N}\sum_{j=1}^{N} PD^j} \tag{S8}$$