

Disregarding Duration Uncertainty in Partial Order Schedules? Yes, we can!

Alessio Bonfietti¹, Michele Lombardi¹, and Michela Milano¹

DISI, University of Bologna, alessio.bonfietti@unibo.it,
michele.lombardi2@unibo.it, michela.milano@unibo.it

Abstract. In the context of Scheduling under uncertainty, Partial Order Schedules (POS) provide a convenient way to build flexible solutions. A POS is obtained from a Project Graph by adding precedence constraints so that no resource conflict can arise, for any possible assignment of the activity durations. In this paper, we use a simulation approach to evaluate the expected makespan of a number of POSs, obtained by solving scheduling benchmarks via multiple approaches. Our evaluation leads us to the discovery of a striking correlation between the expected makespan and the makespan obtained by simply fixing all durations to their average. The strength of the correlation is such that it is possible to disregard completely the uncertainty during the schedule construction and yet obtain a very accurate estimation of the expected makespan. We provide a thorough empirical and theoretical analysis of this result, showing the existence of solid ground for finding a similarly strong relation on a broad class of scheduling problems of practical importance.

1 Introduction

Combinatorial Optimization approaches have a tremendous potential to improve decision making activities. Most optimization techniques, however, require complete problem knowledge to be available, and with good reason. Indeed, the dynamic and uncertain nature of real world problems is a true curse to deal with. With the exception of specific settings, the scalability of stochastic optimization approaches is orders of magnitude worse than their deterministic counterparts.

Scheduling problems make no exception to this rule and their stochastic variants tend to be even harder to solve than the (already hard) classical formulations. In the literature, the case of uncertain activity durations [2] has received the greatest attention. This is because it provides a convenient framework to model a number of unexpected events (delays, failure of unary resources, unknown input...) and because there is hardly any practical scheduling problem where the durations are really deterministic. Many have tackled this class of problems by shifting the activity start times [17], or by doing that *and* re-adjusting the schedule when a variation occurs [16].

A somehow orthogonal family of approaches has focused instead in providing flexible solutions as Partial Order Schedules (POSs). A POS is an augmentation of the original Project Graph, where a number of precedence constraints has

been added to prevent the occurrence of resource conflicts, whatever the activity durations are. A POS can be obtained through a variety of methods (the reader may refer for details to [7, 4, 5, 10–13]). A POS can be designed to optimize some probabilistic performance metric, such as the expected makespan (a frequent pick) or the n -th quantile (see [3]). Unfortunately, even simply computing such metrics is by itself a difficult problem. Hence, a more convenient approach is to optimize a deterministic scenario (e.g. worst case durations) and rely on the flexibility of the POS for a good performance once the uncertain comes into play.

In this paper, we tackle the rather unusual problem of characterizing the run-time behavior of a POS. Other authors have evaluated the quality of a POS via intuition-based metrics [12, 13, 1], or even via simulation [14]. However, those efforts have been focused in comparing solution approaches. Conversely, *we focus on inferring general properties of the expected makespan of a POS*. The raw data for our investigation is obtained by simulating a large number of POSs, obtained by solving classical scheduling benchmarks via multiple methods. Our main, rather disconcerting, outcome is that in a large variety of situations there is a very strong correlation between the expected makespan and the makespan obtained when all the activities take their expected duration. In particular, their difference appears very resilient to scheduling decisions. *In such a situation it is possible to disregard the uncertainty altogether, solve a deterministic problem, and end up with a close-to-optimal solution for a (much harder!) stochastic problem.* We provide a reasonable explanation for the observed behavior, supported by a mathematical model that we check against the empirical data. We deduce that the behavior we observed, although not universally guaranteed, can be reasonably expected on a broad class of practical problems.

2 Experimental Setup

Origin of our POSs: The POSs for our analysis represent solutions of classical scheduling benchmarks. They have been obtained via two methods: 1) the constructive approach from [11] and 2) the application of a chaining step inspired by [12] to solutions obtained via ILOG CP Optimizer. For each considered instance, we kept all the solutions found by both solvers in the optimization process. In the following, whenever non-explicitly mentioned, the presented data refers to the chaining approach. As target benchmarks we have employed all the RCPSP instances in the PSPLIB [9] from the j30, j60, and j90 sets (named after the number of activities), plus most of the job shop scheduling instances by Taillard [15] (up to 30 jobs and 20 machines). The PSPLIB instances cover a wide range of resource usage patterns, while the job shop instances provide data for a radically different scheduling problem.

The Simulation Approach: In the simplest terms, our simulator is just a composition of Monte-Carlo sampling and critical path computation. Formally, we model the duration of each activity a_i as an *independent* random variable δ_i ranging in the interval $[d_i, D_i]$. The value D_i is the maximum duration and it is

always equal to the (fixed) duration value from the benchmark instances. The minimum duration d_i is computed as $\alpha \cdot D_i$, where $\alpha \in]0, 1[$. The value of α and the probability distribution for the δ_i variables are simulation parameters. The makespan is formally defined as a function $\tau(\delta)$ of the duration variables. We use the special notation T to refer to the value $\tau(D)$, i.e. the makespan for the worst case durations.

The makespan function has unknown probability distribution, and therefore unknown expected value $\mathbb{E}[\tau]$. Our simulator builds an approximation for $\mathbb{E}[\tau]$ by sampling the duration variables to obtain a set of scenarios Δ . Each scenario $\delta^{(k)} \in \Delta$ is an assignment of values to the δ_i variables. Then for each $\delta^{(k)}$ we compute the makespan by running a simple critical path algorithm: since the POS are resource-feasible, there is no need to take into account the capacity limits. By doing so, we obtain the sample average:

$$\bar{\tau} = \frac{1}{|\Delta|} \sum_{\delta^{(k)} \in \Delta} \tau(\delta^{(k)}) \quad (1)$$

which is an approximation for the expected value $\mathbb{E}[\tau]$. We simulated each of our POSs 10,000 times (i.e. $|\Delta| = 10,000$). We have considered two types of distribution for the δ_i variables, namely 1) a uniform distribution and 2) a discrete distribution where the duration can be either αD_i with probability p or D_i with probability $1 - p$. We have run tests with $p = 0.5$ and $p = 0.9$ and $\alpha = 0.1, 0.25, 0.5$ and 0.75 . In the following, whenever non explicitly mentioned, the presented data refers to $\alpha = 0.5$ and uniformly distributed durations.

3 Our Main Result

As mentioned in Section 1, it is common to build a POS to minimize the worst case makespan, so as to provide worst case guarantees. Even in this setting, however, having a good expected performance is a highly desirable property. We therefore decided to start our investigation by comparing the expected makespan of a POS (or rather the approximation $\bar{\tau}$), with its worst case value T .

This comparison can be done effectively by plotting the two quantities one against each other (say, T on the x- and $\bar{\tau}$ on the y-axis). It is clear that $\bar{\tau} \leq T$. That said, given how complex the structure of a POS can be and the amount of variability involved, one could expect the plot to contain a cloud of points distributed almost uniformly below the main bisector. A significant discrepancy between different benchmarks and simulation settings could also be expected.

Surprisingly, we observe instead that in *all the considered cases there exists a strong, seemingly linear, correlation between the average makespan and the worst case one*. Figure 1 shows the described plots for 4 different configurations, chosen as examples: (a) the POSs for the j30 instances from the PSPLIB, solved with the chaining approach and uniform distributions; (b) the j30 set with the constructive approach and uniform distributions; (c) the j60 set with the constructive approach, discrete distribution and $p = 0.9$. Finally, (d) the POSs from the Taillard instances with the chaining approach and uniform distributions.

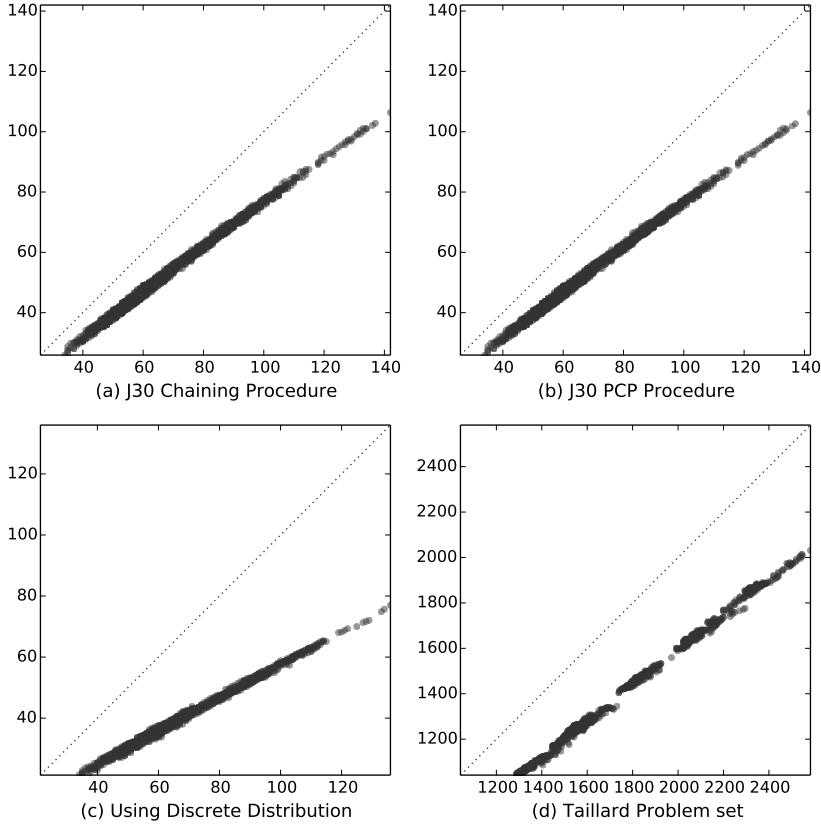


Fig. 1. Expected makespan (y-axis) over worst case makespan (x-axes) for various experimental settings

The Real Correlation: A necessary first step to understand the observed correlation is to identify exactly its terms. In our experimentation, we are assuming all activities to have uniformly distributed durations in the range $[d_i, D_i]$, so that their expected value is $\mathbb{E}[\delta_i] = 0.5 \cdot (d_i + D_i)$. Moreover, since we are also assuming that $d_i = \alpha D_i$, we have that $\mathbb{E}[\delta_i] = \frac{1}{2}(1 + \alpha)D_i$, i.e. the expected durations are proportional to the worst case ones. This means that by fixing all the activities to $\mathbb{E}[\delta_i]$ rather than to D_i , the critical path stays unchanged and the makespan becomes:

$$\tau(\mathbb{E}[\delta]) = \frac{1}{2} \sum_{i \in \pi} (\alpha D_i + D_i) = \frac{1 + \alpha}{2} \sum_{i \in \pi} D_i = \frac{1 + \alpha}{2} T \quad (2)$$

Hence, the observed correlation between $\bar{\tau}$ and T could be just a side effect of a actual correlation between $\bar{\tau}$ and $\tau(\mathbb{E}[\delta])$, because T and $\tau(\mathbb{E}[\delta])$ are *proportional*. To test if this is the case, we have designed a second set of simulations where,

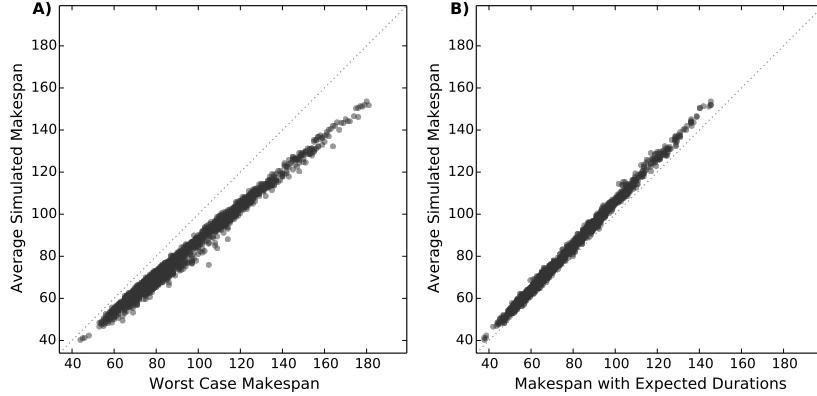


Fig. 2. Correlation of the simulated makespan with α varying on a per-task basis

for each task, we choose at random between $\alpha = 0.1$ or $\alpha = 0.75$. By doing so, Equation (2) does no longer hold. As a consequence, we expect the correlation between $\bar{\tau}$ and T to become more blurry. Figure 2A and 2B report, for the j60 benchmarks, scatter plots having $\bar{\tau}$ on the y-axis. On the x-axis the plots respectively have the worst case makespan and the makespan with expected durations. The data points are much more neatly distributed for Figure 2B. In summary, **there is evidence for the existence of a very strong correlation between the expected makespan $\mathbb{E}[\tau]$ and the makespan $\tau(\mathbb{E}[\delta])$.**

In the (very uncommon) case that the POS consists of a single path, this is a trivial result. In such a situation the makespan function τ can be expressed as:

$$\tau(\delta) = l(\pi) = \sum_{a_i \in \pi} \delta_i \quad (3)$$

where π is the path (a sequence of activity indices $\pi(0), \pi(1), \dots, \pi(k-1)$) and $l(\pi)$ is the path length. For the linear properties of the expected value, it follows that $\mathbb{E}[\tau] = \sum_{i \in \pi} \mathbb{E}[\delta_i]$, i.e. the expected makespan is exactly $\tau(\mathbb{E}[\delta])$. On the other hand, a *general* POS has a much more complex structure, with multiple paths competing to be critical in a stochastic setting. In such case, the presence of the observed correlation is therefore far from an obvious conclusion.

Makespan Deviation: What is even more interesting, however, is that $\bar{\tau}$ and $\tau(\mathbb{E}[\delta])$ remain remarkably close one to each other, rather than diverging as their values grow. It is therefore worth investigating in deeper detail the behavior of the difference $\bar{\tau} - \tau(\mathbb{E}[\delta])$, as an approximation for $\mathbb{E}[\tau] - \tau(\mathbb{E}[\delta])$. We refer to this quantity as Δ and we call it *makespan deviation*.

Figure 3A shows the deviation for j60, over the makespan with expected durations. The plot (and every plot from now on) is obtained using a unique α value for a whole benchmark, because this allows to conveniently compute

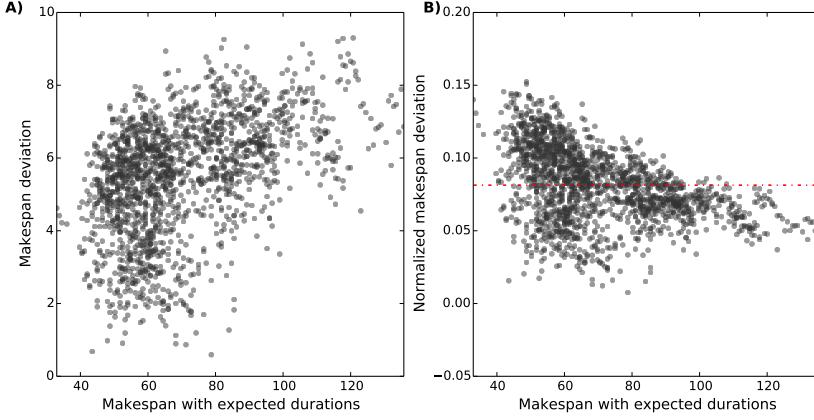


Fig. 3. Deviation and normalized deviation over $\tau(\mathbb{E}[\delta])$

$\tau(\mathbb{E}[\delta])$ as $\frac{1+\alpha}{2}T$. The deviation appears to grow slowly as the makespan (with expected durations) increases.

It is possible to determine the rate of growth more precisely via a test that involves the normalized makespan deviation, i.e. the quantity $\Delta/\tau(\mathbb{E}[\delta])$. Specifically, let us assume that a linear dependence exists, then we should have:

$$\mathbb{E}[\tau] = a \cdot \tau(\mathbb{E}[\delta]) + b \quad (4)$$

Now, note that $\tau(\mathbb{E}[\delta]) = 0$ can hold iff all the activities have zero expected duration. This is only possible if they have zero maximum duration, too. Hence, if $\tau(\mathbb{E}[\delta]) = 0$, then $\mathbb{E}[\tau] = 0$. This implies that $b = 0$, i.e. there is no offset. Hence, we can deduce that $\mathbb{E}[\tau]$ and $\tau(\mathbb{E}[\delta])$ are linearly dependent iff:

$$\frac{\Delta}{\tau(\mathbb{E}[\delta])} = a - 1 \quad (5)$$

i.e. if the normalized makespan deviation is constant. Figure 3B shows for j60 the normalized Δ over $\tau(\mathbb{E}[\delta])$. We have added to the plot a line that shows what the data trend would be in case of a constant normalized deviation. As one can see, the normalized Δ tends instead to get lower as $\tau(\mathbb{E}[\delta])$ grows.

Therefore, the makespan deviation $\mathbb{E}[\tau] - \tau(\mathbb{E}[\delta])$ appears to grow **sub-linearly** with the makespan obtained when all activities take their expected duration, i.e. $\tau(\mathbb{E}[\delta])$. From another perspective, this means that the deviation has sub-linear sensitivity to scheduling decisions. In such a situation, it becomes possible to disregard the duration uncertainty and optimize under the assumption that each activity a_i has duration $\mathbb{E}[\delta_i]$, being reasonably sure that the resulting POS will be close to optimal also in the stochastic setting. This makes for a tremendous reduction in the complexity of scheduling under duration uncertainty, with far reaching consequences both on the theoretical and the practical side.

Given the importance of such a result, it is crucial to reach an understanding of the underlying mechanism, so as to assess to which extent this behavior depends on our experimental setting and to which extent it is characteristic of scheduling problems in general. In the next section, we develop mathematical tools for this analysis.

4 Our Analysis Framework

Independent Path Equivalent Model: We already know from Section 3 that for a single path π the expected makespan is equal to the sum of the expected durations and hence $\Delta = 0$. This is however a rather specific case. In general, it is possible to see a POS as a collection Π of paths π . From this perspective, the makespan can be formulated as:

$$\tau(\delta) = \max_{\pi \in \Pi} l_\pi(\delta) \quad (6)$$

i.e. for a given instantiation of the activity durations, the makespan is given by the longest path. The worst case makespan T will correspond to the worst case length of one or more particular paths: we refer to them as *critical paths*.

In general, many paths will not be critical. Moreover, several paths will likely share some activities. Finally, the number of paths $|\Pi|$ may be exponential in the Project Graph size. For those reasons, computing $\mathbb{E}[\tau]$ for the model from Equation (6) is not viable in practice. Hence, to understand the impact of multiple paths on $\mathbb{E}[\tau]$ we resort to a simplified model. Specifically, we assume that a POS consists of n identically distributed, independent, critical paths. Formally:

$$\tau(\delta) = \max_{i=0..n-1} \lambda_i(\delta) \quad (7)$$

where each λ_i is a random variable representing the length of a path i and ranging in $[\alpha T, T]$. The notation is (on purpose) analogous to that of our simulation. We refer to Equation (7) as the *Independent Path Equivalent Model* (IPE Model).

Effect of Multiple Paths: Now, let F_λ denote the (identical) Cumulative Distribution Function (CDF) for all λ_i , i.e. $F_\lambda(x)$ is the probability that a λ_i variable is less than or equal to x . Then for the CDF of τ we have:

$$F_\tau(x) = \prod_{i=0}^{n-1} F_\lambda(x) = F_\lambda(x)^n \quad (8)$$

in other words, for τ to be lower than a value x , all the path lengths must be lower than x . Therefore, **increasing the number of critical paths n reduces the likelihood to have small makespan values and leads to a larger $\mathbb{E}[\tau]$** . Figure 4A shows this behavior, under the assumption that all λ_i are uniformly distributed with $T = 1$ and $\alpha = 0$ and hence have an average length of 0.5. The drawing reports the Probability Density Function (PDF), i.e.

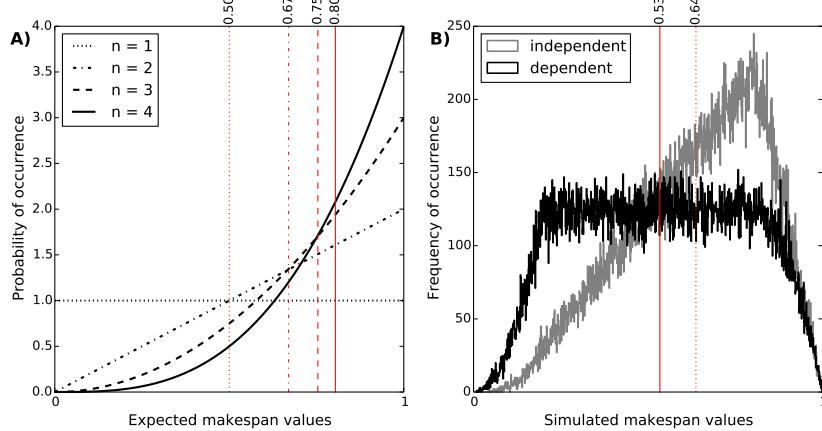


Fig. 4. Effect of multiple paths

the probability that $\tau = x$, with $n = 1, 2, 3, 4$. As n grows, the PDF is skewed to the right and the corresponding expected value (marked by a vertical red line) moves accordingly. Note that moving from a single path to multiple ones causes a relevant shift of $\mathbb{E}[\tau]$ from the average critical path length (i.e. 0.5). This means that the POS structure may in theory have a strong impact on the makespan deviation. However, this is not observed in Figure 1 and Figure 2, raising even more interest in the reasons for this behavior.

4.1 Worst Case Assumptions

The IPE Model differs from the exact formulation in Equation (6) by three main simplifications: 1) all paths are critical, 2) all paths are identically distributed, and 3) all paths are independent. Due to those simplifications, **the IPE Model actually corresponds to a worst case situation and is therefore well suited to obtain conservative estimates.**

Non-critical Paths: An actual POS may contain a number of non-critical paths. In the IPE Model, a non-critical path is modeled by introducing a random variable λ_i with a maximum value strictly lower than T . Any such variable can be replaced by one with the same (scaled) distribution and maximum T , for an increased expected makespan.

Non-identical Distributions: In general, each critical path λ_i will have a different probability distribution F_{λ_i} . However, in Appendix A, it is proved that it is possible to obtain a worst case estimate by assuming all F_{λ_i} to be equal to the one having the lowest value for the integral $\int_0^\infty F_{\lambda_i}(x)dx$.

Path Dependences: Path dependences arise in Equation (6) due to the presence of shared variables. Consider the case of a graph with two paths π_i and π_j sharing a subpath. For each assignment of durations $\delta^{(k)}$, the length of the common subpath will add to that of both π_i and π_j , making them more similar. This can be observed in Figure 4B that reports the empirical distributions of two POSs, both consisting of two paths with two activities each. In the POS corresponding to the grey distribution, the two paths are independent, hence $l(\pi_i) = \delta_h + \delta_k$ and $l(\pi_j) = \delta_l + \delta_m$. In the POS corresponding to the black distribution, the two paths share a variable, i.e. $l(\pi_i) = \delta_h + \delta_k$ and $l(\pi_j) = \delta_h + \delta_m$. All the durations are uniformly distributed, δ_h and δ_l range in $[0, 0.8]$, while δ_j, δ_m range in $[0, 0.2]$. The distribution for the dependent setting is much closer to that of a single path (see Figure 4A) and has a lower average than the independent case.

4.2 Asymptotic Behavior of the Expected Makespan

We are now in a position to use the IPE Model to obtain a conservative estimate of $\mathbb{E}[\tau]$. In particular, we will show that **the makespan deviation Δ for the IPE Model is bounded above by a quantity that depends 1) on the variability of the activities on the paths and 2) on the number of paths**. For proving this, we will consider each critical path to be a sequence of m activities. This can be captured by assuming that $\lambda = \sum_{j=0}^{m-1} \rho_j$, where each ρ_j is an independent random variable with expected value μ_{ρ_j} and relative variability β_{ρ_j} . From this definition, it follows:

$$\sum_{j=0}^{m-1} \mu_{\rho_j} = \mathbb{E}[\lambda] \quad \text{and} \quad \sum_{j=0}^{m-1} \beta_{\rho_j} = (1 - \alpha)T \quad (9)$$

Note that with $m = 1$ we obtain the original IPE Model. Now, let Z be a random variable equal to $\tau - \mu_\lambda$, where $\mu_\lambda = \mathbb{E}[\lambda]$. The Z variable is designed so that in the model $\mathbb{E}[Z]$ corresponds to the makespan deviation Δ . Now, from Jensen's inequality [8], we know that:

$$e^{\mathbb{E}[tZ]} \leq \mathbb{E}[e^{tZ}] \quad (10)$$

because the exponential function e^{tx} is convex in x . The term t is a real valued parameter that will allow some simplifications later on. From the definition of Z and since the exponential is order preserving, we obtain:

$$\mathbb{E}[e^{tZ}] = \mathbb{E}\left[e^{t \max_i(\lambda - \mu_\lambda)}\right] = \mathbb{E}\left[\max_{i=0..n-1} e^{t(\lambda - \mu_\lambda)}\right] \quad (11)$$

By simple arithmetic properties and by linearity of the expectation $\mathbb{E}[]$:

$$\mathbb{E}\left[\max_{i=0..n-1} e^{t(\lambda - \mu_\lambda)}\right] \leq \sum_{i=0}^{n-1} \mathbb{E}\left[e^{t(\lambda - \mu_\lambda)}\right] = n \mathbb{E}\left[e^{t(\lambda - \mu_\lambda)}\right] \quad (12)$$

Note that using a sum to over-approximate a maximum is likely to lead to a loose bound. This is however mitigated by the use of a fast-growing function like the exponential. Now, since $\lambda_i = \sum_j \rho_j$, we can write:

$$n \mathbb{E} [e^{t(\lambda - \mu_\lambda)}] = n \mathbb{E} [e^{t \sum_j (\rho_j - \mu_{\rho_j})}] = n \mathbb{E} \left[\prod_{j=0}^{m-1} e^{t(\rho_j - \mu_{\rho_j})} \right] = n \prod_{j=0}^{m-1} \mathbb{E} [e^{t(\rho_j - \mu_{\rho_j})}]$$

because the ρ_j variables are independent. Now, each term $\rho_j - \mu_{\rho_j}$ has zero mean and spans by definition in a finite range with size β_{ρ_j} . Therefore we can apply Hoeffding's lemma [6] to obtain:

$$n \prod_{j=0}^{m-1} \mathbb{E} [e^{t(\rho_j - \mu_{\rho_j})}] \leq n \prod_{j=0}^{m-1} e^{\frac{1}{8} t^2 \beta_{\rho_j}^2} = n e^{\frac{1}{8} t^2 \sum_j \beta_{\rho_j}^2} \quad (13)$$

By merging Equation (10) and Equation (13) we get:

$$\mathbb{E}[tZ] = t \mathbb{E}[Z] \leq \log \left(n e^{\frac{1}{8} t^2 \sum_j \beta_{\rho_j}^2} \right) = \log n + \frac{1}{8} t^2 \sum_j \beta_{\rho_j}^2 \quad (14)$$

which holds for every $t \in \mathbb{R}$. By choosing $t = \frac{\sqrt{8 \log n}}{\sqrt{\sum_j \beta_{\rho_j}^2}}$, we finally obtain:

$$\mathbb{E}[Z] = \mathbb{E}[\tau] - \mu_\lambda = \Delta \leq \frac{1}{\sqrt{2}} \sqrt{\sum_{j=0}^{m-1} \beta_{\rho_j}^2 \sqrt{\log n}} \quad (15)$$

where the two main terms in the product at the right-hand side depend respectively on the variability of the paths and on their number, thus proving our result. Note that Equation (15) identifies the terms that have an impact on Δ , and it also bounds the *degree* of such impact. Since the IPE Model represents a worst case, the provided bound is applicable to general POSs, too.

5 Empirical Analysis of the Asymptotic Behavior

We now start to employ the mathematical framework from Section 4 for an empirical evaluation, to get a better grasp of the behavior of the makespan deviation. Our main tool will be the bound from Equation (15). As a preliminary, when moving to real POSs some of the parameters of the IPE Model cannot be exactly measured and must be approximated. As a guideline, we use the parameters of the critical path in the POS as representative for those of the paths in the IPE model. Hence:

- the value μ_λ corresponds to $\tau(\mathbb{E}[\delta])$, i.e. to $\frac{1+\alpha}{2} T$.
- each β_{ρ_i} is equal to $(1-\alpha)D_i$.
- for m , we take the number of activities on the critical path.

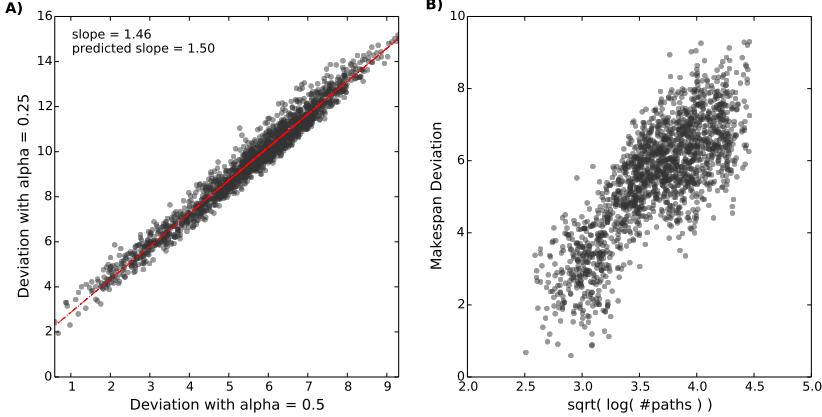


Fig. 5. A) Normalized Deviations with $\alpha = 0.5$ and $\alpha = 0.25$. B) Dependence of the deviation on the number of paths.

– for n , we take the total number of paths in the POS.

We expect the largest approximation error to come from considering all paths (regardless the degree of their dependence) and from assuming that all of them are critical. Now, we proceed by investigating how the makespan deviation Δ depends on the two main terms from Equation (15).

Dependence on the Variability: For our experimentation we have that:

$$\sqrt{\sum_{j=0}^{m-1} \beta_{\rho_j}^2} = \sqrt{\sum_{j=0}^{m-1} ((1-\alpha)D_i)^2} = (1-\alpha) \sqrt{\sum_{j=0}^{m-1} D_i^2} \quad (16)$$

Hence, the variability term in Equation (15) grows linearly with $(1-\alpha)$. Therefore, in order to check if the bound reflects correctly the dependence of Δ on the variability, we can repeat our simulation with different α values and see if we observe a linear change of the makespan deviation. We have experimented with $\alpha = 0.1, 0.25, 0.5$, and 0.75 .

Figure 5A shows a scatter plot for j60 where the makespan deviation for $\alpha = 0.25$ and $\alpha = 0.5$ are compared. The presence of a linear correlation is apparent. Equation (16) allows one to predict the slope of the line for two values α'' and α' , which should be $\frac{1-\alpha''}{1-\alpha'}$. For $\alpha'' = 0.25$ and $\alpha' = 0.5$, we get 1.5. By fitting a trend line it is possible to measure the actual slope, which in this case is 1.46, remarkably close to the predicted one. **This all points to the fact that the asymptotic dependence identified by our bound is in fact tight.**

Dependence on the Number of Paths: The path related term in the bound predicts that the makespan deviation will grow (in the worst case) with $\sqrt{\log n}$. In

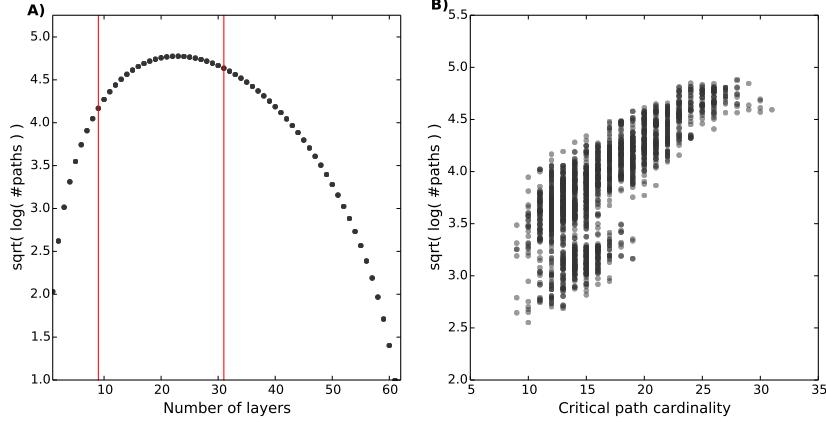


Fig. 6. The path-dependent term in the bound vs. the path cardinality.

this section, we investigate how accurate this estimate is in practice. In a first attempt to assess the correlation between Δ and the number of paths, we simply plot the deviation values against $\sqrt{\log n}$, using (we recall) the total number of paths in the graph as a proxy for the number of critical paths. If the bound reflects a tight asymptotic dependence, we should observe a linear correlation.

Figure 5B shows such a plot for j60 and the correlation does not appear to be linear. This may hint to an overestimation in our bound, but it could also be due to our approximations, or to the presence of correlations between the variability- and the path-related term in Equation (15). Now, we have recalled that the number of paths in a graph may be exponential in the graph size. The number of *independent* paths, however, is polynomially bounded by $|A|$, i.e. the size of the graph. In fact, every activity can be part of at most one independent path. **For the number of paths to be exponential, the activities must have multiple predecessors and successors.**

Layered Graph Approximation: Next, we will investigate the relation between the number of paths and their structure. This can be conveniently done on a layered graph, i.e. a graph where the nodes are arranged in m layers, such that layer $k - 1$ is totally connected with layer k . The number of paths in a layered graph is estimated by the quantity $(q/m)^m$, where $q = |A|$ is the graph size. There are two consequences: 1) to increase the number of paths exponentially, we must increase their cardinality; 2) since m appears also at the denominator, at some point the number of paths will start to *decrease* with growing m values.

This can be observed in Figure 6A, reporting the value of $\sqrt{\log(q/m)^m}$ for $q = 62$ and m ranging in $\{1 \dots 62\}$. Figure 6B reports instead the value of $\sqrt{\log(n)}$ over the cardinality of the critical path for the j60 benchmark, which also features 62 activities per graph (60 + 2 fake nodes). The two red bars in Figure 6A mark the minimum and the maximum for the critical path cardinality

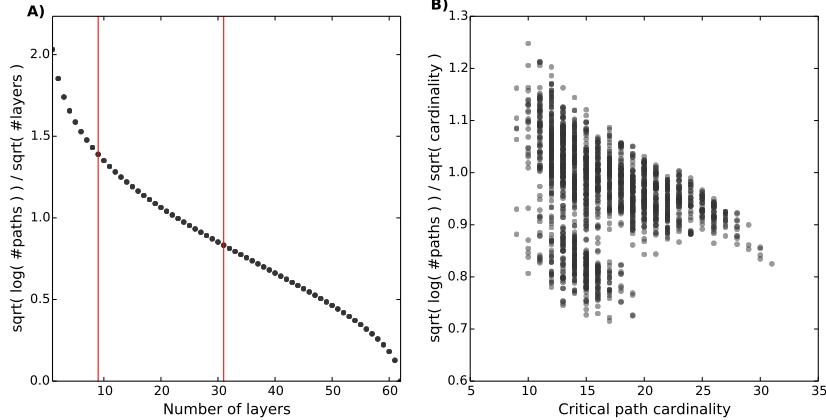


Fig. 7. The makespan deviation decreases with the cardinality of the critical path

in j60. The shape (and the range!) of the empirical plot matches closely enough that of the highlighted segment of the theoretical curve. At a careful examination, this makes a lot of sense. When resolving resource conflicts, a scheduling solver operates by arranging the activities into a sequence of parallel groups. **Tighter resource restrictions lead to longer sequences and fewer parallel activities in each group. In a POS, each of those groups translates to a layer, making a layered graph a fairly good approximate model.**

Path Cardinality and Variability: Now, we will show that the variability of a path π gets smaller if π contains a large number of activities with uncertain duration. If most of the activities in the graph have uncertain duration, this means that the variability of a path π will decrease with its cardinality. For our proof, we will assume for sake of simplicity that all ρ_i variables in the IPE Model are identical. Hence, in our experimental setting, their variability will be equal to $\frac{1}{m}(1-\alpha)T$. If we also assume to have a layered graph, Equation (15) becomes:

$$\Delta \leq \frac{1}{\sqrt{2}} \sqrt{m \beta_{\rho_j}^2} \sqrt{\log \left(\frac{q}{m} \right)^m} \quad (17)$$

$$\text{i.e. } \Delta \leq \frac{1}{\sqrt{2}} \frac{1}{\sqrt{m}} (1-\alpha) T \sqrt{m \log \left(\frac{q}{m} \right)} \quad (18)$$

In Equation (18), the role of path cardinalities in the deviation bound becomes explicit. In particular, paths with a lot of variables are less likely to have very long extreme lengths, reducing the deviation by a \sqrt{m} factor. The degree of the dependence is such to completely counter the exponential growth of the path number, so that the product $\frac{1}{\sqrt{m}} \sqrt{m \log(q/m)}$ behaves like $\sqrt{\log(q/m)}$.

Hopefully Equation (18) provides a reasonable approximation even when its simplifying assumptions are not strictly true. For testing this, in Figure 7A and

7B we report, next to each other, the value of $\sqrt{\log(q/m)}$ for m ranging in $\{1 \dots 62\}$ (which is strictly decreasing) and the value of $\frac{1}{\sqrt{m}}\sqrt{\log(n)}$ over the critical path length for j60. As in Figure 6A we use red lines to mark on the theoretical plot the minimum/maximum of the critical path cardinality for j60. Again, the figures are similar enough in terms of both shape and range. Hence, **if most of the activities have uncertain duration, an increase of the path cardinality tends to have a beneficial effect on the deviation Δ .**

Back to our Main Result: In Equation (18) the deviation bound is proportional to T . However, under the simplifying assumption that all activities are identical, increasing T requires to increase m as well, causing a reduction of the term $\frac{1}{\sqrt{m}}\sqrt{m \log(q/m)}$. It follows that the deviation should grow sub-linearly with the makespan. We conjecture that this same mechanism is at the base of the sub-linear dependence we have observed in Figure 3A and checked by analyzing the normalized deviation $\Delta/\tau(\mathbb{E}[\delta])$.

Now, we already know that the ratio $\frac{1}{\sqrt{m}}\sqrt{m \log(q/m)}$ does decrease with m for our benchmarks. Hence, if we observe a linear correlation between m and $\tau(\mathbb{E}[\delta])$, we should expect our bound to grow sub-linearly. In Figure 8A we report, for the j60 benchmark, the length of the critical path over the makespan with average duration, i.e. $\tau(\mathbb{E}[\delta])$. The plot confirms the existence of a linear correlation. At this point, repeating our experiment from Figure 3B with the (normalized) bound values instead of the deviation would be enough to confirm our conjecture. However, we have an even more interesting result. Figure 8B reports a scatter plot having on the x-axis the normalized bound and on the y-axis the normalized deviation. The plot shows the existence of a clear linear correlation between the two. This has two important consequences: **1) the hypothesis that the sub-linear growth is due to a correlation between the number of paths and their variability is consistent with the observed data.** Moreover, **2) the bound based on the IPE Model provides a tight prediction of the rate of growth of the normalized deviation.**

6 Concluding Remarks

Our results: In this paper we provide a detailed empirical and theoretical analysis of general run time properties of POSs. We observe, in a variety of settings, the existence of a very strong correlation between the expected makespan and the makespan obtained by assuming that all activities have their expected duration. Specifically, their difference (makespan deviation) appears to exhibit sub-linear variations against makespan changes. As a likely cause for this behavior, we suggest a strong link existing between the number of paths in a POS (that tends to increase the deviation) and the number of activities with uncertain durations in the paths (that makes the deviation smaller). We provide support for our hypothesis by means of a mathematical framework (the IPE Model) and of an extensive empirical evaluation. In the process, we end up identifying a number of important mechanisms that determine the behavior of a POS.

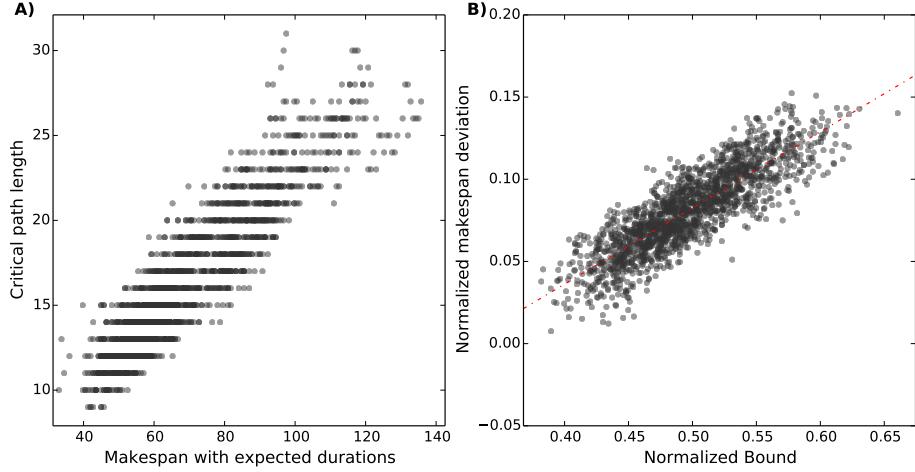


Fig. 8. A) Linear correlation between the makespan and the critical path cardinality.
B) Normalized bound from Equation (15) over the normalize deviation $\Delta/\tau(\mathbb{E}[\delta])$

It is important to observe that the strong resilience of the deviation to makespan variations represents a resilience to *scheduling decisions*. As an immediately applicable consequence, this makes it possible to disregard the duration uncertainty and build a POS so as to optimize the makespan of the scenario where all the activities take their expected value. The solution of such *deterministic* problem will have a good chance to be close to optimal for its *stochastic* counterpart as well. These result can be profitably applied to dramatically lower the complexity of many real world scheduling problems.

Limitations and Open Problems: The applicability conditions of our result require a deeper investigation. The single most important assumption for the validity of our analysis is the independence of the activity durations. Strongly dependent durations (e.g. simultaneous failures) may undermine our results and will certainly deserve a dedicated analysis. Moreover, we will need to adjust the approximations from Section 5, to tackle problems where only a handful of activities have uncertain duration (as opposed to all of them like in our experiments).

We are also interested in the identification of extreme cases: the IPE Model predicts that POSs with a lot of short and independent paths (e.g. Open Shop Scheduling solutions with many resources) should have the largest (normalized) makespan deviations. It is worth investigating what the rate of growth of the deviation would be in that case.

Finally, while we have a good understanding of why in our experimentation the makespan deviation *varies* little, we still don't know exactly why it *is* also very small. This open question, which is also tightly connected with the problem of predicting the value of the deviation, rather than its rate of growth. We plan to address both topics in future research.

Appendix A

Since τ in the IPE Model is non-negative, the expected value $\mathbb{E}[\tau]$ can be computed using the alternative formulation:

$$\mathbb{E}[\tau] = \int_0^\infty (1 - F_\tau(x))dx \quad (19)$$

Let $\hat{F}_\lambda(x)$ be a second distribution such that:

$$\int_0^\infty \hat{F}_\lambda(x)dx \leq \int_0^\infty F_\lambda(x)dx \quad (20)$$

Our goal is to prove that by replacing $F_\lambda(x)$ with $\hat{F}_\lambda(x)$ we increase the expected makespan. Formally, we want to prove that:

$$\int_{\alpha T}^\infty (1 - F_\lambda(x)^n)dx \leq \int_0^\infty (1 - \hat{F}_\lambda(x)^n)dx \quad (21)$$

$$\text{i.e.:} \quad \int_0^\infty \hat{F}_\lambda(x)^n dx \leq \int_0^\infty F_\lambda(x)^n dx \quad (22)$$

$$\text{i.e.:} \quad \int_0^\infty \hat{F}_\lambda(x)^{n-1} \hat{F}_\lambda(x)dx \leq \int_0^\infty F_\lambda(x)^{n-1} F_\lambda(x)dx \quad (23)$$

After applying integration by parts, we obtain:

$$\begin{aligned} & \left[\hat{F}_\lambda(x) \int_0^x \hat{F}_\lambda(x)^{n-1} dx \right]_0^\infty - \int_0^\infty \hat{F}_\lambda(x)^{n-1} \hat{f}_\lambda(x)dx \leq \\ & \leq \left[F_\lambda(x) \int_0^x F_\lambda(x)^{n-1} dx \right]_0^\infty - \int_0^\infty F_\lambda(x)^{n-1} f_\lambda(x)dx \end{aligned} \quad (24)$$

Where $f_\lambda(x)$ is the PDF of λ and (by definition) it is the derivative of the CDF $F_\lambda(x)$. The notation for $\hat{F}_\lambda(x)$ is analogous. Hence we get:

$$\begin{aligned} & \left[\hat{F}_\lambda(x) \int_0^x \hat{F}_\lambda(x)^{n-1} dx \right]_0^\infty - \left[\frac{1}{n} \hat{F}_\lambda(x)^n \right]_0^\infty \\ & \leq \left[F_\lambda(x) \int_0^x F_\lambda(x)^{n-1} dx \right]_0^\infty - \left[\frac{1}{n} F_\lambda(x)^n \right]_0^\infty \end{aligned} \quad (25)$$

Now, by the definition of CDF, we have that $F_\lambda(0) = \hat{F}_\lambda(0) = 0$ and $\lim_{x \rightarrow \infty} F_\lambda(x) = \lim_{x \rightarrow \infty} \hat{F}_\lambda(x) = 1$. Therefore, by substitution and by combination with Equation (22) we obtain:

$$\int_0^\infty \hat{F}_\lambda(x)^n dx \leq \int_0^\infty F_\lambda(x)^n dx \Leftrightarrow \int_0^\infty \hat{F}_\lambda(x)^{n-1} dx \leq \int_0^\infty F_\lambda(x)^{n-1} dx \quad (26)$$

Which is a recursive step. By repeated application until $n = 2$ the result is proven.

References

1. M. A. Aloulou and M. C. Portmann. An efficient proactive reactive scheduling approach to hedge against shop floor disturbances. In *Proc. of MISTA*, pages 223–246, 2005.
2. J. C. Beck and A. J. Davenport. A survey of techniques for scheduling with uncertainty. Available from <http://www.eil.utoronto.ca/profiles/chris/gz/uncertainty-survey.ps>, 2002.
3. J. C. Beck and N. Wilson. Proactive Algorithms for Job Shop Scheduling with Probabilistic Durations. *Journal on Artificial Intelligence Research (JAIR)*, 28:183–232, 2007.
4. Amedeo Cesta, Angelo Oddi, and S.F. Smith. Scheduling Multi-Capacitated Resources under Complex Temporal Constraints. *Proc. of CP*, pages 465–465, 1998.
5. D. Godard, P. Laborie, and W. Nuijten. Randomized Large Neighborhood Search for Cumulative Scheduling. In *Proc. of ICAPS*, pages 81–89, 2005.
6. W Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American statistical association*, 58(301):13–30, 1963.
7. G. Igelmund and F. J. Radermacher. Preselective strategies for the optimization of stochastic project networks under resource constraints. *Networks*, 13(1):1–28, January 1983.
8. J. L. W. V. Jensen. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica*, 30(1):175–193, 1906.
9. R. Kolisch. PSPLIB - A project scheduling problem library. *European Journal of Operational Research*, 96(1):205–216, January 1997.
10. P. Laborie. Complete MCS-Based Search: Application to Resource Constrained Project Scheduling. In *Proc. of IJCAI*, pages 181–186. Professional Book Center, 2005.
11. M. Lombardi, M. Milano, and L. Benini. Robust Scheduling of Task Graphs under Execution Time Uncertainty. *IEEE Transactions on Computers*, 62(1):98–111, January 2013.
12. N. Policella, A. Cesta, A. Oddi, and S. F. Smith. From precedence constraint posting to partial order schedules: A CSP approach to Robust Scheduling. *AI Communications*, 20(3):163–180, 2007.
13. N. Policella, S. F. Smith, A. Cesta, and A. Oddi. Generating Robust Schedules through Temporal Flexibility. In *Proc. of ICAPS*, pages 209–218, 2004.
14. R. Rasconi, N. Policella, and A. Cesta. SEam: Analyzing schedule executability through simulation. In *Proc. of Advances in Applied Artificial Intelligence*, pages 410–420, 2006.
15. E. Taillard. Benchmarks for basic scheduling problems. *European Journal of Operational Research*, 64(2):278–285, 1993.
16. S. Van de Vonder. A classification of predictive-reactive project scheduling procedures. *Journal of Scheduling*, 10(3):195–207, 2007.
17. S. Van de Vonder. Proactive heuristic procedures for robust project scheduling: An experimental analysis. *European Journal of Operational Research*, 189(3):723–733, 2008.