A Bayesian approach to set the tolerance limits for a statistical project control method

Zhi Chen[a] , Erik Demeulemeester[b], Sijun Bai[a], Yuntao Guo[a]

[a]*School of management, Northwestern Polytechnical University, Xi'an 710129, Shaanxi, PR China*
[b]*Research Center for Operations Management, Department of Decision Sciences and Information Management, Faculty of Economics and Business, KU Leuven, BE-3000 Leuven, Belgium*

Corresponding author at: School of Management, Northwestern Polytechnical University. 1 Dongxiang Road, Chang'an District, Xi'an 710129, Shaanxi, PR China.

E-mail address: chenzhi8418@gmail.com (Z. Chen),

Erik.Demeulemeester@kuleuven.be (E. Demeulemeester).

A Bayesian approach to set the tolerance limits for a statistical project control method

Abstract: In this paper we address the project schedule control problem under an uncertain environment. We propose a new method to set the tolerance limits based on the Earned Value Management/Earned Schedule (EVM/ES) schedule performance metrics. These tolerance limits can help a project manager to identify whether the schedule deviations from the baseline schedule are within the possible deviations derived from the expected variability of the project or if corrective actions must be taken to get the project back on track. We view the project control problem as a statistical hypothesis test with the null hypothesis being that the project progress is out of control. First, a simulation is performed to generate two types of empirical conditional distributions of the monitored schedule indicator. Afterwards, an algorithm that uses the derived conditional distributions as inputs is proposed to optimize the tolerance limits. An extensive computational experiment is carried out to assess the performance of the proposed approach. Additionally, sensitivity experiments are conducted to analyze four underlying factors that may influence the power of the proposed method. Experimental results show that our approach can keep the first type error under the required level ($\alpha = 0.05$) in any situation, meanwhile reducing the second type error significantly compared with three other methods in the literature.

Keywords: Project management; Schedule control; Bayesian approach; Earned value management; Earned schedule

1. Introduction

Project control lies at the core of project management, it involves comparing the actual performance of the project with a baseline schedule and analyzing the deviations. When the differences indicate that the project is behind schedule or exceeding the planned budget, the project manager may take early corrective actions to get the project back on track.

The classical and most popular way for project control is Earned Value

Management (EVM). EVM is based on the project Work Breakdown Structure (WBS), and incorporates the time and cost control in a unified framework. To be specific, EVM uses three measures: the planned value (PV) or budgeted cost of work scheduled (BCWS), the actual cost (AC) or actual cost of work performed (ACWP), and the earned value (EV) or budgeted cost of work performed (BCWP). The use of these three metrics can yield four indicators:

- Cost variance (CV=EV-AC)
- Cost performance index (CPI=EV/AC)
- Schedule variance (SV=EV-PV)
- Schedule performance index (SPI=EV/PV).

Whenever CV<0 and CPI<1, the project demonstrates a budget overrun. At the same time, if SV<0 and SPI<1, the project is delayed. Figure 1 illustrates how EVM works. As shown by Figure 1, EVM is a high level management method: both EV and AC are aggregate metrics. If warning signals from EVM occur, the project manager needs to drill down to a low level WBS in order to find the problematic activities.

The traditional EVM has been criticized by many researchers (Lipke 2003; Vanhoucke and Vandevoorde 2007; Khamooshi and Golafshani 2014) that it uses the cost-based data to assess the cost performance as well as the schedule performance of projects. For example, a negative SV in money units cannot give managers the information how long the project is behind the schedule. Another shortcoming of EVM is the loss of controllability when the project is close to its end. As all the planned activities will be nearly finished, the EV will tend to the PV, and, as a consequence, the SV will converge to zero and the SPI will conclude at one. This gives project managers the illusive impression that the project is a little bit late even if the project actually suffers a serious delay. To overcome this limitation, Lipke (2003) proposed the use of the Earned Schedule (ES) instead of EV for schedule control. Figure 1 clearly demonstrates this concept: ES is the time when the current earned value should have been achieved. Based on the ES, we have two additional performance indicators, SV(t)=ES-AT, and SPI(t)=ES/AT, where AT is the actual time. Whenever SV(t)<0 and SPI(t)<1, the project is behind schedule. Otherwise, the project is in good condition.

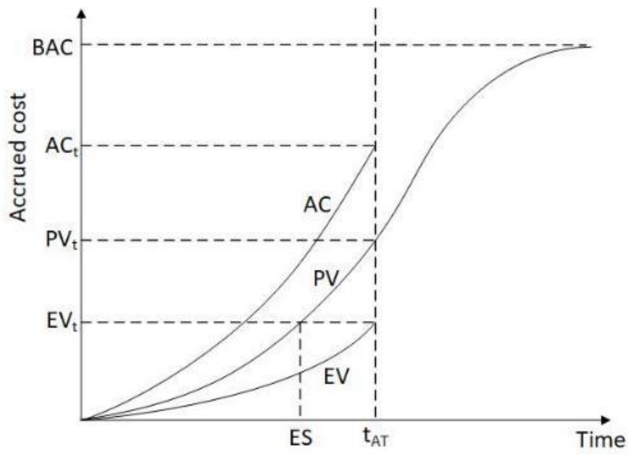

Figure 1: Earned value and earned schedule

Although EVM/ES is a powerful technique that can timely detect deviations from the plan, it does not take the stochastic nature of a project into account. In practice, projects are subject to considerable uncertainties: even if the project goes smoothly, the actual cost and progress do not always happen according to plan. In other words, the indicators in EVM/ES will distribute around their expected value (0 for CV, SV and SV(t), 1 for CPI, SPI and SPI(t)). At some review time, the schedule performance indicators may report that the project is delayed from the baseline. However, this does not automatically mean that corrective actions should be taken right now. The project manager needs to evaluate whether this delay is within the possible range of variability or if it is caused by some structural problems. In order to make such decisions, the project manager needs the tolerance limits for taking corrective actions. Whenever the indicators exceed the tolerance limits, the project manager has enough confidence to believe that some actions must be taken now to get the project back on track, otherwise the project will not finish on time and within budget.

In this paper, we focus our attention on the project schedule control and address the setting of the tolerance limits from the perspective of statistical control methods. We view the project control problem as a statistical hypothesis test: the null hypothesis is that the project is late. If no signal of breaking the tolerance limits occurs, we reject the null hypothesis and believe the project will be finished on time. It is possible that such judgement may be false, but the probability will be kept as small as possible ($\alpha < 0.05$) for the whole schedule control process in our method. First, we use simulation to get four conditional probability distributions of the schedule performance indicators. Based on these distributions we develop an algorithm to optimize the tolerance limits. We test our approach on a large set of project instances and compare our results with three other methods that appeared in the literature. Experimental results show that our method has more discriminative power than the other approaches.

The remainder of the paper is organized as follows. In Section 2, we present a literature review on the project schedule control methodologies under an uncertain environment. Subsequently, Section 3 gives our new methodology for constructing the tolerance limits. In Section 4, the design of computational test experiments and the performance evaluation metrics are discussed. Section 5 presents our computational results and a comparison with other approaches from the literature. Finally, some conclusions are drawn to highlight the contributions given by the paper to the project control problem in Section 6.

2. Literature review

In this section, a literature review on the project schedule control methodologies under an uncertain environment is given. Generally speaking, these methods can be divided into two categories: analytical methods and statistical methods. The analytical methods are usually based on the concept of buffers in the project, and use some simple rules as the tolerance limits. On the other hand, the statistical methods often employ some statistical tools to aid the project schedule control, such as statistical process control, statistical machine learning techniques, etc.

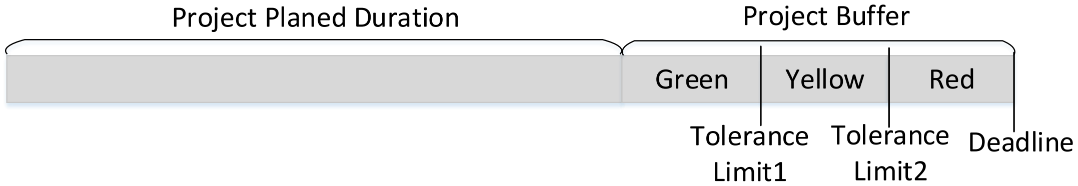

Figure 2: Project schedule control in the critical chain method

Perhaps the easiest and most intuitive way for schedule control under uncertainties is the buffer consumption monitoring proposed by Goldratt (1997). In this control system, a project buffer is divided into three zones as shown by Figure 2. The zones are often represented in green, yellow and red, representing the expected variation zone, the normal variation zone and the abnormal variation zone respectively. Typically, these zones are each sized to one third of the buffer. When the observed buffer consumption is in the green zone, no special action is required. When the buffer consumption enters the yellow zone, the project manager should keep an eye on the progress: the project is still under control, but the manager should prepare for action. Moreover, when the red zone is reached, the project manager must take actions immediately, otherwise the problems will possibly jeopardize the project due date. In Goldratt's buffer consumption management, the project buffer is usually divided into three equal parts and does not change during the whole project lifecycle. This simple way is often questioned for its efficiency. Hu, Cui and Demeulemeester (2015) proposed an alternative method, called Relative Buffer Management Approach (RBMA). In their method, the two tolerance limits vary linearly over the proportion of the critical chain completed (PCC). As shown in Figure 3, at any time point t, the proportion of the buffer consumed (PBC) for green, yellow and red zone can be calculated according to two linear functions. The basic idea of RBMA is that at the beginning of the project more project buffer should be given to the red zone, while at the end of the project more project buffer should be given to the green zone.

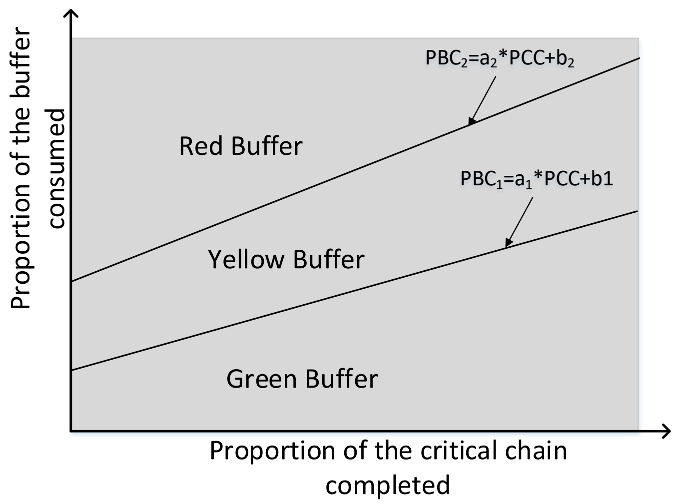

Figure 3: Tolerance limits in the relative buffer management approach

Integrating the project buffer into EVM/ES is another common way in the literature. Different from buffer consumption monitoring, the process of schedule control in this method is to monitor the EVM/ES metrics, and there is only one tolerance limit at any review point. If the EVM/ES metrics value at time t are less than the tolerance limit, then the project manager should take corrective actions. Usually the tolerance limit is calculated based on the project buffer. There are three common methods to set the tolerance limits in the literature. The first method is the planned duration based buffer consumption (PD-BC) way (Colin and Vanhoucke 2015a). Taking the SPI(t) and SV(t) as an example, the tolerance limits for the SV(t) and SPI(t) at any review time instant t are defined as

$$\overline{SV}(t)_t = -PB \cdot \frac{t}{PD + PB} \tag{1}$$

$$\overline{SPI}(t)_t = 1 - \frac{PB}{PD + PB} = \frac{PD}{PD + PB}. \tag{2}$$

where PB is the project buffer. The method shares the same idea as Hu et al. (2015) that the project buffer should be linearly distributed during the whole project lifecycle. The second way is the planned value based buffer consumption (PV-BC) way. In the work of Martens and Vanhoucke (2017) they claimed that setting the allowable buffer consumption linearly with the project duration omits the fact that normally the PV does not increase linearly with the project duration. For most projects, the curve of the cumulative PV is S-shaped, which means that most of the project effort is usually performed in the middle of its life cycle. Based on this fact, they suggested that the allowable buffer consumption should be assigned linearly with the EV, not the project duration. In PV-BC, the tolerance limits for the SV(t) and SPI(t) at any review time instant t are calculated as

$$\overline{SV}(t)_t = -PB \cdot \frac{EV_t}{PV} \tag{3}$$

$$\overline{SPI}(t)_t = 1 - \frac{PB \cdot EV_t}{t \cdot PV}, \tag{4}$$

where EV_t is the earned value at time t. The last way is the risk based buffer consumption (RI-BC) way proposed by Pajares and Lopez-Paredes (2011). At time t, the schedule risk baseline (SRB_t) is the schedule variance of the activities that are completed before ES_t in the baseline schedule. Thus, the total project schedule variance is SRB_{PD}. Both of these values are derived from simulation. Based on the SRB_t the tolerance limits for the SV(t) and SPI(t) can be defined as:

$$\overline{SV}(t)_t = -PB \cdot \frac{SRB_t}{SRB_{PD}} \tag{5}$$

$$\overline{SPI}(t)_t = 1 - \frac{PB \cdot SRB_t}{t \cdot SRB_{PD}}. \tag{6}$$

The main difference among the PD-BC, PV-BC and RI-BC lies in the project buffer allocation approaches during the project lifespan. Figure 4 demonstrates the three different buffer allocation approaches at the different stages of the project.

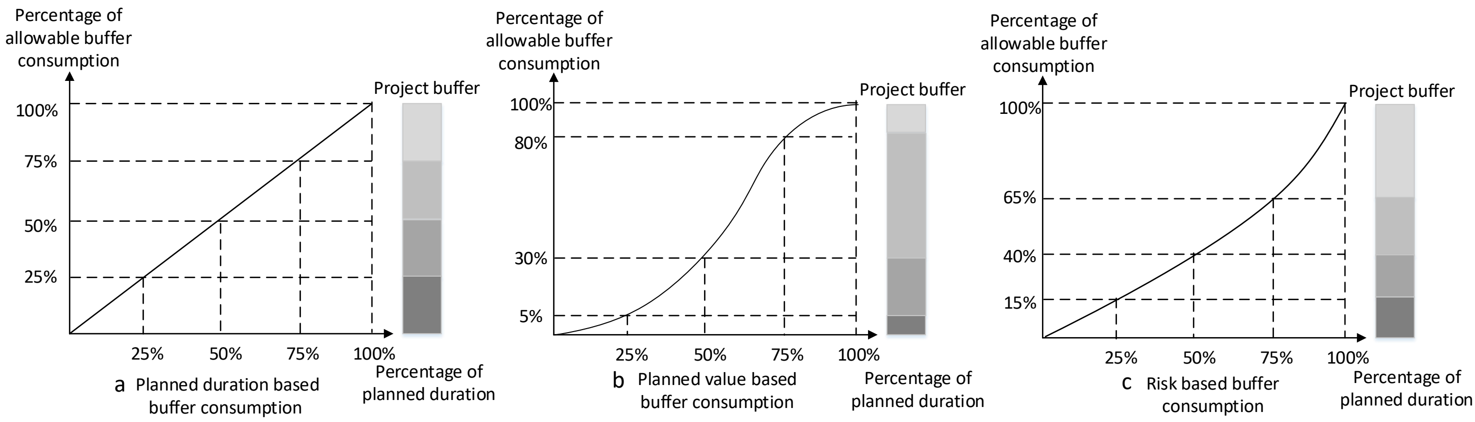

Figure 4: Three different buffer allocation approaches

The first endeavor to apply the statistical method to project control is the use of statistical process control (SPC) charts. These control charts are usually used to routinely monitor and control processes in manufacturing. As shown in Figure 5, the chart contains a center line that represents the mean value for the in-control process. Two other horizontal lines, called the upper control limit (UCL) and the lower control limit (LCL), are also shown on the chart. This typical SPC chart combined with other rules or patterns is used to detect some assignable causes of variation. In the literature, most of the related research (Bauch and Chung 2001; Wang et al. 2006; Leu and Lin 2008; Aliverdi, Naeni, and Salehipour 2013; Colin and Vanhoucke 2015b) applies the Shewhart SPC charts or individual control charts to monitor the EVM/ES based indicators (CV, CPI, SV, SPI, SV(t), SPI(t)) or any transformation of them. The SPC charts may be an intuitive tool to control the project schedule and cost. However, there are three shortcomings when applying this method in reality. The first problem is the normality of the metrics. One requirement for the SPC charts is that the values of the monitored quality characteristic must be normally distributed. However, there is no clear evidence that EVM/ES based indicators fit such distribution (Wang et al. 2006;

Lipke 2002). Another assumption of SPC charts is the independence of the observations, but the EVM/ES based indicators usually are considered to be dependent (Martens and Vanhoucke 2017). The second disadvantage of SPC charts for project control is the availability of the input data to construct the central line as well as the upper and lower control limits. Usually, the source of such data comes from records of similar projects or the early stage of the project. However, the quality of such data is questioned, because in most cases every project is a unique temporary endeavor, and the early stage of a project is generally the most unstable phase during the life cycle. The last grave flaw of SPC charts is the static control limits. The SPC charts are originally designed to monitor the quality of an on-going process, such a process renders the UCL and LCL to be symmetric and static over time. However, the process of completing a project is different from that of producing products in manufacturing lines. The control limits of a project usually fluctuate during the whole life cycle. In order to overcome the drawbacks of SPC charts, many simulation-based control methods are proposed in the literature. These approaches usually assume that the activity duration fits some kind of distribution (beta distribution, uniform distribution or exponential distribution), and then use simulation combined with some risk analysis techniques to set the tolerance limits for projects. For example, in the works of Colin and Vanhoucke (2014) and Acebes et al. (2014), the lower and upper control limits are constructed using the $\theta\mathrm{th}(1 < \theta < 1)$ and $(1-\theta)\mathrm{th}$ quantile of the derived metric's empirical cumulative distribution function, where θ is the level of confidence set by the project manager. Acebes et al. (2015) view the project control process as an anomaly detection, classification and regression problem. In their work, several different statistical learning methodologies are combined with simulation to predict the probabilities of success in time and the expected total duration of the project, when such probabilities are less than a predefined threshold (tolerance limit), the project is thought to be abnormal.

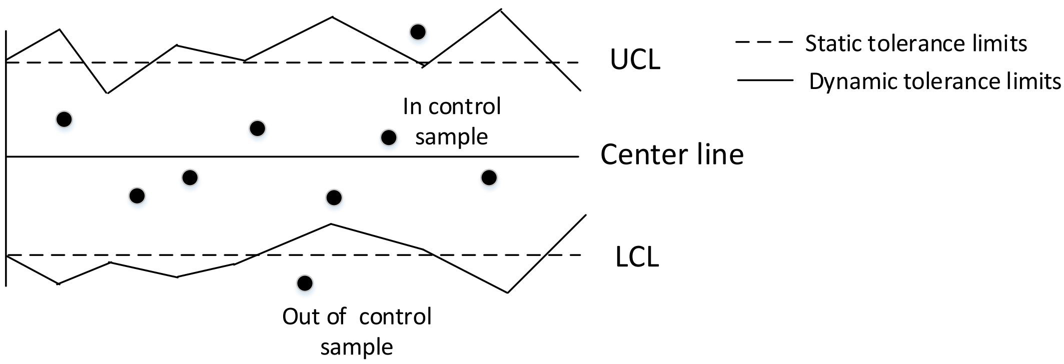

Figure 5: Illustrative control chart with static and dynamic tolerance limits

In Section 3, we propose a new way to set the tolerance limits for an EVM/ES system. Although our approach is also based on the simulation method, we designed an algorithm to optimize the tolerance limits instead of simply setting them as the $\theta\mathrm{th}$ or $(1-\theta)\mathrm{th}$ quantile of the empirical cumulative distribution function. We test our approach on the standard problem instances, and compare the results with the methods in Martens and Vanhoucke (2017), Colin and Vanhoucke (2014) and Pajares and Lopez-Paredes (2011) using several kinds of performance measurements. The evaluation results show that our tolerance limits are more discriminative. At the same time our approach is much more flexible which can allow users to set any control level based on different performance measurements.

3. A Bayesian approach for constructing the tolerance limits

In this section, we will first introduce two types of conditional probabilities, which will be used as the inputs of the algorithm. Subsequently, the detailed procedures of the algorithm will be presented. Due to the advantages of ES-based metrics in project schedule control, we only use the indicator SV(t) as the control metric in our approach.

3.1. Two types of conditional probabilities

During the project execution, the schedule control is not a real-time process. A project manager usually reviews the project progress at some discrete times. Assuming that there are in total K review points, so the tolerance limits in fact are a vector with K values, denoted by $\overline{SV}(t)=\left(\overline{SV}(t)_1,\overline{SV}(t)_2,\cdots,\overline{SV}(t)_K\right)$. In order to compare the performance of our method with some project buffer based control approaches, we add a project buffer at the end of the planned duration, so the deadline of the project, denoted by δ, is the sum of the planned duration and the project buffer.

Given a tolerance limit $\overline{SV}(t)_k$ at review point $k, k \in \pi = \{1, 2, \cdots, K\}$, two possibilities may occur: $SV(t)_k < \overline{SV}(t)_k$ or $SV(t)_k \geqslant \overline{SV}(t)_k$. If $SV(t)_k < \overline{SV}(t)_k$, then the control system will give a warning signal to the project manager that the project probably will exceed the deadline. Otherwise $SV(t)_k \geqslant \overline{SV}(t)_k$, which means the project is still in control. As a project manager, we want to set tolerance limits that make the probabilities $\mathrm{P}(RD > \delta | SV(t)_k < \overline{SV}(t)_k)$ and $\mathrm{P}(RD \leqslant \delta | SV(t)_k \geqslant \overline{SV}(t)_k)$ both as large as possible. The first conditional probability can be used to measure the efficiency of the tolerance limit $\overline{SV}(t)_k$, while the second conditional probability can be used to scale the reliability of the tolerance limit $\overline{SV}(t)_k$. In most cases, if we increase the first probability, then the second probability will decrease, this means that a high efficiency usually will result in a low reliability. Using Bayes' theorem, we can easily get another two conditional probabilities: $\mathrm{P}(SV(t)_k < \overline{SV}(t)_k | RD > \delta)$ and $\mathrm{P}(SV(t)_k < \overline{SV}(t)_k | RD \leqslant \delta)$. The first probability can be used to measure the performance of the tolerance limit $\overline{SV}(t)_k$, and the second one can be viewed as a measurement of overreaction. Now the best tolerance limit should make the first probability as large as possible and the second one as small as possible. Given the tolerance limit $\overline{SV}(t)_k$ for any $k \in \pi$, the four conditional distributions can be easily obtained from simulation. In our optimization algorithm, which will be elaborated in Section 3.2, either the distributions $\mathrm{P}(RD > \delta | SV(t)_k < \overline{SV}(t)_k)$ and $\mathrm{P}(RD \leqslant \delta | SV(t)_k \geqslant \overline{SV}(t)_k)$ or $\mathrm{P}(SV(t)_k < \overline{SV}(t)_k | RD > \delta)$ and $\mathrm{P}(SV(t)_k < \overline{SV}(t)_k | RD \leqslant \delta)$ can be used to optimize the tolerance limit $\overline{SV}(t)_k$. In order to distinguish them, we call the first two distributions the first type distributions and the last two distributions the second type distributions.

Similarly to the conditional probabilities at some tracking point, we can also get the conditional probabilities at the project level. However, at the project level, we only care about whether or not the warning signals are generated, without considering the precise number of generated signals. If we denote the event $SV(t)_k < \overline{SV}(t)_k$ and $SV(t)_k \geqslant \overline{SV}(t)_k$ as $S_k^{<}$ and $S_k^{\geqslant}$, respectively, then the event of giving a warning signal at the project level is $S_P^{<} = S_1^{<} \vee S_2^{<} \vee \cdots \vee S_K^{<}$, where the operator $\vee$ is the logical disjunction, and the reverse occasion of not giving a warning signal at the project level is $S_P^{\geqslant} = S_1^{\geqslant} \wedge S_2^{\geqslant} \wedge \cdots \wedge S_K^{\geqslant}$, where the operator $\wedge$ is the logical conjunction . Using $S_P^{<}$ and $S_P^{\geqslant}$ to replace $S_k^{<}$ and $S_k^{\geqslant}$ respectively, we can get the four conditional probabilities at the project level. These conditional probabilities have the same meaning as in the tracking point level, so they can be employed to assess the performance of all tolerance limits (Colin and Vanhoucke 2014, 2015a; Martens and Vanhoucke 2017).

3.2. The algorithm to set the tolerance limits

In our approach, we view the project control problem as a statistical hypothesis test. The null and alternative hypotheses are:

H_0: The project is out of control.

H_1: The project is in control.

Given a set of tolerance limits, if during the project control process the system gives a warning signal, we should accept the null hypothesis and take corrective actions. Otherwise, we should accept the alternative hypothesis. However, like for any hypothesis test, such decisions may make two types of mistakes as shown in Table 1. Usually, these two error rates are traded off against each other: the effort to reduce one type of error generally results in increasing the other type of error. In this problem, if the control system does not report any warning signal, we must have enough confidence $(1-\alpha)$ to ensure that the project will meet the deadline. Otherwise, the control process is meaningless. At the same time, we want our tolerance limits to have a high efficiency, because unnecessary corrective actions will increase the cost of the project. In order to find such tolerance limits, an optimization procedure is designed as outlined in Algorithm 1.

Table 1: The first and the second type errors in the project schedule control

	H_0 is true The project is out of control	H_1 is true The project is in control
Signal Accept null hypothesis	Right decision	Wrong decision Type II error
No signal Reject null hypothesis	Wrong decision Type I error	Right decision

For the two types of conditional distributions of the review points, either the first type or the second type can be used to optimize the tolerance limits. In Algorithm 1, we present our procedures using the second type distributions, but it can be easily adapted to be used with the first type distributions. In Section 5, we will compare the results of the two versions of the algorithm. Before optimizing the tolerance limits, a simulation with nrs runs will be performed to get the empirical cumulative distribution function of $SV(t)_k$ and the RD_i of every replication i. Our algorithm starts by setting all the tolerance limits to 0. As one can imagine, such tolerance limits will result in a high performance, but also a high overreaction. So the aim of the algorithm is to reduce the overreaction under the condition of meeting the required significance level. The basic idea of the algorithm is that we choose the review point that reduces the maximum overreaction at the expense of one unit of performance decrease (steps 5-9), and then decrease this tolerance limit by one step (step 11). For every review point k, we first calculate the $performance_k$ and $overreaction_k$, then we calculate the $performance_{k,step}$ and $overreaction_{k,step}$ presuming that the tolerance limit k is decreased by one step (step 3). In order to find the review point whose tolerance limit should be reduced in every loop, we first divide all the K review points into two sets: the first set ζ contains all the review points for which reducing the tolerance limit by one step does not change their performance value; the second set ξ includes the other review points for which reducing their tolerance limits will result in a performance decrease. Apparently, we should give priority to the review point that has a maximum overreaction decrease from the set ζ. If this set is empty then we choose the review point that has a maximum overreaction decrease at the expense of one unit of performance decrease from set ξ. During the while-loop, it may be the case that reducing the tolerance limit by one step cannot decrease the overreaction value for all the review points. However, the probability $\mathrm{P}(S_P^{<}|RD \leqslant \delta)$ is still larger than 0, and $\mathrm{P}(RD > \delta|S_P^{\geqslant})$ is smaller than the significance level. This is because the variable $step$ is too small, reducing the tolerance limit by one step makes no differences on the overreaction rate. In this case we only need to increase the value of $step$ and recalculate $performance_{k,step}$ and $overreaction_{k,step}$ for all the review points (step 10). Although our algorithm can adjust the $step$ automatically, the initial $step$ value should be set at a suitable level: if $step$ is too small, it may be the case that the algorithm needs too much computation time. On the contrary, if $step$ is too large, the algorithm cannot find the best tolerance limits and the confidence level may be surpassed.

Algorithm 1: Optimizing the tolerance limits

Input: The distribution of the $SV(t)_k, k \in \pi$; α; δ; $step$; $RD_i, i = 1, 2, \cdots, nrs$

Output: tolerance limits $\overline{SV}(t)$

1. set $\overline{SV}(t)_k = 0, k \in \pi$
2. calculate the probability $\mathrm{P}(RD > \delta | S_P^{\geqslant})$ and $\mathrm{P}(S_P^{<} | RD \leqslant \delta)$
3. calculate $performance_k = \mathrm{P}(SV(t)_k < \overline{SV}(t)_k | RD > \delta)$,

   $overreaction_k = \mathrm{P}(SV(t)_k < \overline{SV}(t)_k | RD \leqslant \delta)$,

   $performance_{k,step} = \mathrm{P}\big(SV(t)_k < (\overline{SV}(t)_k - step) | RD > \delta\big)$,

   $overreaction_{k,step} = \mathrm{P}\big(SV(t)_k < (\overline{SV}(t)_k - step) | RD \leqslant \delta\big)$ for all $k \in \pi$
4. **while** $\mathrm{P}(RD > \delta | S_p^{\geqslant}) < \alpha$ and $\mathrm{P}(S_p^{<} | RD \leqslant \delta) > 0$
5. $\zeta \leftarrow \{k | performance_{k,step} - performance_k = 0\}, \xi = \pi / \zeta$
6. **if** $\zeta \neq \varnothing$ $\quad i \leftarrow \arg\max(overreaction_k - overreaction_{k,step}), k \in \zeta$,

   $difference = overreaction_i - overreaction_{i,step}$
7. **if** $\xi \neq \varnothing$ $\quad j \leftarrow \arg\max\left(\dfrac{overreaction_k - overreaction_{k,step}}{performance_k - performance_{k,step}}\right), k \in \xi$,

   $ratio = \dfrac{overreaction_j - overreaction_{j,step}}{performance_j - performance_{j,step}}$
8. **if** $difference > 0$ $\quad k = i$
9. **else if** $ratio > 0$ $\quad k = j$
10. **else** $\quad step = step + 1$, calculate $performance_{k,step}$ and $overreaction_{k,step}$ for all $k \in \pi$, continue next loop
11. $\overline{SV}(t)_k = \overline{SV}(t)_k - step$,

    $performance_k = performance_{k,step}$,

    $overreaction_k = overreaction_{k,step}$,

    $performance_{k,step} = \mathrm{P}\big(SV(t)_k < (\overline{SV}(t)_k - step) | RD > \delta\big)$,

    $overreaction_{k,step} = \mathrm{P}\big(SV(t)_k < (\overline{SV}(t)_k - step) | RD \leqslant \delta\big)$
12. calculate the probability $\mathrm{P}(RD > \delta | S_p^{\geqslant})$ and $\mathrm{P}(S_p^{<} | RD \leqslant \delta)$

4. Design of experiments

4.1. Project instances and data generation

In order to assess the performance of the proposed method, we test our approach on a fictitious project data set and on a real-life project data set.

The artificial project data set is generated by RanGen (Demeulemeester, Vanhoucke, and Herroelen 2003; Vanhoucke et al. 2008), and this data set has been widely used in the literature (Colin and Vanhoucke 2014, 2015a; Colin et al. 2015; Martens and Vanhoucke 2017) to evaluate the performance of miscellaneous schedule control methods. The project networks in this data set are generated according to a serial/parallel (SP) topological indicator SP with 9 values between 0 and 1 (SP = {0.1, 0.2, … ,0.9}). For each level, 100 different project networks are generated with 30 non-dummy activities, so there are in total 900 instances in this data set. The indicator SP was first introduced by Vanhoucke et al. (2008), and is used to measure the closeness of a project to a completely serial or parallel project. If m is the maximal progressive level, which is defined as the maximum number of activities lying on a single path, and n is the number of total activities of the project, then $SP = \frac{m-1}{n-1}$. If the project is completely serial, then SP equals 1. On the contrary, for a completely parallel project, SP equals 0. In Section 5, we will analyze the influence of this indicator on the performance of our schedule control method. For every project network in this data set we randomly generate the expected duration and cost of every activity using integral numbers with a uniform distribution as in the works of Martens and Vanhoucke (2017) and Colin and Vanhoucke (2014). For every non-dummy activity, the planned duration d_i^* is sampled from $\mathrm{U}[8,56]$, the fixed cost is sampled from $\mathrm{U}[10,90]$, and the variable cost is sampled from $\mathrm{U}[100,900]$. The planned value of every activity equals the fixed cost plus the variable cost multiplied by its planned duration.

The real-life project data set is constructed by Batselier and Vanhoucke (Batselier and Vanhoucke 2015), which currently contains 125 projects collected from various sectors. Each project has a project card, which gives an overview of the project and indicates the authenticity and completeness of the project data. The authenticity of the project data is assessed by project authenticity and tracking authenticity. Full project authenticity means that the baseline activity duration, resource usage and activity cost data are all obtained directly from the project owner, and that the data collector does not make any personal assumptions. Similarly, full tracking authenticity implies that the actual activity durations, actual costs and real project makespans are all obtained from the project owner. In order to guarantee the validity of the real case study, we only choose the projects with full project and tracking authenticity. Finally, 93 out of the 125 projects meet our requirement, and are selected in the experiment.

4.2. Project progress simulation and tracking

In our experiments, a two-phase Monte Carlo simulation is performed to generate the fictitious activity durations. The sample size is 10,000 in each simulation, as this number can ensure that the ratio of the average standard deviation of the tolerance limits to the critical path length of the project is less than 1%. The first-phase simulation is used to construct the tolerance limits using the method in Section 3, while the second-phase simulation is meant to emulate real project executions, which will be used to assess the performance of the tolerance limits obtained from the first-phase. Both in the first and in the second-phase simulations, the activity durations are modeled using a beta distribution $B_1(\frac{d_i^*}{4}, \frac{23d_i^*}{8}, 2, 5)$ with a standard deviation of $0.42d_i^*$, where d_i^* is the expected duration of activity i. In the work of Colin and Vanhoucke (2014), a similar two-phase simulation is also used. In the first-phase, a controlled uniform distribution simulation which allows only a limited maximum variation is performed to construct the tolerance limits. In the second-phase another uniform distribution simulation with a larger variance is performed to simulate the real progress of the project. We do not adopt their way: on the contrary, we fix the variance of the second-phase simulation, and change the variance of the first-phase simulation from small to large. To be specific, we choose another two beta distributions $B_2(\frac{d_i^*}{2}, \frac{15d_i^*}{8}, 3, \frac{21}{4})$ and $B_3(\frac{d_i^*}{5}, 5d_i^*, \frac{5}{4}, \frac{25}{4})$ with standard deviations $0.22d_i^*$ and $0.62d_i^*$ respectively to construct the tolerance limits, and then use the same distribution B_1 to emulate real project executions and assess the three different tolerance limits. In Section 5, a sensitivity analysis will be undertaken to analyze the impact of the first-phase variation on the performance of the tolerance limits.

In most project simulation methods, one assumption is that the duration of the activities is independent. However, in reality, activities in a project often share some common factors that will influence their duration, such as the common use of resources. So usually the activity durations are dependent. In the literature, these common factors are classified as project level risks by Colin and Vanhoucke (2015a), which means they have a potentially disastrous impact on multiple activities in a project and on the overall project objectives. Thus the uncertainties of an activity come from two different sources: the variation of the activity itself and the project risk. In order to consider the project risks in the simulation, Trietsch et al. (2012) proposed a linear association approach to model the activity times. A set of n positive random variables $\{Y_i\}$ are linearly associated if $Y_i = BX_i$, where $\{X_i\}$ is another set of n independent positive random variables and B is also a positive random variable, independent from $\{X_i\}$. If the mean value of X_i and B are e_i and λ respectively, and if the variance of X_i and B are η_i^2 and σ^2 respectively, then the covariance of Y_i and Y_j is

$$\begin{aligned}\mathrm{COV}(Y_i, Y_j) &= \mathrm{E}\big\{[Y_i - \mathrm{E}(Y_i)][Y_j - \mathrm{E}(Y_j)]\big\} = \mathrm{E}[(BX_i - \lambda e_i)(BX_j - \lambda e_j)] \\ &= \mathrm{E}(B^2 X_i X_j - B\lambda e_j X_i - B\lambda e_i X_j + \lambda^2 e_i e_j) = \mathrm{E}(B^2) e_i e_j - \lambda^2 e_i e_j \\ &= (\lambda^2 + \sigma^2) e_i e_j - \lambda^2 e_i e_j = \sigma^2 e_i e_j \neq 0 \end{aligned} \tag{7}$$

In our two-phase simulation, we also use this linearly associated approach to model the activity durations. For every simulation of the project, a random risk factor B is sampled from a beta distribution, then all the sampled activity durations that only consider variation at the activity level are multiplied by the sampled risk factor. Although Trietsch et al. (2012) claimed that the risk factor is influenced by both additive and multiplicative causes and follows a lognormal distribution, we still use the beta distribution instead of a lognormal distribution, because in the lognormal distribution there may be the probability that a very small (close to 0) or a very large number is sampled: such extreme values result in a project duration that is impossible in reality.

Compared with the lognormal distribution, the beta distribution allows project managers to set a minimum and a maximum value, and determine the shape of the probability density function. In reality, the extreme bad risk factors are usually small probability events, so the right skewed distribution is more intuitive and practical. Table 2 lists three beta distributions that represent three linearly dependent levels. The parameters (a, b, m, μ) are the minimum, maximum, mode and mean values of the distribution. The three beta distributions are all right-skewed, and have a mean value that equals 1, which means they all generate a neutral risk on average. In order to get the distributions in Table 2, one only needs to calculate the shape parameters of the beta distribution function by solving the set of Eq. (8) as follows:

$$\begin{cases} p = -\frac{(a+b-2m)(a-\mu)}{(m-\mu)(a-b)} \\ q = \frac{(a+b-2m)(b-\mu)}{(m-\mu)(a-b)} \end{cases} \tag{8}$$

Table 2: Parameters of the risk beta distributions

Linearly dependent level	Parameters (a, b, m, μ)	Standard deviation σ
Low	(0.7, 3, 0.93, 1)	0.15
Medium	(0.6, 4, 0.88, 1)	0.22
High	(0.5, 5, 0.83, 1)	0.29

Both in the first and the second-phase simulation, the project progress is measured at its 5%, 10%, …, 95% total duration, i.e. the review time $t = k \cdot 0.05 \cdot \delta, k \in \pi$ and $K = 19$. In Colin and Vanhoucke (2014, 2015a) and Martens and Vanhoucke (2017), the project progress is reviewed when the project is 5%, 10%, …, 95% completed. For this review method, it may be the case that the time of the last review points already exceeds the deadline of the project, when the project suffers a serious delay. As the aim of the progress control is to make sure that the project can be finished before the deadline, there is no sense to review the progress when the project is already overdue. During the review process, it is assumed that the earned value of all activities accrues linearly over time.

4.3. Project control performance measurements

In Section 3.1 we have already proposed four project level performance metrics, which are efficiency, reliability, performance and overreaction. In order to have a comprehensive evaluation of the tolerance limits, we additionally introduce two performance metrics: they are the signal density (Martens and Vanhoucke 2017) and the average first time occurrence of a warning signal. For a project, if its progress is out of control, the tolerance limits should generate as many warning signals as possible, since every signal corresponds with an opportunity to take corrective actions. The signal density measures the average amount of warning signals the tolerance limits generate when a project is late. It is defined as follows:

$$signal\ density = \frac{\sum_{i=1}^{nrs}\sum_{k=1}^{K} f\big(SV(t)_k < \overline{SV}(t)_k\big)\cdot f(RD_i > \delta)}{\sum_{i=1}^{nrs} f(RD_i > \delta)}, \tag{9}$$

where $f(x)$ is an indicator function, defined as

$$f(x) = \begin{cases} 1 & \text{if } x \text{ is true} \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

Again, during the schedule control process, the project manager wants the tolerance limits to generate a warning signal as soon as the project is out of control. The earlier the tolerance limits generate a signal, the more time the project manager has to take corrective actions. The average first time occurrence of a warning signal is calculated as follows:

$$average\ first\ time\ occurrence = \frac{\sum_{i=1}^{nrs} \min(k|SV(t)_k < \overline{SV}(t)_k)\cdot f(RD_i > \delta)}{\sum_{i=1}^{nrs} f(RD_i > \delta)} \tag{11}$$

5. Results

5.1 General results

Table 3 summarizes the general performance results of our approach and three other methods from literature for the fictitious project data set. In this experiment, the project buffer size is set at 10% of the planned duration, and activity dependence is set at the medium level. The significance level α, which equals $1 - reliability$, is set at 5%. Note that Algorithm 1 cannot produce tolerance limits that yield an exact reliability of $1 - \alpha$. Normally, it will be smaller than $1 - \alpha$, but the difference is less than 1%, and can be neglected in practice. The average first time occurrence in Table 3 is the percentage value of the first time reporting a warning signal over the total project duration. From Table 3 we can see that there is no big difference between the evaluation results of type 1 and type 2 distributions: type 1 distributions yield a lower overreaction, while type 2 distributions produce a larger signal density and an earlier first time occurrence than the type 1 distributions. We compare the performance of our approach with three other methods in the literature, namely the θth quantile of the empirical cumulative distribution function (θth-ECDF) employed by Colin and Vanhoucke (2014), a PV-based buffer consumption (PV-BC) way introduced by Martens and Vanhoucke (2017), and a risk-based buffer consumption (RI-BC) way proposed by Pajares and Lopez-Paredes (2011). For the θth-ECDF method, we choose the θ value that can produce a 95% reliability. The other two methods are analytical approaches and do not have the freedom to set the reliability. As displayed in Table 3, the methods θth-ECDF, PV-BC and RI-BC all have a comparatively higher performance and signal density and can report a warning signal earlier than our method. However, they also yield a very high overreaction and a low efficiency, which means they falsely give too many warning signals for a project that can be completed before the deadline. We further adapt step 4 of Algorithm 1 and let it yield a tolerance limit that can produce a similar performance value with that of θth-ECDF, PV-BC and RI-BC methods. The comparison results are in Table 4. As presented in Table 4, with the same performance value, our method can greatly reduce the overreaction and increase the efficiency of the tolerance limits. This is especially remarkable when compared with the commonly used θth-ECDF method: our approach can improve the efficiency with 22.7 percentage points, and the overreaction rate is only about a quarter of θth-ECDF's value.

Table 3: Performances of different tolerance limits for the fictitious project data set

	Efficiency (%)	Reliability (%)	Performance (%)	Overreaction (%)	Signal density	Average first time occurrence(%)
Type 1	93.35	94.77	89.49	3.99	7.5	46
Type 2	93.82	94.75	89.43	4.25	8.0	44
θth-ECDF	69.17	94.60	91.40	22.31	12.0	16
PV-BC	63.89	98.51	98.44	29.04	14.1	21
RI-BC	54.64	99.13	99.31	43.16	15.9	14

Table 4: Comparison with three other methods for Type 2 distributions

	Efficiency (%)	Reliability (%)	Performance (%)	Overreaction (%)	Signal density	Average first time occurrence(%)
θth-ECDF	91.89	95.71	91.90	5.19	8.7	43
PV-BC	74.26	98.87	98.45	18.49	13.0	26
RI-BC	66.37	99.42	99.32	27.03	14.4	21

Table 5 and Table 6 give the results of the empirical experiment. As the type 1 and type 2 distributions yield similar results, we only report the results based on the type 2 distribution. In the first phase simulation of our method, the activity duration is sampled from distribution B_1, and the activity dependence is set at the medium level. At the same time, we also set the value of θ in θth-ECDF to let it produce a 95% reliability. From Table 5 we can observe that 28 projects do not meet the deadline, which accounts for 30% of the total instances in the experiment, while the other 70% projects finished on time. Results in Table 6 further support the conclusions that were obtained from the fictitious project experiment: although the θth-ECDF, PV-BC and RI-BC methods can produce a higher performance and signal density and can report a warning signal earlier than our method, they give too many false warning signals, thus rendering a very low efficiency.

Table 5: Statistics of the expermental results for different tolerance limits

	Late projects		Early projects	
	Signal	No signal	Signal	No signal
Type 2	24	4	5	60
θth-ECDF	25	3	18	47
PV-BC	28	0	20	45
RI-BC	28	0	26	39

Table 6: Performances of different tolerance limits for the real-life project data set

	Efficiency (%)	Reliability (%)	Performance (%)	Overreaction (%)	Signal density	Average first time occurrence(%)
Type 2	82.76	93.75	85.71	7.69	9.7	48
θth-ECDF	58.14	94.00	89.29	27.69	10.6	18.5
PV-BC	58.33	100.00	100.00	30.77	11.8	16.0
RI-BC	51.85	100.00	100.00	40.00	11.9	13.1

5.2 Sensitivity analysis

In this section, we conduct four sensitivity experiments to analyze the impacts of four factors on the performance of our schedule control method. The four influencing factors are: the standard deviation of the first-phase simulation, the level of activity dependence, the size of the project buffer, and the project network serial/parallel indicator SP values. As the projects in the real-life project data set show a very limited SP range, we only use fictitious projects in the sensitivity analysis. In the sensitivity experiment of project buffer, four different sizes of the project buffer (10%, 20%, 30% and 40% of the planned duration) are added after the planned duration of the project. In the critical chain project management (Goldratt, 1997) the project buffer size is set at 50% of the planned duration. However, in our experiment we cannot go that far, because for some project instances the percentage of having a longer project duration than the deadline is then less than 5% in the simulation, which is the significance level in our experiment, when the project buffer size is set at 50% of the planned duration. This means that we do not need to control the project schedule, because the significance level will never be exceeded. As the four factors have the same influence patterns on type 1 and type 2 distributions, we only report the results on the type 2 distributions.

Figure 6 displays the sensitivity results when considering the change of one factor in each subgraph. From Figure 6a, we can see that a small standard deviation of the first-phase simulation will cause a high performance and signal density but also a low efficiency. However, a higher standard deviation of the first-phase simulation has little impact on all metrics. The results from Figure 6b show that the tolerance limits become more discriminative when the activity dependence increases. This phenomenon can be easily interpreted: as the activity dependence rises, longer activity durations tend to cluster together, resulting in long project durations, while shorter activity durations combine with other shorter activity durations, resulting in short project durations. Thus it is easier for the tolerance limits to tell when the project is out of control. The project buffer size has a remarkable effect on performance and signal density as illustrated in Figure 6c. With an increase of the project buffer size, which causes the percentage of late projects to decrease in the simulation, it is less capable of detecting late projects using the tolerance limits. The influence pattern of SP values on the performance metrics is much like that of the project buffer size as shown in Figure 6d. When the project contains more parallel activities (SP is small), it is much easier to give a false warning signal, thus causing a low efficiency, while when the project contains more serial activities, the performance and signal density will decline. However, both of these impacts are only obvious when the SP values are less than 0.4. One common feature among all the subgraphs of Figure 6 is that the change of any influence factor has little impact on the reliability. Although our algorithm can only ensure the reliability of the tolerance limits to be $1-\alpha$ for the first-phase simulation, experimental results show that this reliability can also be ensured for the second-phase simulation under all conditions.

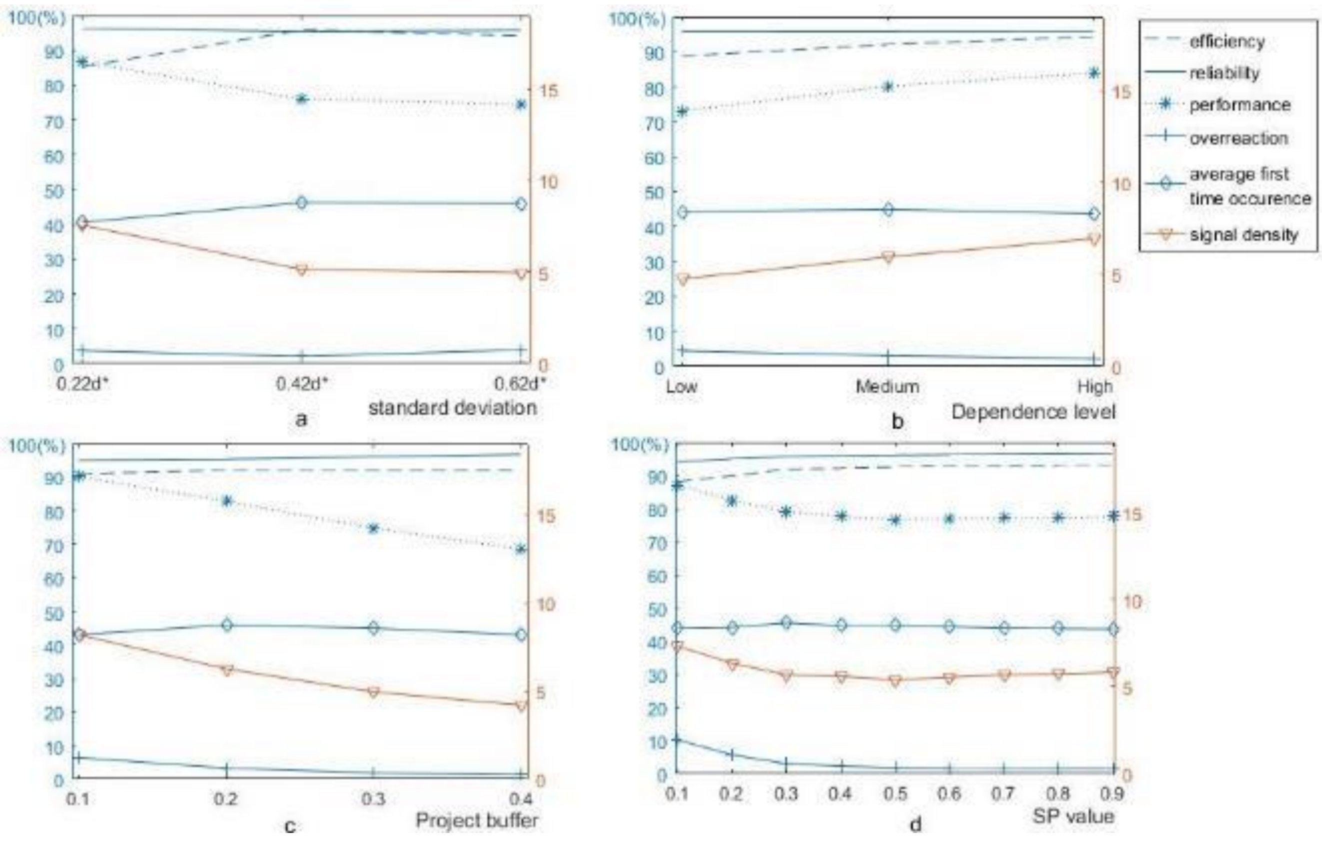

Figure 6 The impact of four influencing factors on the schedule control performance

In order to quantify individual and joint effects of the four influencing factors on the performance metrics, we use the variance-based global sensitivity analysis method extended fourier amplitude sensitivity testing (eFAST) (Saltelli, Tarantola, and Chan

1999) to calculate the sensitivities. Table 7 gives the first order sensitivity indices, which measure the contribution of the influencing factors to the output variance individually. Data in Table 7 suggest that the standard deviation of the first-phase simulation contributes most to the variance of the efficiency and the average first time occurrence, while the project buffer contributes most to the variance of the performance and the signal density. The SP value is the most important input for the variance of the reliability and the overreaction. For some performance metrics, the main variance comes from interactions among input variables. For example, the sums of the first order index for efficiency, reliability and overreaction are less than 50%, and for the average first time occurrence, the sum of the first order index is only 19.17%. Table 8 shows the total order sensitivity indices, which measure the contribution to the output variance of each influencing factor, including all variance caused by its interactions with any other input factors. Comparing Table 8 with Table 7, we find that when considering the total effect, the most important factor for each output remains the same as for the main effect except for signal density. For the signal density, the standard deviation of the first-phase simulation becomes the most influential parameter when considering the interactions with other variables. Besides, this parameter also comprises large sources of variance for reliability, performance and overreaction. In addition, the interaction effect of the SP value on the average first time occurrence is also considerable. The dependence level can be considered as non-influential parameters for all the outputs both for the first order and the total order sensitivity indexes.

Table 7: First order sensitivity indices

	Efficiency	Reliability	Performance	Overreaction	Signal density	Average first time occurrence
Standard deviation	0.3233	0.0533	0.1391	0.0289	0.1866	0.1483
Dependence level	0.0738	0.0037	0.0957	0.0308	0.1048	0.0053
Project buffer	0.0028	0.1521	0.3203	0.1246	0.3170	0.0285
SP value	0.0354	0.1943	0.0512	0.2460	0.0333	0.0096

Table 8: Total order sensitivity indices

	Efficiency	Reliability	Performance	Overreaction	Signal density	Average first time occurrence
Standard deviation	0.8578	0.6359	0.4616	0.4195	0.5056	0.6487
Dependence level	0.1600	0.1044	0.2061	0.2107	0.1515	0.3658
Project buffer	0.1326	0.2740	0.5274	0.4034	0.4168	0.4958
SP value	0.4493	0.7759	0.3196	0.7525	0.3147	0.6161

6. Discussion and conclusions

In this paper, we propose a new project schedule control method that considers the stochastic nature of a project in reality. We combine the advantages of the well-known statistical process control charts with the traditional earned value management/earned schedule approach in our method. Comparing with the classical SPC method, this approach is more flexible, and allows to monitor dependent and not normally distributed data. Thus it is more suitable for project schedule control. We view the project schedule control process as a statistical hypothesis test. Based on the simulation results, an algorithm is designed to optimize the tolerance limits. These refined tolerance limits can meet the significance level and meanwhile reduce the probability of the type Ⅱ error significantly. We apply our tolerance limits on a wide range of different project networks in a large computational experiment, and compare our approach with three other schedule control methods from the literature. Results show that our approach is more discriminative than others.

We have conducted sensitivity experiment to analyze the impact of the first-phase simulation standard deviation, the activity dependence, the project buffer size and the network SP values on the performance of the proposed method. Experimental results show that any change in one of the considered factors allows our tolerance limits to always meet the required significance level, i.e. the reliability keeps very stable at $1-\alpha$. An increase in the project buffer size will impair the performance of the tolerance limits. If the project manager sets a large project buffer size, which causes the probability of not finishing the project before the deadline to be very small, setting the significance level at 0.05 may cause a low performance and signal density when using our method. But this disadvantage can be avoided by setting a smaller significance level, for example 2%. Meanwhile, the serial/parallel topological indicator SP of the project network has a great influence on performance, signal density and overreaction, but this impact is only obvious when the SP value is less than 0.4. The more parallel the project network is, the more unnecessary warning signals the tolerance limits send. This trend is different from the results of the PV-BC analytical method in Martens and Vanhoucke (2017), where the increase of the SP values leads to an increase of the overreaction and a decrease of the efficiency. However, compared with our stabilized reliability, theirs changes from 93.91% to 99.69%, which means that the PV-BC method cannot meet the required significance level when applied on a parallel project network.

Future research could combine the earned value management/earned schedule with some other statistical process control charts, such as the cumulative sum (CUSUM) control charts and the exponentially weighted moving average (EWMA) control charts. As these control charts are more sensitive to a small and gradual drift in the process, they may give an earlier warning signal when the project has some structural and systemic changes over the life cycle. Another improvement could be the expansion of the rules for signaling the out of control state. Experimental results show that when the project network is close to being completely parallel, the overreaction will increase dramatically if one solely relies on the tolerance limits. So the development of some specific detection rules will reduce the overreaction and improve the efficiency for this occasion.

Acknowledgements

This work was supported by [National Natural Science Foundation of China] under Grant [number 71572148]; [Natural Science Basic Research Plan in Shaanxi Province of China] under Grant [number 2015JM7382]; [Social Science Foundation in Shaanxi Province of China] under Grant [number 2015R005]; and [Soft Science Research Plan in Shaanxi Province of China] under Grant [number 2015KRM039]. The authors would like to acknowledge the China Scholarship Council for the financial support and the Research Center for Operations Management of KU Leuven for providing a visiting research position for the lead author.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Acebes, F., Pajares, J., Galán, J. M., and López-Paredes, A. 2014. "A new approach for project control under uncertainty. Going back to the basics." *International Journal of Project Management*, 32(3), 423-434.

Acebes, F., Pereda, M., Poza, D., Pajares, J., and Galán, J. M. 2015. "Stochastic earned value analysis using Monte Carlo simulation and statistical learning techniques." *International Journal of Project Management*, 33(7), 1597-1609.

Aliverdi, R., Naeni, L. M., and Salehipour, A. 2013. "Monitoring project duration and cost in a construction project by applying statistical quality control charts." *International Journal of Project Management*, 31(3), 411-423.

Batselier, J. and Vanhoucke, M. 2015. "Construction and evaluation framework for a real-life project database." *International Journal of Project Management*, 33(3), 697-710.

Bauch, G. T., and Chung, C. A. 2001. "A statistical project control tool for engineering managers." *Project Management Journal*, 32(2), 37-44.

Colin, J., and Vanhoucke, M. 2014. "Setting tolerance limits for statistical project control using earned value management." *Omega*, 49, 107-122.

Colin, J., and Vanhoucke, M. 2015a. "A comparison of the performance of various project control methods using earned value management systems." *Expert Systems with Applications*, 42(6), 3159-3175.

Colin, J., and Vanhoucke, M. 2015b. "Developing a framework for statistical process control approaches in project management." *International Journal of Project Management*, 33(6), 1289-1300.

Demeulemeester, E., Vanhoucke, M., and Herroelen, W. 2003. "RanGen: A random network generator for activity-on-the-node networks." *Journal of scheduling*, 6(1), 17-38.

Goldratt, E. 1997. *Critical Chain. Great Barrington*, MA: North River Press.

Hu, X., Cui, N., and Demeulemeester, E. 2015. "Effective expediting to improve project due date and cost performance through buffer management." *International Journal of Production Research*, 53(5), 1460-1471.

Khamooshi, H., and Golafshani, H. 2014." EDM: Earned Duration Management, a new approach to schedule performance management and measurement." *International Journal of Project Management*, 32(6), 1019-1041.

Leu, S. S., and Lin, Y. C. 2008. "Project performance evaluation based on statistical process control techniques." *Journal of Construction Engineering and Management*, 134(10), 813-819.

Lipke, W. 2002. "A study of the normality of earned value management indicators." *The Measurable News*, 4(1), 1-16.

Lipke, W. 2003. "Schedule is different." *The Measurable News*, 31(4), 31-34.

Martens, A., and Vanhoucke, M. 2017. "A buffer control method for top-down project control." *European Journal of Operational Research*, 262(1), 274-286.

Pajares, J., and Lopez-Paredes, A. 2011. "An extension of the EVM analysis for project monitoring: The Cost Control Index and the Schedule Control Index." *International Journal of Project Management*, 29(5), 615-621.

Saltelli, A., Tarantola, S., and Chan, K. S. 1999. "A quantitative model-independent method for global sensitivity analysis of model output." *Technometrics*, 41(1), 39-56.

Trietsch, D., Mazmanyan, L., Gevorgyan, L., and Baker, K. R. 2012. "Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation." *European Journal of Operational Research*, 216(2), 386-396.

Vanhoucke, M., Coelho, J., Debels, D., Maenhout, B., and Tavares, L. V. 2008. "An evaluation of the adequacy of project network generators with systematically sampled networks." *European Journal of Operational Research*, 187(2), 511-524.

Vanhoucke, M., and Vandevoorde, S. 2007. "A simulation and evaluation of earned value metrics to forecast the project duration." *Journal of the Operational Research Society*, 58(10), 1361-1374.

Wang, Q., Jiang, N., Gou, L., Che, M., and Zhang, R. 2006. "Practical experiences of cost/schedule measure through earned value management and statistical process control." In: Wang, Q., Pfahl, D., Raffo, D.M., and Wernick, P. (Eds.), *Software Process Change* (pp. 348-354). Springer, Berlin, Heidelberg.