Incorporation of activity sensitivity measures into buffer management to manage project schedule risk

Xuejun Hu ^{a,*}, Nanfang Cui ^a, Erik Demeulemeester ^b, Li Bie ^c

^a *School of Management, HuaZhong University of Science and Technology, Wuhan, China*

^b *Department of Decision Sciences and Information Management, KU Leuven, Belgium*

^c *Management School, South-Central University For Nationalities, Wuhan, China*

Critical Chain Scheduling and Buffer Management (CC/BM) has shown to provide an effective approach for building robust project schedules and to offer a valuable control tool for coping with schedule variability. Yet, the current buffer monitoring mechanism faces a problem of neglecting the dynamic feature of the project execution and related activity information when taking corrective actions. The schedule risk analysis (SRA) method in a traditional PERT framework, on the other hand, provides important information about the relative activity criticality in relation to the project duration which can highlight management focus. It is implied, however, that control actions are independent from the current project schedule performance. This paper attempts to research these defects of both tracking methods and proposes a new project schedule monitoring framework by introducing the activity cruciality index as a trigger for effective expediting to be integrated into the buffer monitoring process. Furthermore, dynamic action threshold settings that depend on the project progress as well as the buffer penetration are presented and examined in order to exhibit a more accurate control system. Our computational experiment demonstrates the relative dominance of the integrated schedule monitoring methods compared to the predominant buffer management approach in generating better control actions with less effort and an increased tracking efficiency, especially when the increasing buffer trigger point is combined with decreasing sensitivity action threshold values.

Key words: Buffer management; Schedule monitoring; Activity sensitivity;

---

* Corresponding author. Tel.: +8615927657485
Email-address: huxuejun0626@hust.edu.cn (X. Hu).

1. Introduction

In today's highly competitive and rapidly changing marketplace, projects are increasing subject to a wide range of constraints, such as resource scarcity, network complexity and various uncertainties, demanding more effective techniques to improve the monitoring of the project schedule so as to ensure a successful scheduling outcome (Herroelen & Leus, 2005; Hu et al., 2015). Traditional project schedule management methods, namely the Critical Path Method (CPM) and the Program Evaluation and Review Technique (PERT), can no longer meet the diversified needs of modern projects. To this end, Goldratt (1997) applied the theory of constraints (TOC) to project management and proposed the Critical Chain Scheduling and Buffer Management (CC/BM) methodology, which has proven to be a popular and effective approach regarding both project scheduling and project control under enormous complexity and uncertainty (Yang et al., 2007; Bevilacqua et al., 2009; Ma et al., 2014; Peng & Huang, 2014; Yang & Fu, 2014; Zhang et al., 2015a, 2005b, among others).

CC/BM relies on deterministic scheduling techniques in order to build a resource feasible schedule that is made robust by inserting various types of buffers (*project buffer, feeding buffer, resource buffer*), and improves the schedule performance during project execution to meet the deadline through *buffer monitoring*. Consequently, the size of buffers and how they are monitored directly determine the project completion time as well as the schedule risk, and hence play crucial role in the successful application of CC/BM (Khemakhem & Chtourou, 2013). The vast majority of the research efforts on buffer management (BM) have concentrated on buffer sizing (see Tukel et al. (2006) and Bie et al. (2012) for good overviews). However, interest in buffer monitoring techniques, an issue of major importance in practice, is generally insufficient among the research community.

The existing buffer monitoring mechanism mainly follows the practice established by Goldratt (1997) for production operations (Umble & Umble, 2006; Kuo et al., 2009). That is, the buffer is divided into three regions (the “green–yellow–red” system) and explicit action levels for decision-making are set in terms of the buffer penetration along with the project

progress. Herroelen & Leus (2001) for the first time included an activity crashing mechanism in their factorial experiment based on the three-stage buffer control system. Leach (2005) questioned Goldratt's static monitoring notion by maintaining that buffer trigger points are set to make the tracking process in line with the actual execution of a project, thus minimizing false warning signals and ensuring needed actions. Therefore, he recommends the two buffer trigger lines (i.e. the green-to-yellow transition and the yellow-to-red transition) vary linearly over the planned project duration in order to more accurately monitor or respond to schedule deviations. We refer to this method as *relative buffer management approach (RBMA)*, which is currently widely used in practice and will be compared to our proposed methods. Bie & Cui (2010) presented a more realistic buffer monitoring method by dynamically calculating the buffer size and the time instant for monitoring, so that the two control trigger points are timely adjusted according to the dynamic environment of project implementation. In a more recent study, Hu et al. (2015) put forward a new CC/BM-based schedule monitoring procedure that evaluates the probability of successful project completion relative to the cost of crashing and that determines when to expedite which activity in a cost-effective manner. Zhang et al. (2015b) established an effort buffer deviation monitoring and control model for software projects based on the grey prediction model which proves to provide reliable control results in an empirical study. Colin & Vanhoucke (2015) cleverly combined the earned value management/earned schedule (EVM/ES) method with the concept of buffers inspired by CC/BM and proposed two new project control approaches with multiple control points for the purpose of minimizing the effort spent by the project manager.

In general, the buffer monitoring logic well indicates the project schedule progress as a whole and informs decision-making about whether to take control action or not. However, we find it has two problems that cannot be neglected in a real world application: (1) it does not provide the activity-level information through the buffer-related performance measures. Namely, the various impacts of individual activity duration variability on the project completion time are not discretely analyzed, which might lead to inaccurate warning signals and hence, inefficient control actions can be taken; (2) it fails to clarify which activities

deserve more of management control when drilling down to lower WBS (Work Breakdown Structure) levels to take corrective actions. To counter these practical issues, we introduce the activity sensitivity measure from the *schedule risk analysis* method that enables critical activities to be identified for control purposes. First, a new buffer threshold is defined that distinguishes the control effort into two aspects, namely "*take action/more attention*" and "*no action/less attention*". Then we explore how the activity-based tracking policy can be incorporated into the project-based buffer management system by the use of a (static) sensitivity action threshold. Next, two new ways of dynamically setting up the sensitivity action thresholds are proposed according to the portion of project completion or the penetration level of the buffers. Our computational studies show that, relative to the previous BM approach, the integration of the activity sensitivity measure is capable of improving the project schedule performance with comparatively less control effort, especially in the case where decreasing sensitivity action threshold values are used.

The outline of this paper is as follows. Section 2 briefly reviews relevant research related to the schedule risk analysis method and gives an introduction to its working principles. In Section 3, the primary BM system as well as the activity sensitivity based tracking system is described and then the integrated schedule risk management framework is proposed. Section 4 presents a simulation study to illustrate the application of the proposed methods and also to testify their superiority in comparison to an alternative BM approach. In the last section, we discuss the implications of this study and identify some future research directions.

2. Activity-based schedule risk analysis

Since the introduction of the well-known PERT analysis in project management, the *schedule risk analysis* (SRA, Hulett, 1996) method has been set up to implement schedule control principally on a subset of highly sensitive activities that are considered to have high schedule risk (i.e. contribute most to the delay of a project or have a high impact on project completion). Various sensitivity measures have been proposed in the literature to measure activity importance, so that the most targeted actions can be taken to effectively control the project duration (Williams, 1992; Elmaghraby, 2000; Fortin et al., 2010; Vanhoucke, 2010;

etc.). Table 1 lists the four basic sensitivity measures that have been widely acknowledged or investigated in previous studies. The literature has also presented other ranking indices for activity importance (see, for example, Cho and Yum. 1997; Bowman, 2003, 2006, 2007; Madadi & Iranmanesh, 2012; Creemers et al., 2014), which will not be further discussed due to the scope of this research.

Table 1.

The four basic activity sensitivity measures

Index		Explanation	Formulation
	d_i	the duration of activity i	
	tf_i	the total float of activity i	
	$C_{\max}$	the project duration/makespan	
Van Slyke (1963)	Criticality index, CI	It measures the probability that an activity falls on the critical path.	$CI_i = \Pr\{tf_i = 0\}$
Williams (1992)	Significance index, SI	It reflects the significance of individual activities on the project duration.	$SI_i = E\left(\frac{d_i}{d_i + tf_i} \times \frac{C_{\max}}{E(C_{\max})}\right)$
Williams (1992)	Cruciality index, CRI	It reflects the relative importance of an activity.	$CRI_i = \text{Corr}(d_i, C_{\max})$ $= \frac{\text{Cov}(d_i, C_{\max})}{\sqrt{\text{Var}(d_i) * \text{Var}(C_{\max})}}$
PMBOK (2008)	Schedule sensitivity index, SSI	It combines the activity duration and project duration sample standard deviations with the criticality index.	$SSI_i = \left[\sqrt{\frac{\text{Var}(d_i)}{\text{Var}(C_{\max})}}\right] * CI_i$

In general, this type of *activity-based* project tracking approach divides activities into two classes, namely *highly sensitive* activities and *insensitive* activities, by setting up an *action threshold* that defines the degree of management attention/control. All activities with a sensitivity value higher than or equal to the action threshold are said to be highly sensitive activities and thus they deserve more attention during the project tracking process and require corrective actions in case of delays (Vanhoucke, 2010). In the example case of Fig. 1 below, the sensitivity threshold has been set to 0.4 such that only the most sensitive activities 1, 4, 6

and 7 need to be considered when corrective actions are necessary.

Fig. 1 Action threshold based on activity sensitivity values

While the SRA method offers a great measure (activity sensitivity) to identify important/influential activities for control purposes, it faces a problem that action decisions are independent from the overall project schedule performance during the tracking process. To put it another way, corrective actions are required to be taken whenever highly sensitive activities experience delays regardless of the current project executing state (i.e. the degree of project catch-up or project delay), which seems **not** to be an appropriate practice in reality.

3. An integrated schedule monitoring model

From what has been illustrated, we can see that BM and SRA have both outstanding merits and intrinsic flaws, stimulating the research interest of this paper for a more systematic approach to improve the monitoring of the project schedule. In this section, we will first elaborate on the BM system and the CRI-based measurement system, respectively. Then an integrated schedule monitoring framework is presented. Throughout the paper, our primary focus is on the time performance of a project without regard to the cost measurement. Besides, the assumption, which is consistent with the past application of CC/BM, is made that one considers only one execution mode (one duration with the corresponding resource requirements) for each activity.

3.1. Buffer-related performance measurement system

In the single-project environment, the *critical chain* is defined as the longest path that not only considers the precedence relations but also takes into account the resource dependencies.

A *project buffer (PB)* is added to the end of the critical chain in order to protect due date performance from variations in the durations of the critical activities. *Feeding buffers (FBs)* are strategically placed whenever a non-critical task joins the critical chain in order to mitigate the duration variability of non-critical chain activities and to protect the critical chain. We might have to recompute the buffered schedule to obtain a precedence and resource feasible *baseline schedule* while maintaining the original critical chain. The project *due date* is then set equal to the planned project makespan, plus the PB size.

Accordingly, buffer monitoring compares the proportion of project completion with the degree to which the applicable buffer is penetrated and expedites if the buffer is eroding too quickly (Blackstone et al., 2009). Several buffer-related performance measures are summarized and listed along the following lines:

Buffer Size, BS

Buffer Consumed, BC=Estimated Activity Completion Time-Planned Activity Completion Time

Proportion of Buffer Consumed, PBC=BC/BS

Proportion of Chain Completed, PCC=Current Execution Time/Estimated Project Completion Time

---

Fig. 2 Buffer threshold settings in the RBMA method of Leach (2005)

Fig. 3 Activity Cruciality index CRI based buffer monitoring approach (CRI-BMA)

Unlike the common practice that the buffer is divided into three regions and two buffer trigger points are set to activate management actions (see Fig. 2), we just adopt one trigger point from a perspective of the critical chain, referred to as the *Relative Buffer Consumption Index RBCI*, which is defined by the following equation:

$$RBCI = \frac{PBC}{PCC} \quad (1)$$

The *RBCI* can be a reasonable measure since it well indicates the overall project

schedule performance by calculating the current project buffer consumption rate with respect to the progress made in finishing the critical chain. Specifically, if $RBCI$ is smaller than 1, it means that the remaining buffer can be enough to protect the variation of the unfinished activities, and that the project is less likely to delay, thus nothing needs to be done or less attention is required. Conversely, if $RBCI$ is greater than 1, it means that the project buffer is consumed at a faster rate compared to the progress of the critical chain, and there is a high probability that the project will suffer a delay, requiring more management attention for taking immediate corrective actions.

In order to fully repair schedule disruptions with minimum control effort, the management team may have to expedite some of the project activities. Nevertheless, it should be noted that crashing a certain activity may not contribute to the improvement of the project duration at all. This leads to several technical questions: (1) if two (or more) parallel tasks both experience a delay at some time point t with approximately the same feeding buffer penetration, which activity needs to be crashed more, or do we have to crash both activities? (2) if crashing the current activity is not enough to recover the buffer because of the maximum crash time limitation, the subsequent activity on the same chain then also needs to be expedited. Is there a reasonable ratio regarding the crash time for these two sequential tasks so that the project uncertainty can be furthest reduced? To answer these questions, we introduce the activity cruciality index CRI, which is expected to play a significant role in stimulating the management's focus when corrective actions are necessary. As a matter of fact, Herroelen & Leus (2001) have long ago addressed the need for an adequate evaluation of the criticality of project activities during the buffer control process, but they had not looked further into this important issue.

3.2. CRI-based measurement system

The reasons why we have chosen the CRI measure are twofold. First, Vanhoucke (2010) performs an SRA reliability study to investigate the effectiveness of the four basic sensitivity measures for controlling the project duration. His research shows that the CRI and SSI perform better than the CI and SI in providing efficient action guidelines. Second, the size of buffers to some extent reflects the uncertainty of the project duration, and CRI just measures the portion of total project duration uncertainty that can be explained by the duration

uncertainty of an individual activity (Williams, 1992). Both CRI and the buffer involve the idea of "*schedule risk*", as captured by the uncertainty/variability in the project completion time, making it reasonable as well as suitable to introduce CRI into our buffer monitoring model.

So back to the first issue raised in Section 3.1, we now know that the duration reduction of the activity with the highest CRI value has the highest beneficial effect on the project completion time. Therefore, it is only necessary to expedite one non-critical activity with a larger CRI value for a higher control efficiency as well as a lower impact on the critical chain. Regarding the second issue, it is better that the duration of the previous activity is shortened to its minimum crashed working time if its CRI is larger than that of the subsequent task; otherwise, the previous activity does not have to be expedited that much. So the recommendation is to compute the ratio of CRI values between the previous activity and the subsequent one, and to distribute the expediting efforts according to this ratio.

As to how to determine an appropriate CRI action threshold, denoted as α hereinafter, three strategies are presented and evaluated in the context of this work:

Strategy 1. The value of α is kept fixed throughout the life of a project, denoted as $\alpha = \hat{\alpha}$.

Strategy 2. Considering the dynamic feature of the project progress, the value of α decreases along the completion of project activities. Being aware that most activities are executed according to the scheduled start times in the preliminary phase and that actions on those sensitivity activities often have no or only a small effect on the project duration, the manager has reason to set high action thresholds in order not to trigger a lot of actions at first. In the later stages, however, timely control actions will often have a more significant impact on the project duration objective, demanding an increased management attention and thus considering more activities as highly sensitive (which is obtained by setting low action thresholds).

In this case, a linear equation based on the fixed threshold value, $\alpha = \hat{\alpha} - b * PCC (b > 0, PCC \in [0,1])$, is used to indicate the decrease of α as the project progresses, where PCC represents the proportion of the critical chain already performed at

each project review and $-b$ is the slope of the decreasing line.

Strategy 3. The value of α varies according to the penetration level of the buffers. More specifically, the bigger the $RBCI$ value (which implies a somewhat serious schedule disruption and that the remaining buffer might not be enough to resist risks), the lower the CRI action threshold should be (meaning that more sensitive activities should be subject to further control), and vice versa. Consequently, the former case exhibits a lower value of α whereas the latter case shows a higher value of α , differentiated by the transition point $RBCI = 1$.

3.3. Integration of CRI measure and buffer monitoring

Considering the complementarities between BM and SRA, we propose an integrated schedule monitoring framework that is referred to as the *activity cruciality index CRI based buffer monitoring approach (CRI-BMA)*, in order to gain a better understanding of the dynamic project schedule performance and to be able to make faster and more effective corrective decisions in real time. Specifically, project tracking is done on *the project level* using the buffer as an indicator to inform managers of the need/degree for action and on *the activity level* using the CRI index as a trigger to determine which activities to expedite.

The integrated control system can be divided into two parts, namely project-based buffer management and activity CRI-based expediting decision. On the one hand, managers keep track of buffer penetration with the completion of activities during project execution. By comparing the actual project implementation against the baseline schedule, one can easily calculate the $RBCI$ value and determine whether to take further control action or not (Hu et al., 2015). On the other hand, the CRI values for the unfinished project activities are dynamically calculated using the Monte Carlo Simulation (MCS) approach. Many scholars have explored the use of MCS in network simulation studies which provides a reliable decision basis for the project management team to enhance the actual execution of a project (for a more detailed discussion, we refer to Van Slyke (1963), Bowman (2006), Creemers et al. (2014), etc.). Then, the CRI action threshold shall be determined and a subset of highly sensitive activities recognized based on the individual CRI values, the present schedule progress, and the management subjective attitude, etc. If these highly sensitive activities are perceived to suffer a delay, appropriate control actions such as expediting/crashing, using

more resources, subcontracting, etc., should immediately be taken in order to effectively rebuild the project progress.

Fig. 3 shows how the buffer threshold and the CRI information can be used together to monitor project progress. A detailed description is further outlined in Fig. 4. As a matter of fact, our simulation model is expanded with an automatic *expediting framework* to simulate human intervention to bring late running projects partly back on track.

Fig. 4 Integrated schedule monitoring framework combining BM with SRA

Step 1. An initial schedule, which satisfies both the task sequence restrictions and the resource constraints, is constructed using the branch-and-bound algorithm of Demeulemeester & Herroelen (1992). Then we determine the critical chain/non-critical chains using the procedure of Tukel et al. (2006) and calculate relevant buffer sizes using the root square error method (RSEM, Newbold, 1998). For more information about critical chain scheduling, we refer to Rabbani et al. (2007),

Long & Ohsato (2008), or software (e.g., ProChain, Product Development Institute (PDI), Microsoft Project, etc.

Step 2. Initiate the project at $t = 0$.

Step 3. Increase the review period, $t = t + 1$. It is important to note that the execution of a CC/BM schedule uses the *roadrunner scheduling policy* (which is also named *relay race behavior* or *roadrunner mentality*, Tian & Demeulemeester, 2014), having gating tasks start no earlier than their baseline starting times and giving priority to activities on the critical chain.

Step 4. First, determine whether the current execution time is a *monitoring point* or not. The judgmental principles of a monitoring point can be summarized as follows: (1) an activity is completed; (2) a chain breaks off, or new activities are inserted; (3) a gating task is about to start. Or, management can implement a continuous monitoring policy by calculating the buffer trigger point dynamically on a real time basis or at the appropriate time intervals.

Next, check the buffer consumption and compute the *RBCI* index. Note that three action strategies are available as mentioned above, involving different ways of setting the CRI action thresholds. In the first two cases: (1) if *RBCI* is smaller than 1, turn to step 5; (2) if *RBCI* is greater than 1, meaning that there is a high probability that the project will delay, management should plan actions ahead of schedule in case the situation continues to deteriorate, thus turn to step 6. And the CRI action threshold value is set to either $\hat{\alpha}$ or $\hat{\alpha} - b * PCC$. For the third case, the CRI action threshold is inversely related to the value of *RBCI*.

Step 5. No management action is needed.

Step 6. It indicates that problems might arise with the progress of project execution, requiring project managers to inspect possible reasons, analyze potential risks and map out coping strategies, so as to prepare for the next step. Meanwhile, the CRI-based activity sensitivity measure will be adopted as an action guideline for further control (see step 7) and integrated into the buffer tracking process.

Step 7. A simulation-based estimator of CRI for each unfinished activity i is calculated as

follows:

$$\hat{CRI}_i = \frac{\sum_{k=1}^m (d_i^k - \bar{d}_i) * (C_{\max}^k - \bar{C}_{\max})}{(m-1) * S_{d_i} * S_{C_{\max}}} \quad (2)$$

with S_{d_i} and $S_{C_{\max}}$ the sample standard deviations of variables d_i and $C_{\max}$, m the number of the Monte Carlo simulation runs (superscript k refers to the d_i or $C_{\max}$ of simulation run k). The simulated execution of the remaining project uses the parallel *schedule generation scheme (SGS)* with the *earliest starting time (EST)* priority rule. The CRI action threshold is determined as indicated in step 4 (which will not be reiterated here).

Next, identify activities in progress or the successors and compare their CRI values with the CRI action threshold—if the action threshold is exceeded and those highly sensitive activities show a delay, turn to step 8 to take immediate action, otherwise turn back to step 3.

Step 8. The specific corrective action is taken that consists of a duration reduction of some selected activities following the predefined expediting rule in Section 3.2.

It should be pointed out that the proposed schedule monitoring process combines two types of information for controlling the project duration. The idea is that when management has general knowledge about the current project schedule performance (dynamically obtained through BM during project tracking) and has certain information on the sensitivity of individual project activities (obtained through simulation-based SRA considering variability), a more reliable action response and a better management focus can be derived to improve monitoring of the project schedule as well as to effectively control the effort a project manager should put in the tracking process.

4. Simulation study

In this section, we provide the results of an extensive simulation study performed to show the capability of the proposed schedule monitoring methods. Our computational experimentation was conducted using the 30-activity instances of the well-known PSPLIB

data set (Kolisch & Sprecher, 1996) as well as Patterson's data set (Patterson, 1984), and the program was run on a personal computer using the MATLAB language.

4.1 Experimental layout

Since we assume the activity durations to come from a right skewed distribution, we use a *lognormal distribution* to generate random task durations, allowing us to simulate projects with varying levels of uncertainty (represented by the *coefficient of variation*, cv) while maintaining the same mean (baseline) durations. A lognormal distribution is also used by Herroelen & Leus (2001), Bie et al. (2012) and Hu et al. (2015).

In our simulation, the buffer consumption will be checked by the project manager every time an activity is completed (i.e., monitoring point, or project review) by comparing the actual implementation against the baseline schedule. If the project buffer is consumed to the specified limit (i.e. the buffer threshold), the successive critical chain activities are made eligible for expediting; if the feeding buffer consumption penetrates into the red region defined by Goldratt (i.e., between 66 percent and 100 percent of the feeding buffer size), the activities eligible for crashing are the unfinished non-critical tasks that are on the same feeding path. Furthermore, each time a feeding buffer is fully consumed, the subsequent critical activity is made eligible for crashing. In the CRI-BMA model, activity crashing is actually implemented considering both the buffer threshold and the CRI information (see Fig. 3). We assume that the decrease in the activity durations is positively correlated with the CRI values, that is, high risk activities are crashed more. Vanhoucke (2011) also implemented this kind of improvement action to simulate a human intervention. To be more specific, each activity is subjected to a maximum crash time, which equals its original duration multiplied by the sensitivity measure. So the crash time for activity i can be generated from a uniform distribution over $[0, d_i * CRI_i]$. In the RBMA model, however, there is no sensitivity distinction among activities when it comes to corrective actions. We assume the maximum crash time for each activity to equal half of its original duration from a practical perspective. So the crash time for activity i in the RBMA method can be generated from a uniform distribution over $[0, d_i * 0.5]$.

At the finish of each project, the tracking efficiency is measured which consists of a

comparison between the time and effort the project manager has spent, and the effect of corrective actions on the project time performance. For that purpose, the following performance indicators are defined:

- *Number of crashing activities, CA*. This is calculated as the sum of all activities that need correction actions. This indicator measures the frequency with which management will act during project tracking. A larger CA means more activities are subject to corrective actions, increasing the complexity of project management as well as the probability of schedule changes.
- *Crashing time, CT*. This is calculated as the sum of the crashing times of all activities in one project. A longer CT normally means higher expediting costs, with no regard to the crashing cost per unit time of each activity in this article.
- *Timely project completion probability, TPCP*. This indicator refers to the probability that a project is completed within the projected deadline (due date).
- *Real project duration, RPD*. This indicator refers to the simulated project completion time with or without threshold triggered corrective actions.

For each project instance and for each value of cv , 1000 simulation iterations ($m=1000$) were generated to calculate the average performance. One thousand repetitions were deemed adequate since no significant differences were found when a larger number of simulation runs were tested.

4.2. Test results and analysis

To illustrate how to reasonably set a fixed CRI action threshold value (**Strategy 1**), we first conduct **Experiment 1** (see 4.2.1) in which a single project example is implemented and various action thresholds are tested. Then **Experiment 2** (see 4.2.2) is conducted to make a performance comparison among different schedule monitoring methods using multiple project instances of the two date sets.

4.2.1. Effect of CRI action threshold

The simple project we selected consists of 11 activities, and only one kind of resource is needed with a constant availability of 7 units. The project network, represented in activity-on-the-node representation and zero-lag finish-start precedence constraints, is given

in Fig. 5. The first number above each node represents the mean (50 percent estimate) duration of each activity, which is assumed to be half of the safe estimate (90 percent) as was described in most literature (see, e.g. Leach, 2005; Tukul et al., 2006; Bie et al., 2012; etc.), and the second number represents the per period requirement of each activity for a single renewable resource. Fig. 6 shows the result of critical chain scheduling and buffer insertion. Before the actual execution of the project, the CRI value is calculated for each activity (see Table 2) according to Eq. (2) with cv being generated from a uniform distribution over $[0.4,1.2]$ in each simulation run and $m=1000$. Although we highly recommend the management team to dynamically update the activity CRI values at each project review, we have not done so in our simulation study in order to save computational time, and for simplicity.

Fig.5 A simple project network

Fig. 6 CC/BM schedule for the example network

Table 2

The CRI value of each activity for the project network of Fig. 5.

Activity	1	2	3	4	5	6	7	8	9	10	11	Median
CRI	0.501	0.118	0.011	0.181	0.340	0.075	0.207	-0.055	0.483	0.497	0.155	0.181

Next, eleven threshold levels from 0 to 0.5 with an interval of 0.05 based on Table 2 were tested to illustrate the effect of the CRI action threshold on the performance of the proposed CRI-BMA model. Three levels of cv (i.e. $cv \in [0.4, 0.8, 1.2]$) were used, representing a project uncertainty that is Low (L), Medium (M) or High (H), respectively. Table 3 shows the results, and Fig. 7 visually presents the change trends for each performance indicator.

Table 3

Effect of the CRI action threshold on the performance of CRI-BMA under three uncertainty levels

α	CA			CT			TPCP			RPD		
	L	M	H	L	M	H	L	M	H	L	M	H
0	0.780	1.120	0.947	1.276	1.988	1.743	0.958	0.874	0.865	51.112	50.919	49.540
0.05	0.781	1.117	0.940	1.313	2.013	1.738	0.956	0.876	0.864	51.082	50.909	49.559
0.1	0.698	0.978	0.844	1.258	1.881	1.569	0.955	0.874	0.860	51.073	50.964	49.664
0.15	0.644	0.905	0.787	1.202	1.849	1.612	0.957	0.869	0.861	51.114	50.966	49.630
0.2	0.595	0.835	0.719	1.189	1.783	1.541	0.948	0.868	0.856	51.129	51.038	49.700
0.25	0.478	0.669	0.562	1.082	1.584	1.382	0.944	0.863	0.851	51.242	51.216	49.845
0.3	0.484	0.676	0.562	1.050	1.614	1.382	0.943	0.864	0.852	51.269	51.183	49.846
0.35	0.303	0.470	0.418	0.800	1.274	1.159	0.943	0.858	0.850	51.492	51.462	50.021
0.4	0.306	0.470	0.414	0.810	1.308	1.139	0.936	0.857	0.846	51.486	51.449	50.047
0.45	0.306	0.469	0.421	0.845	1.280	1.145	0.941	0.856	0.848	51.446	51.476	50.035
0.5	0	0	0	0	0	0	0.910	0.812	0.809	52.289	52.688	51.127

Fig. 7 Effect of the CRI action threshold on the performance of CRI-BMA

It can be seen from the above results that when the CRI action threshold is set to a lower value, meaning that more activities are considered as adequately sensitive to be subject to expediting actions in case they are in trouble, the management will have to react more frequently, thus the values of CA and CT are relatively larger. Conversely, a higher action threshold obviously results in a smaller selection of activities subject to control actions, thus less management focus is needed during project tracking. This leads to a lower crashing time, and hence a lower on-time probability. Specially, a threshold value of zero makes all activities on the chain highly sensitive. In this case, the number of activities that need to be expedited is the largest, which undoubtedly aggravates the difficulty of management control. The remark has to be made that multiple project instances have been tested and have shown a similar changing trend as in Fig. 7.

Generally speaking, a better time performance demands shorter project durations with higher on-time completion probabilities. Meanwhile, fewer crashing activities and lower expediting time signify less management control efforts during project tracking. To achieve a balanced project performance, it is better that the CRI action threshold is set to a value somewhere near the median of all the activity CRI values (i.e. $\hat{\alpha} = 0.2$ for this example). In another aspect, the median setting is adopted for the convenience of large-scale simulations in the subsequent section, so that fifty percent of project activities are regarded as highly sensitive and the subject of further control in a static case. In practice, of course, the manager can exploit other threshold values based on the importance of the project or the management attitude towards risk.

4.2.2. Performance comparison of various methods

In this section, the three ways the CRI action thresholds are set will be compared with each other as well as with the RBMA method on the basis of **Experiment 1**, that is, the fixed CRI action threshold value $\hat{\alpha}$ has been set as the median of the CRI values of all activities. Table 4 provides the detailed threshold settings for these tracking methods in our comparison experiment. Note that the slope (± 0.2) is adequately set based on the experimental tests. Other possible values have been tested and have shown no significant differences in the

conclusions. Simulation results for the RBMA and the static CRI-BMA method on the two data-sets are shown in Table 5 and Table 6, respectively. The results without any corrective actions are also listed for further comparison.

Table 4.

Description of various methods that are to be tested

Methods	Buffer threshold for action	CRI action threshold
CRI-BMA	$\_1$ $RBCI > 1$	Static: $\alpha = \hat{\alpha}(median)$
	$\_2$	Decreasing: $\alpha = \hat{\alpha} - 0.2 * PCC$
	$\_3$ Differentiated by $RBCI = 1$	RBCI-related: $\alpha = \hat{\alpha} + 0.2 * (1 - RBCI)$
RBMA	$PBC > 0.3 + 0.6 * PCC$ (Leach, 2005)	—

Table 5

Performance comparison of different control strategies on PSPLIB date set

Performance indicator	<i>L</i>			<i>M</i>			<i>H</i>		
	CRI-BMA_1	RBMA	No action	CRI-BMA_1	RBMA	No action	CRI-BMA_1	RBMA	No action
CA	0.671	0.955		0.932	1.721		0.868	1.899	
CT	1.131	1.925		1.537	3.532		1.442	3.903	
TPCP	0.998	0.999	0.975	0.956	0.967	0.918	0.931	0.945	0.896
RPD	62.651	62.122	65.446	65.759	64.723	69.011	65.618	64.543	70.861

Table 6

Performance comparison of different control strategies on Patterson's date set

Performance indicator	<i>L</i>			<i>M</i>			<i>H</i>		
	CRI-BMA_1	RBMA	No action	CRI-BMA_1	RBMA	No action	CRI-BMA_1	RBMA	No action
CA	0.763	1.197		0.945	1.827		0.840	1.908	
CT	1.164	1.733		1.351	2.669		1.163	2.792	
TPCP	0.990	0.992	0.970	0.945	0.935	0.891	0.921	0.933	0.857
RPD	42.790	42.350	44.604	44.032	43.365	47.131	43.385	42.740	49.427

First of all, as uncertainty in the project environment increases the on-time completion rate TPCP declines as predicated in all cases. It's obvious that the higher the uncertainty level, the more risks the project will be faced with, inevitably leading to a poorer completion performance. Once control actions are taken, however, the project completion time RPD increases first and then decreases with the increase of cv . Timely actions appear to be more efficient regarding the project duration in case of higher uncertainty since there is a higher probability that project activities are subject to real delays, and hence corrective actions will often have a more significant impact on the duration objective. Besides, both action strategies achieve higher TPCPs as well as shorter RPDs compared with that of the case where no corrective actions are involved under three uncertainty levels, indicating that buffer control can be quite effective in protecting the project due date.

Secondly, for the RBMA method both the number of expedited activities and the crashing time increase with the value of cv . As we know, the project is more likely to encounter delays under a higher uncertainty, which incurs a higher frequency for the manager to make changes to the original plan according to the buffer consumption. So there will be more activities subject to corrective actions and a higher level of control effort is needed as the uncertainty increases. When it comes to the CRI-BMA method, on the other hand, CA and CT first increase and then decrease with the increase of cv , exhibiting a somewhat illogical

change trend at first sight that is different from the RBMA method. But when we get down to the details, one can see that the sensitivity measure is the fundamental reason contributing to this phenomenon. As has been pointed out in Section 2, the trigger to take corrective actions in the activity CRI-based tracking approach depends upon the individual activity sensitivity values. In the case where the uncertainty is not very high (L and M), most sensitive activities are executed according to the plan without a remarkable impact on the project deadline, and there is not much need for control actions to bring the project back on track. Hence, those triggered actions often result in an increased control effort. As the size of activity delays increases to a higher uncertainty (H), the CRI-based tracking system now exhibits more realistic action signals due to an increasing probability that problems now have appreciable impact on the project schedule. Hence, actions triggered in case of higher uncertainty based on the CRI measure can reduce the schedule risk to a greater extent and eventually lead to an increase in the tracking efficiency (i.e. a decrease in CA and CT).

Tables 5 and 6 also show that the CRI-BMA method is significantly more beneficial in terms of CA and CT compared to the RBMA under all uncertainty levels, and the performance gap between the two control logics becomes more obvious for increasing values of cv . On the other hand, both tracking methods generate approximately the same TPCPs and slightly different project durations although the CRI-BMA approach are slightly inferior to the RBMA especially in a project environment where uncertainty is very high. The reason can be explained as follows. The buffer threshold of the CRI-BMA control system is generally lower than that of the RBMA (the yellow-to-red transition) as can be seen from Fig. 2 and Fig. 3. Meanwhile the task durations can fluctuate more greatly for larger values of cv and the buffers will be consumed much faster. As such, some unnecessary corrective actions might be prematurely as well as more frequently triggered in the CRI-BMA method, disproportionately contributing to the project completion time in spite of the recognition of highly sensitive activities when the uncertainty is high; while in the RBMA method, corrective actions shall be taken every time the buffer consumption goes beyond the yellow-to-red threshold so that the RPD and TPCP are better protected. However, the project manager should be careful about the interpretation of this result since the RBMA method is obtaining a comparatively shorter project duration and higher on-time completion rate at the

cost of much more crashing activities and crash time. To be more specific, the average crash time for the CRI-BMA is far less than the project duration reduction between “no action” and the tracking method itself, which demonstrates that a far less control effort is sufficient to incur a much more beneficial control result due to the introduction of the CRI sensitivity index. Take for example the high uncertainty case (H) in Table 5, we have $1.442 \ll (70.861 - 65.618)$.

Table 7

Dynamic versus static CRI action threshold settings on PSPLIB date set

Performance indicator	<i>L</i>			<i>M</i>			<i>H</i>		
	Static	Decreasing	RBCI-related	Static	Decreasing	RBCI-related	Static	Decreasing	RBCI-related
CA	0.671	0.524	0.607	0.932	0.727	0.859	0.868	0.674	0.800
CT	1.131	0.855	1.004	1.537	1.159	1.307	1.442	1.075	1.183
TPCP	0.998	0.998	0.998	0.956	0.955	0.957	0.931	0.929	0.931
RPD	62.651	62.657	62.620	65.759	65.849	65.752	65.618	65.750	65.640

Table 8

Dynamic versus static CRI action threshold settings on Patterson's date set

Performance indicator	<i>L</i>			<i>M</i>			<i>H</i>		
	Static	Decreasing	RBCI-related	Static	Decreasing	RBCI-related	Static	Decreasing	RBCI-related
CA	0.763	0.659	0.777	0.945	0.782	0.942	0.840	0.683	0.827
CT	1.164	0.898	1.117	1.351	1.051	1.312	1.163	0.916	1.136
TPCP	0.990	0.990	0.990	0.935	0.931	0.935	0.911	0.906	0.910
RPD	42.790	42.809	42.748	44.032	44.112	43.998	43.385	43.484	43.367

The results for dynamic versus static CRI action threshold simulation runs are listed in Tables 7 and 8. The conclusions can be summarized along the following lines. First, the static case is outperformed in nearly all aspects by the two dynamic ways of setting up the CRI

action threshold, which indicates that a variable action threshold that relates to the dynamic project performance is capable of generating better control actions with less control effort. Second, the integrated schedule monitoring system is most efficient in combination with decreasing CRI action threshold values. Note that BM implements a *dynamically increasing buffer threshold* that varies linearly over the planned duration of the project, which means that the early project stages are crucial and deserve more attention than the late project stages. Also, managers should not use up the project buffer too early (Leach, 2005). The decreasing CRI action threshold setting, on the other hand, assumes a rather weak management attention at the beginning and then an incremental tracking effort because of an increasing probability that the sensitive activities are now subject to real delays in these later stages. This variation trend also confirms the common sense that the management should intensify the monitoring of tasks with the progress of the project. These two monitoring policies complement each other very well, which should positively contribute to the overall performance of a project. Third, the RBCI-related CRI-BMA method is not as useful as the decreasing procedure since the former allows for actions to be taken on a fraction of highly sensitive activities even if the buffer consumption is in a safe condition. On the one hand it ensures a better time performance; but on the other hand, it might trigger some unnecessary/ineffective expediting actions when the project is not really in big trouble.

4.3. Summary of computational results

As illustrated above in Section 4.2, we can come to the conclusion that the proposed CRI-BMA schedule monitoring methods outperform the relative buffer management approach. Due to the introduction of the activity cruciality index CRI, which helps to highlight management focus when corrective actions are necessary, the integrated framework reduces the project manager's control effort (i.e. the crash time and the expediting frequency) to a great extent, while obtaining almost the same results regarding the final project duration and maintaining a relatively high on-time completion probability, compared to the RBMA. More precisely, the buffer consumption provides a reliable warning signal with respect to the project's overall schedule performance. Meanwhile, the CRI provides a certain degree of confidence to the schedule estimates and locally reflects the specific schedule risk probability in relation to individual activities. Hence, a small reduction in the durations of highly

sensitive activities can lead to significant project schedule performance improvements, undoubtedly making the tracking process more effective. In the RBMA, however, there is no such distinction on activity criticality. Any buffer consumption that penetrates into the red zone will demand for the crashing of current or future activities, whereas it might be the fact that the duration reduction of some unimportant activities only has a minor impact on the project objective.

Indeed, the buffer-related measurement system indicates the time point of management control during the project's progress, i.e. when corrective actions are necessary to bring the project back on track; the CRI-based activity sensitivity measure, on the other hand, provides a general direction indicator where the focus of a project manager should be. By the combination of the buffer trigger point and the CRI action threshold, both of which act as warning signals, we can restrict management attention to the smallest possible activity subset which can lead to the best possible results, while still being able to provide a reliable response to the overall project schedule performance as well as to make more effective expediting decisions in terms of the relative importance of individual activities. What's more, a variable CRI action threshold that depends on the portion of work already performed or the consumption of buffers can further improve the tracking efficiency in comparison to the static threshold setting.

5. Conclusions

Buffer management, as the main content of the emerging CC/BM methodology, serves as an ideal control tool to monitor schedule deviations and to guide action during the project execution phase. This article aims at the existing problem of failing to provide more detailed activity information for management action as indicated by the buffer-related warning signal, and presents an integrated schedule monitoring system by introducing the activity cruciality index as a trigger for further control. Our contribution is threefold: (1) The respective advantages and disadvantages of two schedule control systems, namely the project-based BM and the activity-based SRA, are analyzed and their working principles described in detail. (2) We develop a strategic expediting procedure considering activity duration uncertainty that allows for dynamic re-evaluations of the schedule performance over time and that enables the

recognition of schedule-sensitive activities in particular to support more accurate action decisions. (3) We propose two new ways of dynamically setting up the sensitivity action thresholds, one relating with the project completion and the other with the relative buffer consumption, which have been shown to further promote the control precision and the tracking performance of a CC/BM schedule.

While the initial findings are promising, further research is necessary. First, the results obtained in this article are merely retrieved from computational simulations, imposing restrictions on broader management conclusions. However, the study should be relevant to practitioners since it provides useful and practical insights into the project schedule monitoring mechanism based on the combination of CC/BM with the activity criticality. To apply the proposed method to real projects to further testify its effectiveness as well as to explore some of the problems in its practical use will be the first extension. Furthermore, the CRI measure could be embedded into some existing CC/BM software to automatically generate specific action guidelines. Second, cost measurement, for example the crashing cost per unit time for individual tasks, the project completion time penalty and the budget constraint, etc., will be included to exhibit a more comprehensive buffer control system. A third significant area of research would be to consider buffer monitoring problems in a multi-project CC/BM environment. In this case, various priority rules for activity expediting, be it project-related, activity-related or resource-related, might be involved when setting up or triggering appropriate buffer thresholds. Research efforts related to these aspects are hopefully to be seen in our future study.

Acknowledgments:

This research was supported by the National Natural Science Foundation of China (Grant No. 71271097 and Grant No. 71201119). The authors would like to thank the editors and the anonymous reviewers for their invaluable comments and suggestions, which help to improve this paper. We would also 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 period to Xuejun Hu.

References

- Bevilacqua, M., Ciarapica, F.E., Giacchetta, G., 2009. Critical chain and risk analysis applied to high-risk industry maintenance: A case study. *International Journal of Project Management* 27 (4), 419-432.
- Bie, L., Cui, N., 2010. Research on dynamic buffer monitoring in critical chain project management. *Chinese Journal of Management Science* 18 (6), 97-103.
- Bie, L., Cui, N., Zhang, X., 2012. Buffer sizing approach with dependence assumption between activities in critical chain scheduling. *International Journal of Production Research* 50 (24), 7343-7356.
- Blackstone, Jr.J.H., Cox III, J.F., Schleier, Jr.J.G., 2009. A tutorial on project management from a theory of constraints perspective. *International Journal of Production Research* 47 (24), 7029-7046.
- Bowman, R.A., 2003. Sensitivity curves for effective project management. *Naval Research Logistics* 50 (5), 481-497.
- Bowman, R.A., 2006. Developing activity duration specification limits for effective project control. *European Journal of Operational Research* 174 (2), 585-593.
- Bowman, R.A., 2007. Efficient sensitivity analysis of PERT network performance measures to significant changes in activity time parameters. *Journal of the Operational Research Society* 58 (10), 1354-1360.
- Cho, J.G., & Yum, B.J., 1997. An uncertainty importance measure of activities in PERT networks. *International Journal of Production Research* 35, 2737-2758.
- Colin, J., Vanhoucke, M., 2015. A comparison of the performance of various project control methods using earned value management systems. *Expert Systems with Applications* 42 (6), 3159-3175.
- Creemers, S., Demeulemeester, E., Van de Vonder, S., 2014. A New Approach for Quantitative Risk Analysis. *Annals of Operations Research* 213, 27-65.
- Demeulemeester, E., Herroelen, W., 1992. A branch-and-bound procedure for the multiple resource-constrained project scheduling problem. *Management Science* 38 (12), 1803-1818.
- Elmaghraby, S.E., 2000. On criticality and sensitivity in activity networks. *European Journal of Operational Research* 127(2), 220-238.
- Fortin, J., Zieliński, P., Dubois, D., Fargier, H., 2010. Criticality analysis of activity networks under interval uncertainty. *Journal of Scheduling* 13(6), 609-627.
- Goldratt, E.M., 1997. *Critical Chain*. New York: The North River Press.
- Herroelen, W., Leus, R., 2001. On the merits and pitfalls of critical chain scheduling. *Journal of Operations Management* 19 (5), 559-577.
- Herroelen, W., Leus, R., 2005. Project scheduling under uncertainty: Survey and research potentials. *European journal of operational research* 165 (2), 289-306.

- Hu, X., Cui, N., 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.
- Hulett, D.T., 1996. *Schedule risk analysis simplified*. Project Management Network.
- Khemakhem, M. A., Chtourou, H., 2013. Efficient robustness measures for the resource-constrained project scheduling problem. *International Journal of Industrial and Systems Engineering* 14(2), 245-267.
- Kolisch, R., Sprecher, A., 1996. PSPLIB - A project scheduling problem library. *European Journal of Operational Research* 96, 205–216.
- Kuo, T.C., Chang, S.H., Huang, S.N., 2009. Due-date performance improvement using TOC's aggregated time buffer method at a wafer fabrication factory. *Expert Systems with Applications* 36 (2), 1783–1792.
- Leach, L.P., 2005. *Critical Chain Project Management* (2th ed.). Artech House.
- Long, L. D., Ohsato, A., 2008. Fuzzy critical chain method for project scheduling under resource constraints and uncertainty. *International Journal of Project Management* 26 (6), 688–698.
- Ma, G., Wang, A., Li, N., Gu, L., Ai, Q., 2014. Improved critical chain project management framework for scheduling construction projects. *Journal of Construction Engineering and Management* 140 (12), 1-12.
- Madadi, M., Iranmanesh, H., 2012. A management oriented approach to reduce a project duration and its risk (variability). *European Journal of Operational Research* 219, 751–761.
- Newbold, R. C. 1998. *Project management in the fast lane - Applying the theory of constraints*. Boca Raton: The St.Lucie Press.
- Patterson, J.H. (1984). A comparison of exact procedures for solving the multiple constrained resource, project scheduling problem. *Management Science* 30, 854–867.
- Peng, W., Huang, M., 2014. A critical chain project scheduling method based on a differential evolution algorithm. *International Journal of Production Research* 52 (13), 3940–3949.
- Rabbani, M., Fatemi Ghomi, S.M.T., Jolai, F., Lahiji, N.S., 2007. A new heuristic for resource-constrained project scheduling in stochastic networks using critical chain concept. *European Journal of Operational Research* 176 (2), 794–808.
- The Project Management Institute, 2008. *Guide to the project management body of knowledge (PMBOK guide)*. Newtown Square: The Project Management Institute.
- Tian, W., Demeulemeester, E., 2014. Railway scheduling reduces the expected project makespan over roadrunner scheduling in a multi-mode project scheduling environment. *Annals of Operations Research* 213 (1), 271-291.
- Tukel, O.I., Rom, W.O., Eksioglu, S.D., 2006. An investigation of buffer sizing techniques in critical chain scheduling. *European Journal of Operational Research* 172 (2), 401–416.

- Umble, M., Umble, E., 2006. Utilizing buffer management to improve performance in a healthcare environment. *European Journal of Operational Research* 174, 1060–1075.
- Vanhoucke, M., 2010. Using activity sensitivity and network topology information to monitor project time performance. *Omega--International Journal of Management Science* 38 (5), 359–370.
- Vanhoucke, M., 2011. On the dynamic use of project performance and schedule risk information during project tracking. *Omega--International Journal of Management Science* 39 (4), 416–426.
- Van Slyke, R.M., 1963. Monte Carlo methods and the PERT problem. *Operations Research* 11 (5), 839–860.
- Williams, T.M., 1992. Criticality in stochastic networks. *Journal of the Operational Research Society* 43 (4), 353–357.
- Yang J. B., 2007. How the critical chain scheduling method is working for construction. *Cost Engineering* 49 (4), 25–32.
- Yang, S., Fu, L., 2014. Critical chain and evidence reasoning applied to multi-project resource schedule in automobile R&D process. *International Journal of Project Management* 32 (1), 166–177.
- Zhang, J., Song, X., Chen, H., Shi, R., 2015a. Optimisation of critical chain sequencing based on activities' information flow interactions. *International Journal of Production Research*, DOI: <http://dx.doi.org/10.1080/00207543.2015.1043157>
- Zhang, J., Shi, R., Diaz, E., 2015b. Dynamic monitoring and control of software project effort based on an effort buffer. *Journal of the Operational Research Society*, DOI: 10.1057/jors.2014.125.