Buffer Sizing in Critical Chain Project Management with Network Decomposition

Bingling She[1], Bo Chen[1] and Nicholas G. Hall[2]

[1] Warwick Business School, The University of Warwick
b.she@warwick.ac.uk, b.chen@warwick.ac.uk
[2] Fisher College of Business, The Ohio State University
hall.33@osu.edu

Keywords: critical chain project management, buffer sizing, network decomposition

1 Introduction

Approximately 30% of global economic activity is organized using project management, which implies an annual value of about \$27 trillion (Hu et al., 2015; Zhao et al., 2020). An important methodological development in project management is critical chain project management (Goldratt, 1997), or CCPM for short. The critical chain he defines generalizes the critical path (Kelley and Walker, 1959) used in traditional project planning, by incorporating the issue of resource availability. The critical chain is protected by three types of buffer: a project buffer, one or more feeding buffers, and one or more resource buffers. However, buffers that are too small result in replanning and expensive emergency procedures to avoid late delivery of the project. Whereas, buffers that are too large result in uncompetitive bidding for projects and loss of potentially valuable contracts. Hence, accurate buffer sizing is essential to the economic success of project companies.

Previous buffer sizing research, focused predominantly on the critical chain, typically results in excessive buffer sizing, and critical chains being challenged by feeding buffers during planning, as well as inconsistent performance in, e.g., makespan estimation.

We propose a new procedure for buffer sizing through analytical decomposition of the project network, which offers logical advantages over previous ones. The buffers are determined based on uncertainties of all associated chains and comparisons between parallel critical and noncritical parts. Our work also addresses the concerns in the literature that there is no systematic analysis of the project network structure and its relationship with buffers. In addition, we resolve the problem of a challenged critical chain, while simultaneously addressing issues with multiple critical chains. Computational testing on a case study of a real project and extensive simulated data shows that our procedure delivers much greater accuracy in estimating project makespan, and smaller feeding buffers, while the resulting critical chain is never challenged. Additional benefits include delayed expenditure, and reductions in work-in-process, rework, and multitasking.

2 Problem Description and Buffer Sizing Procedure

We consider a project consisting of a task set $V = \{1, 2, \ldots, n\}$, a set $E \subseteq V \times V$ of precedence relationships between tasks, and a renewable resource set $R = \{1, 2, \ldots, m\}$. Tasks 1 and n are dummy, with no duration or resource requirement, representing the start and end points of the project. The constant availability of resource k is r_k, $k \in R$, throughout the project horizon. We assume every non-dummy task i has a non-preemptive stochastic lognormal duration D_i with mean d_i, and a constant resource demand $r_{ik} \leq r_k$ of resource $k \in R$ in execution. In CCPM, we assume that task i consumes exactly one type of resource k_i (Leach, 2014, p.170), i.e., $r_{ik} = r_{ik_i}$ if $k = k_i$, $r_{ik} = 0$ if $k \neq k_i$. For each task i, let $\Gamma_i^{-1} \subseteq V \setminus \{i\}$ denote the set of its immediate predecessors. The set E of precedence relationships is given as: $E = \{(i,j) : i \in \Gamma_j^{-1}, j \in V\}$. For every pair $(i,j) \in E$, task i must be finished before the start of j. All the tasks and precedence relationships form an acyclic task-on-node network PN(V,E). In the network, precedence relationships (i,j) are denoted by $i \to j$ when referring to chains, and tasks are topologically numbered, meaning that if $(i,j) \in E$, then $i < j$. A *resource contention* is a situation where the total resource demand exceeds the resource availability during some time period of the project.

Figure 1 illustrates the main steps of our buffer sizing procedure.

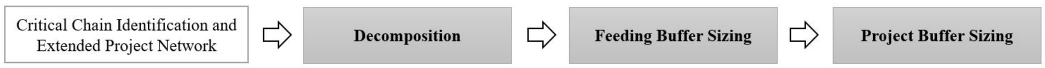

Figure 1. The Proposed Buffer Sizing Procedure

The preparation step in buffer sizing is critical chain identification, i.e, the identification of a baseline schedule without resource contentions, given mean task durations d_i. This problem is a special case of the classical resource constrained project scheduling problem (RCPSP, Demeulemeester and Herroelen (2002, p. 203)), where every task requires only one resource in R. This special case is strongly *NP*-hard, even when $m = 2, d_i = 1$ and $r_{ik_i} = r_{k_i} = 1$ (Bernstein et al., 1989). The literature describes several heuristics to solve the classical RCPSP, with the objective of minimizing the makespan. However, we propose a new heuristic for this special case of the RCPSP, which is designed to produce few additional precedence relationships from breaking resource contentions, and also short project makespans. The output is an extended precedence relationship set that defines the extended project network for our buffer sizing procedure. From the extended network, denoted by PN$(V,\widetilde{E})$, we identify the critical chain and other noncritical chains via the critical path method. Tasks on the critical chain are *critical tasks* and others are *noncritical tasks*.

Next, a project buffer and feeding buffers need to be located, respectively, at the end of the critical chain and wherever a feeding chain joins the critical chain, with well-defined size. Our buffer sizing procedure consists of three main steps.

(1) Decomposition of the network PN$(V,\widetilde{E})$ based on the identified critical chain.
(2) Feeding buffer sizing to account for nonciritical chain uncertanties and avoid the critical chain being exceeded by noncritical chains with the insertion of feeding buffers, using graph theory and linear programming techniques.
(3) Project buffer sizing to absorb uncertainties on the entire network with feeding buffers inserted, by aggregating safety margins of individual components derived from the first step.

For a single project with 150 tasks, the running time of our procedure with Matlab R2018a is less than 10 seconds on a personal computer with processor Intel(R) Core(TM) i5-7400 CPU @3.00GHZ and installed RAM of 8.00GB. Since the procedure is used at the planning stage of projects, this computation time is small enough for practical use.

3 Computational Analysis

We compare our buffer sizing procedure, PP, with five methods in the literature on four performance indicators via simulation. These five benchmarks are: Cut and Paste Method (C&PM, Goldratt (1997)), Root Square Error Method (RSEM, Newbold (1998)), Adaptive Procedure with Density (APD, Tukel et al. (2006)), Monte Carlo simulation method (SMC, Tenera (2008)), and the Method of Yu et al. (2013). The four performance indicators are: the accuracy of project makespan estimation - P_1, the reliability of the estimated project makespan - P_2, the average feeding buffer size - P_3, and the indicator of whether or not the identified critical chain is challenged by the insertion of feeding buffers - P_4.

In addtition to the data of a real project provided on the website of the Operations Research & Scheduling Research Group (OR&S, 2019b; Batselier and Vanhoucke, 2015; Vanhoucke et al., 2016), we use the data of 90 projects with 150 tasks each, randomly generated with RanGen2 software (OR&S, 2019a; Demeulemeester et al., 2003; Vanhoucke et al., 2008, 2016).

The numerical results on the four performance indicators show that our procedure provides more suitable buffer sizing and more accurate project makespan estimation, as well as much smaller feeding buffers while the critical chains are not challenged. Table 1 presents an exemplar set of our consistent results.

Performance Indicator	PP	C&PM	RSEM	APD	SMC	Yu2013
P_1	0.16	0.50	0.56	0.57	1.00	0.67
P_2	0.83	1.00	1.00	1.00	1.00	1.00
P_3	1.75	5.72	6.44	8.23	10.56	2.35
P_4	0	1	1	1	1	0

Table 1. Comparative Performance of Six Methods

4 Concluding Remarks

Because of the large economic value at stake, the sizing of buffers is centrally important to CCPM. Despite extensive research, previous approaches suffer from two significant deficiencies, which occur at the planning stage of projects: erroneous buffer sizing leads to inaccurate estimation of project makespan, and the insertion of feeding buffers overrides the critical chain. To resolve these issues, we have developed a buffer sizing procedure, which analyzes the entire project network by decomposing it, to obtain more accurate information about the relative lengths of critical and noncritical chains, and about interactions between buffers, over complex network structures. Hence, the size of a buffer is determined using global information about the project network instead of local information about the longest chain. Based on extensive computational testing for both real and simulated data, our procedure provides much more suitable buffer sizing and more accurate project makespan estimation, as well as much smaller feeding buffers, than five widely used benchmark methods. Additional benefits of our procedure include delayed expenditure, and reduced work-in-process, rework, and multitasking.

Our work should be of direct value to project management companies, for several reasons. First, all the information required for the network decomposition is immediately available in every well documented project. Second, the algorithmic steps required can easily be implemented as an add-on to commercial project planning software, such as Microsoft Project. Third, the elimination of the issue of challenging the critical chain simplifies project planning and reduces replanning. Fourth, the flexibility in using safety margins enables a project company to adjust its service level, in order to take into account strategic issues that frequently influence project choice and prioritization. Fifth, our buffer sizing procedure enables significantly more accurate and robust estimation of project makespan than earlier methods, thereby helping project companies to avoid the problems of underestimation and overestimation, and their significant costs or opportunity costs. Sixth, by enhancing CCPM, the choice that project companies face between using traditional and

CCPM planning may be clarified. Overall, we hope that the following comment will be helpful to project companies: our work enables significant reduction in buffers relative to other methods, but because the buffers we design are more accurately sized and located, the result is an improvement in project estimation.

References

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.

Bernstein, D., Rodeh, M., and Gertner, I. (1989). On the complexity of scheduling problems for parallel/pipelined machines. *IEEE Transaction on Computers*, 38(9):1308–1313.

Demeulemeester, E. and Herroelen, W. (2002). *Project scheduling: a research handbook*. Kluwer Academic Publishers.

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

Goldratt, E. M. (1997). *Critical Chain*. USA: The 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.

Kelley, J. E. and Walker, M. R. (1959). Critical-path planning and scheduling. In *Proceedings of the Eastern Joint Computer Conference*, pages 160–173. ACM.

Leach, L. P. (2014). *Critical Chain Project Management*. USA: The Artech House Publishers, 3rd edition.

Newbold, R. C. (1998). *Project Management in the Fast Lane – Applying the Theory of Constraints*. USA: CRC Press.

OR&S (2019a). Operations Research and Scheduling Research Group - RanGen. `http://www.projectmanagement.ugent.be/research/data/RanGen`. Last accessed on Feb 18, 2020.

OR&S (2019b). Operations Research and Scheduling Research Group - Real data. `http://www.projectmanagement.ugent.be/research/data/realdata`. Last accessed on Feb 18, 2020.

Tenera, A. B. (2008). Critical chain buffer sizing: A comparative study. In *2008 PMI research conference: Defining the future of project management*, pages 1–14, Warsaw, Poland. PMI.

Tukel, O. I., Rom, W. O., and Eksioglu, S. D. (2006). An investigation of buffer sizing techniques in critical chain scheduling. *European Journal of Operational Research*, 172:401–416.

Vanhoucke, M., Coelho, J., and Batselier, J. (2016). An overview of project data for integrated project managment and control. *The Journal of Modern Project Management*, 3(3):6–21.

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

Yu, J., Xu, Z., and Hu, C. (2013). Buffer sizing approach in critical chain project management under multiresource constraints. In *Proceedings of 2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering (ICIII)*, volume 3, pages 71–75. IEEE.

Zhao, W., Hall, N. G., and Liu, Z. (2020). Project evaluation and selection with task failures. *Production and Operations Management*, 29(2):428–446. doi: 10.1111/poms.13107.