

TEACHING CRITICAL CHAIN PROJECT MANAGEMENT: AN ACADEMIC DEBATE, OPEN RESEARCH QUESTIONS, NUMERICAL EXAMPLES AND COUNTERARGUMENTS

William P. Millhiser
Baruch College, The City University of New York
william.millhiser@baruch.cuny.edu

Joseph G. Szmerekovsky
College of Business, North Dakota State University
joseph.szmerekovsky@ndsu.edu

15-October-2008

Abstract

Goldratt's *Critical Chain* (1997) and the resulting project scheduling practices known as Critical Chain Project Management (CCPM) have been popular in our business schools but a subject of disagreement among our scholars. We survey the academic literature about the pros and cons of CCPM to examine claims in the book. We then present a sequence of numerical practice problems that are designed to clarify a student's understanding of the reading and highlight complications that Goldratt ignored. We conclude with two counterexamples that demonstrate ideas in the book that are not necessarily true in practice. Throughout the article are a number of open questions about CCPM that we as an academic community have not yet answered.

1. INTRODUCTION

Critical Chain (Goldratt 1997) is an engrossing fictional business novel about the complications of classical PERT and CPM project scheduling techniques. As many readers of this journal know, *Critical Chain* attracted much attention among scholars and project management professionals. It was reviewed in both the business and academic press, initiated the area of Critical Chain Project Management (CCPM), inspired dozens of scholarly and practitioner-oriented articles and books, provided the basis for new consulting practices, and even motivated new project planning software tools (see Herroelen and Leus 2001 and Herroelen et al. 2002 for surveys).

Firms around the world are documenting success stories due to CCPM (see Herroelen et al. 2002, p. 45; Woepel 2005, pp. 110-111; Leach 2005, pp. 18-20). The 2006 Franz Edelman Award for Achievement in Operations Research was earned due to a successful CCPM implementation (Srinivasan et al. 2007).

If one wants to teach CCPM, the first problem is selecting a supporting text. While some Operations Management textbooks now acknowledge CCPM in footnotes and sidebars (see, for example, Jacobs et al. 2009, p. 696), we are aware of no textbook that systematically presents CCPM with examples, exercises and case studies the way such resources exist for PERT/CPM methods. It has been our practice to assign Goldratt's *Critical Chain* and supplement the discussion with notes and practice problems that we summarize in this article.

There are a number of issues related to Goldratt (1997). First, academic opinion about critical chain scheduling and buffer management has been mixed. There is concern that we now have a tendency to give Goldratt credit for ideas that were published as early as the 1960s. For example, the notion of a critical chain can be attributed to Wiest's (1964) "critical sequence" (Herroelen and Leus 2001, p. 563) and a project buffer to O'Brien (1965) (Trietsch 2005, p. 31). Others feel some teachings in *Critical Chain* are simply not empirically justified, or worse, contradict well-accepted scheduling research (Herroelen and Leus 2001, Herroelen et al. 2002, Raz et al. 2003, Trietsch 2005).

In addition to scholarly criticism of *Critical Chain*, we feel that introducing the text in a classroom requires care. For the instructor who intends to teach classical and critical chain project scheduling methods, there is a divide between PERT/CPM and CCPM. Some feel *Critical Chain* does not properly complement currently-accepted project management practices, and in so doing, poses an unnecessary methodological choice between CCPM and mainstream practices (Raz et al.

2003). “There is a variety of such methods some of which are mutually incompatible and attempting to describe them all is likely to cause confusion” (Nokes et al. 2003; see p. 26 for a comparison of nine differences). And it is our experience that students ask, “Why do you teach PERT when we learn so much from *Critical Chain*?” How do we explain this to students?

That said, there are few books as cleverly written to interest students in important issues in project planning and scheduling, a compelling argument for including *Critical Chain* in one’s syllabus. In our experience, students of all business disciplines find the reading a worthwhile endeavor, especially considering the need for good project management skills in today’s services-based economy. For example, in a course for undergraduate operations majors taught by one of the authors, students routinely rank the book as the most important learning tool of the semester, if not their major. The other author’s doctoral dissertation choice was motivated by readings in *Critical Chain*.

Unfortunately, the book does not always lend itself to the classroom because students are not necessarily ready to implement Goldratt’s project planning ideas. Some will need to clarify what they read, and many cannot immediately reformulate a project schedule using project and feeding buffers. Furthermore, most are unaware of the decades of project management and scheduling research that precede the book, or that some of the ideas may not be well-advised. Finally, more senior students (such as executives and MBAs) will have difficulty implementing CCPM directly upon completing the reading.

The purpose of this paper, therefore, is to give teachers insights about adopting *Critical Chain* in the classroom. It is intended for professors who are not necessarily experts in the literature of project management or scheduling theory, but are often asked to guide students in a unit on

project management. We assume that the reader has read *Critical Chain*. For those who have not, an alternative is the brief chapter-by-chapter summary in Woepfel (2005).

First, in Section 2, we survey the academic literature on CCPM and mention strategies for sharing these issues with students. As the reader will see, scholars are in disagreement. In Section 3, we momentarily accept the book “as is” and develop a sequence of simple numerical examples that we wish had been included in the text to help students systematically solidify their understanding of the reading. Two are designed to practice resource-constrained project scheduling, like the example in Goldratt’s Chapter 22. The third requires students to reformulate a classic PERT/CPM problem as a CCPM schedule while facing complications ignored by Goldratt. The fourth is a simple example where resources are constrained across multiple projects, of which there is no numerical example in *Critical Chain*. We conclude with two counterexamples to Goldratt’s claims about multitasking and buffer pooling.

2. THE DEBATE: DOES CRITICAL CHAIN PROJECT SCHEDULING WORK?

If project management and scheduling scholars critiqued the book and CCPM but yet companies are reporting success stories, how does this impact the classroom discussion? We now survey this debate grouping the comments by the so-called advantages of CCPM as advocated in the text and urge the reader to share this information with students. We begin with a potential student misunderstanding.

Claim: CCPM Delivers Successful Projects. Some students will overlook the point that *Critical Chain* proposes a recipe for project *planning* success, not absolute project success (Raz et al. 2003, p. 30). Statements like, “Everybody knows projects don’t finish on time or on budget, and even if they do, it means they had to compromise on content” (Goldratt, p. 25) mislead inexperienced students into thinking that time, budget and content are all that matter. Other

factors are known to be key elements to project success (Lipovetsky et al. 1997; Shenhar and Dvir 2007). For example, the Sydney Opera House, arguably one of the world's most iconic architecture projects, was delivered a decade behind schedule and millions over budget. At the other extreme, the LA Subway was delivered on time and under budget, winning the Project Management Institute award for best project in 1994. Yet ridership remains a fraction of what was expected. (For details of these examples and more, see Shenhar and Dvir 2007).

A related issue is that while CCPM is a proposed tool for scheduling constrained resources across multiple projects, the book fails to address how firms choose projects or rank their relative strategic importance. Time permitting, we ask students to read articles such as Wheelwright and Clark (1992) to understand the importance of establishing a process for choosing projects that lead to long-term competitiveness.

Claim: Projects Finish Late. Regarding the foregoing quote that “projects don't finish on time or on budget,” perhaps the most frequently cited empirical evidence of this statement is the Standish Group's long-term study of thousands of IT projects around the globe. The average delay, cost overrun and percent canceled prior to completion are astounding (for details, see Woepfel 2005, p. 105, and Klastorin and Mitchell 2005, p. 23). Leach (2005, p. 6) surveys other well-known project management planning anecdotes in other industries. We are unaware of any scientific study that assesses these metrics across a sample of projects controlled with CCPM.

Claim: Focus on Time, Not Cost. Goldratt said, “Companies are so immersed in the mentality of saving money that they forget that the whole intention of a project is ... to make money. ... It's a simple fact that they try to cut the budget by a few percent and cause the payback period to double” (p. 62). Evidence supports this claim. For example, Port et al. (1990) cite a McKinsey & Co. estimate that firms lose 1/3 of their profits when they accept six-month delays to stay within

product development budgets; spending 50% over the development budget to meet a release date attenuates total profits only 4%.

It is worth noting that this is dependent on the definition of “on time.” If buffers and due dates change then the meaning of “on time” also changes. Clearly, PERT/CPM project management is geared to completing projects on time with minimum cost. If the notion of on time is erroneous this can be a failed approach. However, NPV project management is geared toward maximizing project NPV (Herroelen et al. 1997) and can clearly identify what on time should mean.

Finally, we suspect that W. Edwards Deming would support the focus on time, not cost. Recall that his fourth point is, “end the practice of awarding business on price alone” (Walton 1986).

Claim: Recognize Parallel Activities. “In the case of parallel steps, ... the biggest delay is passed on to the next step” (Goldratt, p. 122). On one hand, this assertion is well-founded. For example, Schonberger (1981) showed that projects with variable activity times will always exceed the time of the deterministic critical path; the greater the variability or number of parallel paths, the greater the delay. However, the more parallel paths in a project, the greater the chance that noncritical chains will turn critical and thus feeding buffers will have been misplaced (Raz et al. 2003, p. 28). Example 3.2 (below) illustrates this point. Note how all feeding buffers cannot be assigned a time duration prescribed by Goldratt’s “50% rule.” See Hoel and Taylor (1999) for further discussion about buffer insertion.

Claim: Pool Safety Time in Buffers (Not Individual Tasks). Since “each step has a minimum 200% safety” (Goldratt, p. 155) which is exhausted needlessly due to the “student syndrome,” Goldratt suggests shifting safety margins from task owners to pooled buffers. The characters ultimately agree upon cutting original time estimates in half, and pooling half of the savings in a

project buffer (p. 156). There is modest support for this approach. One example is weather-related padding used in construction (Clough and Sears 1991). In a different direction, since deadlines are a form of quota, deadline elimination through safety pooling agrees with W. Edwards Deming's point about abolishing numerical quotas (Trietsch 2005, p. 31).

Some argue that any benefits of the pooled buffers have yet to be empirically justified (Raz et al. 2003, p. 27). Moreover, "The ex ante realistic 50% task duration estimate may well be based on loose ground. ... In many cases, the result might be an unnecessarily large amount of protection, which could lead to uncompetitive proposals and the loss of business opportunities" (Herroelen and Leus 2001, pp. 562-564). Indeed, almost every project selection method uses time in evaluating projects. For example, capital budgeting techniques use an interest rate to trade off time and money (Eschenbach and Cohen 2006) and the payback period is itself a time-based metric. Even methods which do not mandate explicit use of time estimates, such as cost/benefit analysis and scoring models, can readily account for time in their analysis. Also, when estimates are used for bidding purposes, best-value bids are not taken purely on cost but several factors including delivery dates (Gransberg and Ellicott 1997; Kashiwagi and Byfield 2002). One numerical experiment has shown that the root-square-error method for buffer size estimation is a more accurate approach (Herroelen and Leus 2001).

Raz et al. (2003) articulate a related concern we often hear from students: "Imposing shortened duration estimates on task owners will reduce their commitment to the estimates. In addition, the knowledge that their estimates will be reduced is likely to encourage task owners to add larger margins so they still have the safety margin they prefer after the correction" (p. 27). Furthermore, since each added buffer is a new item on a Gantt chart, this may result in more clutter, potential confusion, and quite possibly, more unscheduled communication to coordinate the project team (pp. 29-30).

Yeo and Ning (2006) support Goldratt on pooling buffers. They present survey data and a model that suggest that pooled safety buffers are appropriate in a construction equipment procurement supply chain.

Claim: Eliminate the Early vs. Late Start Dilemma with Feeding Buffers. If noncritical activities follow earliest or latest start times, the project leader will lose focus, resulting in costly delays (Goldratt, pp. 70-71). The concept of a feeding buffer first appeared in the literature in the 1980s and 90s; “the basic idea was understood by the professional community long before publication of *Critical Chain*” (Triesch 2005, p. 32).

Example 3.2 (below) was designed to highlight feeding buffer insertion complications. Another issue is that “Pushing activities backward in time in order to insert a feeding buffer may, and mostly will, create resource conflicts. How these conflicts are to be resolved is not described in detail. A possible way for resolving the conflict may be to push the chain of activities feeding a feeding buffer backwards in time until a feasible schedule is obtained again” (Herroelen and Leus 2001, pp. 565-566).

Claim: Avoid Parkinson’s Law with CCPM. Parkinson (1955) said, “work expands so as to fill the time available for its completion,” a phenomenon that was tested empirically (Moss 1978) and confirmed in projects (Schonberger 1981, Gutierrez and Kouvelis 1991). Goldratt implies Parkinson’s Law leads to further delays in what he calls the student syndrome. “[A]lmost half the steps were reported finishing almost exactly on the nose. ... for almost a third of the steps ... the elapsed time was ten to twenty percent longer than the original estimate” (pp. 123-124). The latter claim is supported by Hill et al.’s (2000) study where 32% of 500 software development activities

overran time estimates. (Hill et al. found that 60% of the activities were reported complete ahead of schedule and only 8% finished “on target.”)

Herroelen and Leus (2001) suggest Parkinson’s Law is not necessarily bad, “since you cannot have the workforce under stress all the time” (p. 562). Furthermore, Bender et al. (2008, p. 95) say, “The advantages of procrastination are well documented: the closer to a deadline a task is executed, the less processing time the task appears to require. Hence, it is common for a person to delay executing some onerous job in order to spend as little time as possible working on it.” The message we impress upon students is that given a due-date, it is rational (and likely efficient) to finish a job near the due-date, and in the reality, people do not estimate the time required to complete a task but rather the time by which the task will be completed. That is, they quote due dates, not activity durations. Thinking of the estimate of an activity’s duration as a due date, it no longer implies that someone is lazy or wasting time if they complete the activity near the due date. Rather, they are likely being efficient.

Regarding Parkinson’s Law, it is worth noting that most people will not be assigned to just one task, but multiple tasks, perhaps on multiple projects. If they schedule their time efficiently, they will complete most of their activities on time but not early. Again, this is not because workers are lazy or wasteful, but due to the structure of an optimal schedule for minimizing tardiness-based objectives. Whether or not CCPM avoids Parkinson’s Law is still an open research question.

Claim: Schedule Resource Constraints Across Multiple Projects. While not new, resource leveling through the critical chain—that is, keeping the amount of resources tied up in a project as consistent as possible over the project’s lifetime—is an important element of the book. However, Raz et al. (2003) question the applicability of a critical chain project solution given that the

binding resource across different projects could alternate at different points of time. (We invite the student to explore a simpler case of constraints across projects using example 3.4 below.)

Claim: Shun Multitasking. It has been our experience teaching *Critical Chain* to undergraduates that an important lesson is the loss of focus and increase in lead times induced by what Goldratt calls “multi-tasking” (see pp. 125-127). Undergraduates often remark how they identified and eliminated multitasking in their personal lives as a result of the reading. By multitasking, Goldratt means the assignment of one entity to multiple tasks or projects, potentially leading to task completion delays. However, due to the more commonly accepted meaning of multitasking in North American culture—performing several operations simultaneously thus achieving a time savings—students sometimes miss Goldratt’s point. As a result, we recommend clarifying Goldratt’s figure on p. 126. To further clarify, we invite students to listen to Hamilton (2008) and the humorous Sharp (2008).

Regarding Goldratt’s definition of multitasking, contrary to the claims in the text, there is evidence that multitasking is beneficial. For example, McCollum and Sherman (1991) found that assigning R&D employees to up to three simultaneous projects improved ROI. Trietsch (2005; p. 32) concurs that multitasking is unavoidable and sometimes desirable in practice. And according to Demeulemeester and Herroelen (1996) and Herroelen and Leus (2001), multitasking can lead to activity preemption which has been shown to shorten project durations. We provide a numerical example in Section 3.5.

Indeed, it is well known that multitasking increases scheduling flexibility. In the end, there is a difference between efficient assignment of resources to a project and efficient use of resources within an organization. Also, there is a difference between departmental and individual commitments to complete a task. Departments obviously must accept many requests for work, so

requiring that no one in a department be assigned to more than one task at a time would make the department manager's scheduling task impossible, even if it works for the project manager.

Absent from Book: Scope Creep. Project management professionals tell us that nearly every project accepts new or changing requirements once underway, called scope creep. While not addressed by Goldratt, we find it useful to discuss how one would update a CCPM-based project plan if scope creep occurs. For example, we ask students, "Should scope creep eat at feeding and project buffers, or should project delivery commitments be adjusted?" Though both positions can be reasonably defended, the most important concept for students to consider is a comparison of the cost of the additional work with the cost of a delay in the project completion date. By Goldratt's reasoning, a delay in the project would be a more important cost than the cost of any additional work.

Absent from Book: Implementation. We feel that another complication is that the students are given insufficient information in the text to implement CCPM at work. There are a few resources in print. For example, Woepfel (2005; pp. 114-116) provides a brief 8-step approach and Nokes et al. (2003) give more thorough advice. Case studies of successful CCPM adoptions are described in Woepfel 2005 (pp. 117-125) and Srinivasan et al. (2007). Those with larger budgets may want to consider the consulting, media and software resources of firms such as The Goldratt Institute and ProChain Solutions, Inc.

It is important to point out that successful implementation of CCPM relies of several factors. Much like Total Quality Management programs, Nokes et al. (2003) remind us that, "it would be very difficult for an isolated project manager to run a project using the critical chain unless it is widely understood within the organization and the necessary management support is given."

3. NUMERICAL EXAMPLES AND COUNTEREXAMPLES

We present a sequence of three numerical examples to practice critical chain scheduling. The first is simple resource-constrained project scheduling problem that we find useful for less technical undergraduate students from various business disciplines. Some will find this example unnecessary. Solutions to examples 3.1 to 3.4 are available in the appendix.

3.1 Simple CCPM Formulation: The Project of Jack and Jill

Consider a project with the activity precedence relationships and time estimates given in the following table.

<u>Activity</u>	<u>Immediate Predecessor</u>	<u>Estimated Time (weeks)</u>
A	--	6
B	A	12
C	A	4
D	B, C	10

Two workers are available. Due to her skill set, Jill is responsible for activities A and D.

Similarly, only Jack can do activities B and C.

- Assuming that the time estimates are perfectly accurate, what is the expected project duration? Explain. (Use the time estimates as is; do not insert project or feeding buffers.)
- Which activities are critical? Explain.
- Assume Jack and Jill can be cross trained to do any activity, but will not share activities. Can the activities be reassigned to Jack and Jill such that the project is completed in less time? If so, indicate the new assignment and project duration.
- Consider the following assignment: Jill does activities A and B, Jack does activities C and D. Assume that the times in the foregoing table are estimates padded with safety time. Use the ideas from the *Critical Chain* to reformulate the project schedule using project and feeding buffers. Include a diagram similar to that on Goldratt's page 218.

3.2 PERT vs. CCPM: The Project of Reliable Construction Company

The purpose is to convert a well-known PERT problem into a critical chain plan using Goldratt's methods, spanning the gap between PERT and CCPM methods. It also presents three situations not addressed in *Critical Chain*: when critical activities precede noncritical activities, when there is task time distribution information and when the "50% rule" for feeding buffer allocation is infeasible.

Consider the project of Reliable Construction Co. (Hillier and Lieberman 2001) with the precedence relationships (Table 10.1, p. 469) and optimistic (*o*), most likely (*m*), and pessimistic (*p*) time estimates (Table 10.4, p. 488) and expected critical activities A-B-C-E-F-J-L-N that yield the expected project duration 44 weeks.

- (a) In *Critical Chain*, the characters ultimately decide that "time allotted for each step will only be cut by one-half. On the other end, the project buffer will not be equal to what they trimmed. It will be set to only half of it" (Goldratt, p. 156). Use this rule to reformulate each time estimate and the duration of the project buffer. What is the revised expected project duration?

- (b) Similarly, regarding noncritical activities, "For each feeding path they decide to cut the original time estimates of the steps in half and use half of the trimmed lead time as a 'feeding buffer.'" (Goldratt, p. 158). Use this information to sketch a figure similar to that at the bottom of page 158 for the project. In the figure, indicate the duration of each activity and buffer.

3.3 Resource Constraints within a Project: The Project of Person “X”

Goldratt presents a critical chain project diagram on pages 214 and 218. The following example asks students to develop such a diagram from a numerical example. (We do not tell students at first that it results in an identical figure to that on p. 218.) Consider the following project activity list.

Activity	Immediate Predecessors	Estimated Activity Time (weeks)
A	--	5
B	A	4
C	--	3
D	C	9
E	--	2
F	E	5
G	F	3
H	G	8
I	B, D, H, K, M	4
J	--	3
K	J	2
L	--	1
M	L	7

One person (called person “X”) is the only employee with the skills needed to accomplish activities B, D, F, K, M. Each remaining activity (A, C, E, G, H, I, J, L) has a unique person assigned to the task and can therefore be completed independently of other activities. Develop a critical chain project schedule. Include a figure similar to that on page 218 and indicate the duration of activities and buffers.

3.4 Resource Constraints across Multiple Projects: The Company of Jack & Jill & Jane

The purpose of this problem is to provide practice with resource constraints across projects, an idea that was mentioned but not developed in *Critical Chain*. The Company of J&J&J has two projects. One is the project of Jack & Jill (where Jill must perform A and D and Jack must perform B and C, as given in the table in problem 3.1). The other is the project of Jack & Jane which is identical to the project of Jack & Jill but the activities are labeled E, F, G, and H where

Jill must perform E and H. It is more expensive to delay the project of Jack & Jill. Find the CCPM schedule for these two projects.

We now turn to two counterexamples, starting first with Goldratt's claim that multitasking should be avoided.

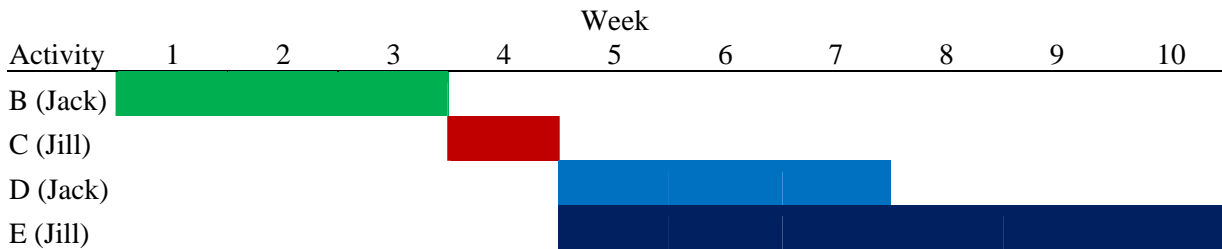
3.5 Counterexample: Is Multitasking Always Bad?

Consider a project with the activity precedence relationships and time estimates given in the following table.

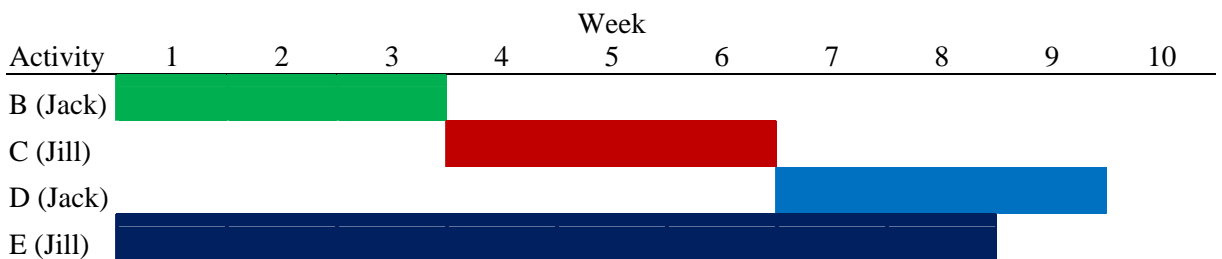
Activity	Immediate Predecessor	Estimated Time (weeks)	Qualified Personnel
A	--	0	--
B	A	3	Jack
C	B	1	Jill
D	C	3	Jack
E	A	6	Jill
F	D, E	0	--

Activities A and F are dummy activities indicating project start and completion. Two workers are available. Due to her skill set, Jill is responsible for activities C and E. Similarly, only Jack can do activities B and D. Since Jack's activities are ordered by the precedence relationships, there is no opportunity for Jack to multitask. However, Jill's activities are not ordered by precedence constraints and therefore she may multitask, that is, work on both activities C and E at the same time. However, since multitasking is inefficient, assume that Jill's total output is only 2/3 of what it would be if she dedicates her time to only one activity. In other words, any week that Jill multitasks activities C and E, that week will require 50% more time than originally budgeted (requiring 3 weeks of multitasking to complete one week of work on both activities simultaneously).

Despite the inefficiency of multitasking, the schedule that completes the project as early as possible requires it. To see why, consider the following schedule which completes the project as early as possible without multitasking:



Without multitasking, we must sequence activities C and E, resulting in a minimum of 10 weeks to complete the project since C has a predecessor and successor that require 3 weeks. Now, consider the following example where multitasking is used:



The project is now completed in only 9 weeks. Though Jill is less efficient (she requires 8 weeks to do 7 weeks' work), the project finishes sooner because her 8 days of work start immediately whereas the 7 days of work were delayed 3 days for activity B. Hence, we conclude that though multitasking is inefficient for individuals it need not be inefficient for projects and organizations as a whole.

3.6 Counterarguments to Anecdotal Reasoning

Goldratt uses examples to show how wasteful buffering is in organizations and how easy it is to eliminate vast amounts of time from a project. For example, Goldratt's hero manages to get a coatings subcontractor to agree to "drop everything else and work on it" when a job arrives (Goldratt, pp. 183-184). The only apparent barriers for the subcontractor are a price increase and that he be given ten days notice (with updates) of the job's arrival. There are three problems with this exchange. First, the subcontractor never promises how long it will take to complete the job, and therefore, they do not actually know what they are getting for the extra money. Second, since the commitment is not to finish the job at a certain time but to start it after arrival, the only way to verify that the promise is being kept would be to have someone in the shop monitor progress during the completion of the job. Third, suppose the same subcontractor has another client making use of CCPM. What if the two jobs which are to be given top priority from the two different customers arrived at the same time? The subcontractor would renege on at least one commitment.

The shortsightedness of Goldratt's examples also extends to cutting time on jobs performed within an organization. Consider the example of the 7 weeks required to purchase a connector (Goldratt, p. 48). Suppose the actual work to place the order requires less than 15 minutes; if the supplier is out of stock someone could purchase the connector at a local electrical supply store in at most a few hours. Goldratt, however, implies that using CCPM would save such projects weeks of delay by reducing the length of the connector acquisition time to at most one day. In rebuttal, we propose three alternative scenarios:

1. The department head explains that the connector request must be handled through standard procedures so that no favoritism is shown.

2. The department head originally agrees. But, the next day when ten other project managers demand equal treatment having heard about the promise, he phones to renege on the promise.
3. The department head, having been trained in operations management techniques, dedicates a single staff member to process work requests which take 15 minutes or less to complete. This way, no such job will wait more than 24 hours before being processed. Similarly, the department head has guaranteed the quality and responsiveness of suppliers so that extended delays from stock outs are not an issue. All ten project managers throughout the company will experience significantly improved service while never employing CCPM.

The last scenario highlights that the issue Goldratt addresses is not a buffering issue. The large buffers are appropriate given how the department is being operated. But, rather than addressing the scheduling issues within the department, Goldratt addresses the buffering issue for one project. This leaves the disease but treats the symptom; CCPM gets credit for treating a symptom while other projects within the organization suffer from the disease. Also, an analogy can be made with just-in-time (JIT) or lean philosophies where the inflated buffer is thought to hide inefficiencies within the organization. Hence, an application of JIT or lean management should fix the problem with no need for CCPM techniques.

3.7 Computer Activities

Instructors of more advanced project management classes may wish to ask students to formulate examples 3.1 to 3.4 using project management software such as academic versions of Microsoft Project and Primavera. Furthermore, either using the software or sketching by hand, it is useful for students to sketch Gantt charts of the project plan. We are unaware of any instructors who teach ProChain Project Scheduling which is \$695 per copy.

4. CONCLUSION

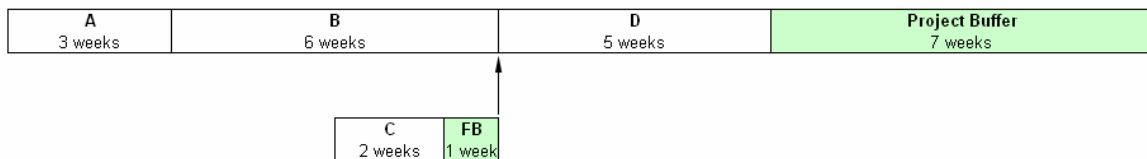
We hope that this document gives instructors actionable ideas for their project management lesson plans and helps them decide what content, if any, regarding CCPM they will include in a syllabus. As a final thought, Dan Triesch's (2005; p. 35) comments are worth repeating.

Since practically all of Goldratt's work had not been refereed, every academic who cites him should first imagine that he/she is required to evaluate Goldratt's work, as would a referee. As for those who may feel unqualified to referee Goldratt's work, they should be even more wary. ... To counterbalance the criticism, however, we must note that CC—although identical to the old spirit of PERT/CPM—provides a fresh opportunity for improvement by a renewed holistic approach to project planning and execution.

APPENDIX: Solutions to Examples 3.1 to 3.4

Example 3.1

- (a) Jack requires $12 + 4 = 16$ weeks to complete activities B and C. Therefore the project duration is $6+12+4+10 = 32$ weeks.
- (b) All activities are critical because the critical chain is either A-B-C-D or A-C-B-D.
- (c) There are multiple correct answers. One possible solution is assign A and B to Jill; C and D to Jack. The project duration reduces from 32 to 28 weeks.
- (d) The activity times for A, B, C, and D would be reduced to 3, 6, 2, and 5 weeks, respectively. The duration of the project buffer would be half the savings, namely, 7 weeks. The feeding buffer for noncritical activity C is one week. Activity C should begin at the end of the sixth week of the project as indicated below. Using Goldratt's notation, this can be visualized as follows:



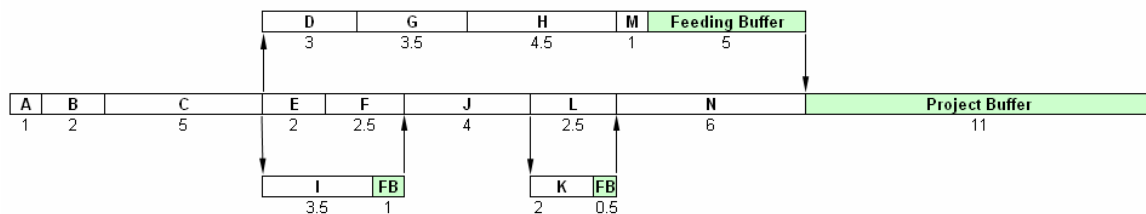
Example 3.2

- (a) If task owners obtain initial time estimates using the PERT three-estimate approach (see Hillier and Lieberman 2001, p. 488), it is unclear from *Critical Chain* how one applies the “50% rule” to these three estimates. One possible solution is to cut the three estimates o , m and p in half, or equivalently, cut the time estimate $\mu = (o+4m+p)/6$ by half. The revised time estimates become:

Activity	Revised Duration (weeks)	Activity	Revised Duration (weeks)
A	1.0	H	4.5
B	2.0	I	3.5
C	5.0	J	4.0
D	3.0	K	2.0
E	2.0	L	2.5
F	2.5	M	1.0
G	3.5	N	6.0

(The method of obtaining optimistic time o and pessimistic time p does not necessarily lend itself to fictitious padding; some students may argue to reduce only m by half, and thus the activity time estimate becomes $\mu = (o + 4(m/2) + p) / 6 = (o + 2m + p) / 6$, a reduction less than 50%.) Since the critical activities are A-B-C-E-F-J-L-N, the project buffer should be half of their combined times, or 11 weeks.

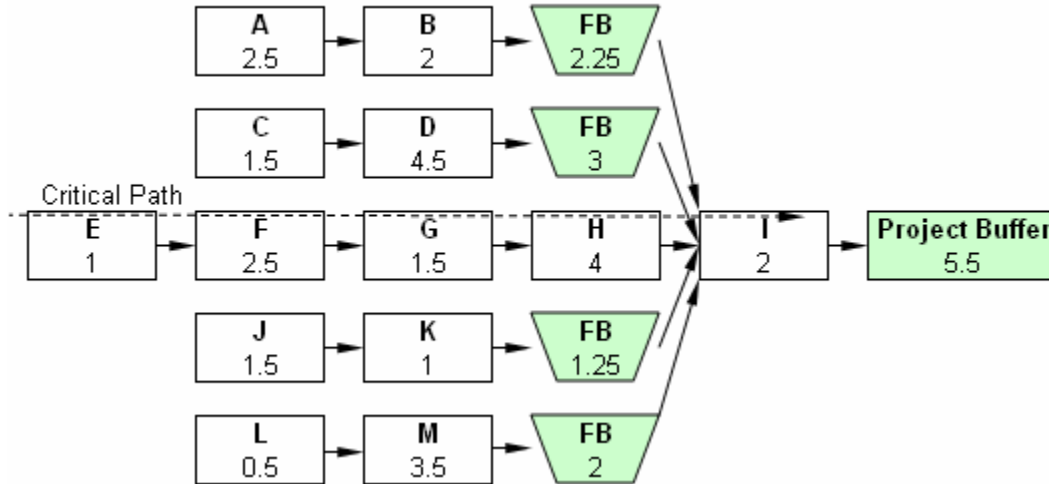
- (b) In the following figure, note that the feeding buffer for the noncritical chain D-G-H-M should be 6 weeks but only 5 are available. Similarly, the feeding buffer for I should be 1.75 weeks but only 1 is available and the feeding buffer for K should be 1 week but only a half-week is available. All durations in the following figure are in weeks.



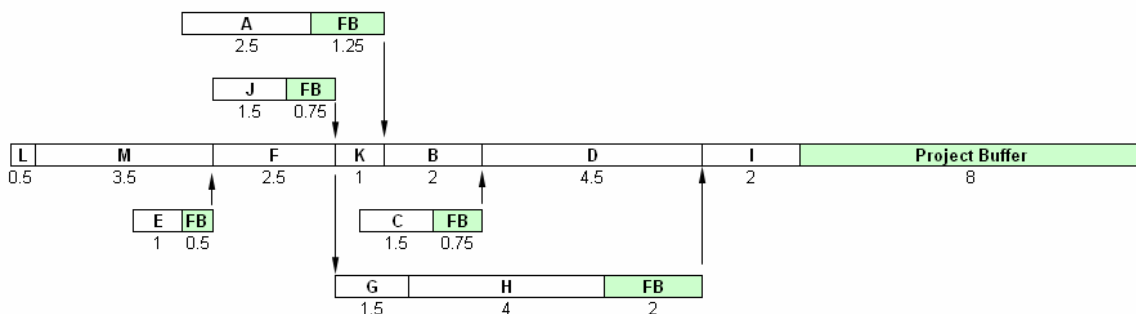
A classical PERT formulation of the original problem reveals that activities on expected critical path are the same as the critical chain above, and thus the expected project duration and variance are 44 and 9 weeks, respectively. Contrast the CCPM formulation's 36-week project duration with the PERT formulation's implication that there is 99.6% chance that the actual project duration is greater than 36 weeks.

Example 3.3

After cutting time estimates in half, students who fail to observe the resource constraint will erroneously obtain a solution similar to that on page 214 with the critical path E-F-G-H-I and a project duration of 16.5 weeks (including the project buffer). Such a diagram follows. All units are weeks.



Noting that person X is responsible for activities B, D, F, K, M (or a total time of 13.5 weeks, 2.5 weeks longer than the 11 weeks of critical path E-F-G-H-I), it is clear that there is a critical chain. We could redraw the foregoing figure similar to that on page 218, as shown below. The critical chain is person X's activities plus any activities that delay X from starting (the shortest of which is L, ½ week) and any activities that delay the project completion when X is done (activity I, 2 weeks). Thus, the expected project duration is now 24 weeks.

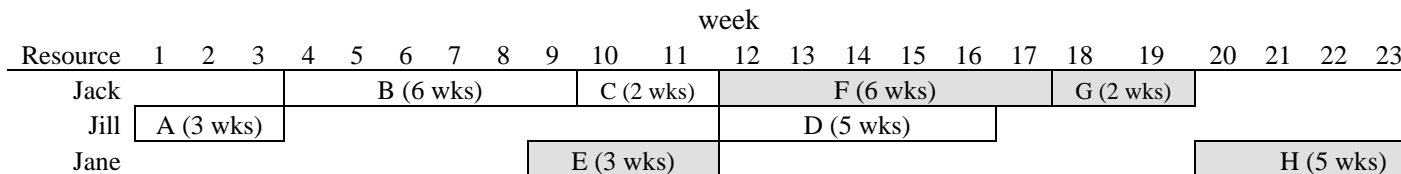


Example 3.4

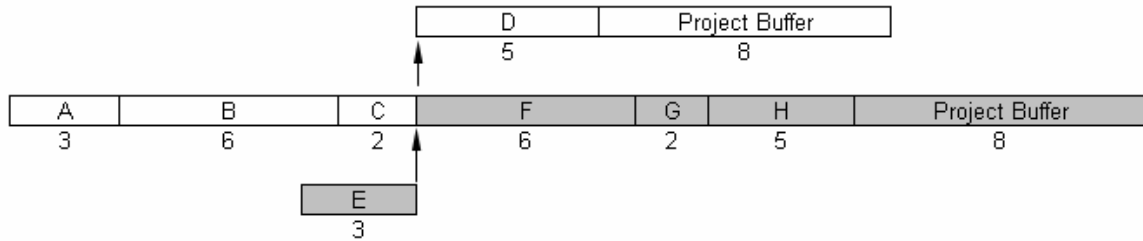
Step 1: Prioritize projects. According to CCPM, the project of Jack & Jill receives priority due to the delay cost.

Step 2: Find independent schedules for the two projects. For the project of Jack & Jill the critical path is A-B-C-D (it is also optimal to transpose B and C) taking time $3+6+2+5=16$ weeks and receiving a buffer of 8 weeks for a completion date of 24 weeks. For the project of Jack & Jane the schedule is the same using the critical path E-F-G-H.

Step 3: Resolve resource conflicts in favor of the higher priority project. The bottleneck resource is Jack since he is needed on both projects from time 3 through time 11. We resolve this by giving priority to the project of Jack & Jill. So, starting at time 3 Jack will execute the sequence of critical activities B-C-F-G. This results in no change in the project of Jack & Jill. But, the project of Jack & Jane is delayed. Jack will not start activity F until he finishes activity C at time 11. So, using the late start schedule for project Jack & Jane, Jane will now start activity E at time 8 so that it finishes at time 11. This means the entire schedule for project Jack & Jane will be delayed 8 weeks with a completion date of 32 weeks, as shown in the following Gantt chart. (grey shading refers to Jack and Jane's project.)



This results in a critical sequence of activities across two projects of A-B-C-F-G-H, as shown below.



Comments:

1. Note that we do not sequence a feeding buffer between E and F since these activities lie on the project's critical chain, though this point is not clear from the book.
2. This CCPM schedule has several attractive properties. Including the project buffer, the project of Jack & Jill has a completion date of 24 weeks. No other schedule can provide an earlier completion date. The project Jack & Jane cannot have a completion date less than 32 weeks. Hence, the CCPM schedule provides the best possibility both for the high priority project and for the low priority project.
3. Suppose we allow Jack to multitask and work on both projects at the same time. Then, even if multitasking is inefficient, it could be possible to complete both projects later than 24 weeks but before 32 weeks. When might this be a good idea? CCPM assumes a very strict priority hierarchy, delaying one project 8 weeks is better than delaying the other project just one week. But what if the cost of delaying the two projects was about equal beyond a certain date?
4. Jack moves between the two projects. In particular Jack needs to be updated about the progress on activities A and E so that he is ready to start B and F once they are available. Whenever such a resource switch occurs along the critical chain, this should be handled with a resource buffer which is not a time buffer but an early warning system for the resource to not be late.

REFERENCES:

- Bender, M.A., R. Clifford, K. Tsihclas. 2008. Scheduling Algorithms for Procrastinators. *J. of Scheduling* **11**(4) 95-104.
- Demeulemeester, E.L., W.S. Herroelen. 1996. An efficient optimal solution procedure for the preemptive resource-constrained project scheduling problem. *Eur. J. of Oper. Res.* **90**(2) 334-348.
- Eschenbach, T., R. Cohen. 2006. Which interest rate for evaluating projects? *Engrg. Management J.* **18**(3) 11-19.
- Goldratt, E.M. 1997. *Critical Chain*. North River Press, Great Barrington, Mass.
- Gransberg, D.D., A.A. Ellicott. 1997. Best-value contracting criteria. *Cost Engrg.* **39**(6) 31-34.
- Hamilton, J. 2008. Think you're multitasking? Think again. National Public Radio. 2-Oct-08 broadcast www.npr.org/templates/story/story.php?storyId=95256794.
- Herroelen, W.S., P. Van Dommelen, E.L. Demeulemeester. 1997. Project network models with discounted cash flows a guided tour through recent developments. *Eur. J. of Oper. Res.* **100**(1) 91-121.
- Herroelen, W., R. Leus. 2001. On the merits and pitfalls of critical chain scheduling. *J. of Oper. Management* **19**(5) 559-577.
- Herroelen, W., R. Leus, E.L. Demeulemeester. 2002. Critical chain project scheduling: Do not oversimplify. *Project Management J.* **33**(4) 48-60.
- Hill, J., L.C. Thomas, D.E. Allen. 2000. Experts' estimates of task durations in software development projects. *Internat. J. of Project Management* **18**(1) 13-21.
- Hillier F., G. Lieberman. 2001. *Introduction to Operations Research*, 7th ed, McGraw Hill 469-491.
- Hoel, K., S.G. Taylor. 1999. Quantifying buffers for project schedules. *Production and Inventory Management J.* **40**(2) 43-47.
- Jacobs, F.R., R.B. Chase, N.J. Aquilano. 2009. *Operations and Supply Management*, 12th ed., McGraw-Hill.
- Kashiwaga, D., R.E. Byfield. 2002. Selecting the best contractor to get performance: on time, on budget, meeting quality expectations. *J. of Facilities Management* **1**(2) 103-116.
- Klasterin, T., G. Mitchell. 2005. A New Paradigm for Project Management. *POMS Chronicle* **12**(3) 23-24.
- Leach, L.P. 2005. *Critical Chain Project Management*, 2nd ed, Artech House, Inc.
- Lipovetsky, S., A. Tishler, D. Dvir, A.J. Shenhar. 1997. The relative importance of project success dimensions. *R&D Management* **27**(2) 97-106.

- McCollum, J.K., J.D. Sherman. 1991. The effects of matrix organization size and number of project assignments on performance. *IEEE Trans. on Engrg. Management* **38**(1) 75-78.
- Moss, R. 1978. An empirical test of Parkinson's Law. *Nature* **273**(5659) 184.
- Nokes, S., I. Major, G. Greenwood, D. Allen, M. Goodman. 2003. *The definitive guide to project management: the fast track to getting the job done on time and on budget*, Prentice Hall.
- O'Brian, J.J. 1965. *CPM in Construction Management: Scheduling by the critical path method*, McGraw Hill.
- Parkinson, C.N. 1955. Parkinson's Law. *The Economist*. **177**(5856) 635 – 637.
- Port, O., Z. Schiller, R.W. King. 1990. A smarter way to manufacture: How “concurrent engineering” can reinvigorate American industry. *Bus. Week*, **3157** 110.
- Raz, T., R. Barnes, D. Dvir. 2003. A Critical Look at the Critical Chain Project Management. *Project Management J.* **34**(4) 24-32.
- Schonberger, R.J. 1981. Why Projects Are Always Late: A Rationale Based on Manual Simulation of a PERT/CPM Network. *Interfaces* **11**(5) 66-70.
- Sharp, J. 2008. How to be the ultimate taskmaster. National Public Radio. 30-Sept-08 www.npr.org/templates/story/story.php?storyId=95212407.
- Shenhar, A.J., D. Dvir. 2007. *Reinventing Project Management*, Harvard Business School Press.
- Srinivasan, M.M., W.D. Best, S. Chandrasekaran. 2007. Warner Robins Air Logistics Center Streamlines Aircraft Repair and Overhaul. *Interfaces* **37**(1) 7-21.
- Trietsch, D. 2005. Why a Critical Path by Any Other Name Would Smell Less Sweet? Towards a Holistic Approach to PERT/CPM. *Project Management J.* **36**(1) 27-36.
- Walton, M. 1986. *The Deming Management Method*. Perigee Books, New York.
- Wheelwright, S.C., K.B. Clark. 1992. Creating Project Plans to Focus. *Harvard Bus. Rev.* **70**(2) 70-82.
- Wiest, J.D. 1964. Some properties of schedules for large projects with limited resources. *Oper. Res.* **12**(3) 395-418.
- Woepfel, M. 2005. *Projects in Less Time: A Synopsis of Critical Chain*, BookSurge Publishing.
- Yeo, K.T., J.H. Ning. 2006. Managing uncertainty in major equipment procurement in engineering projects. *Eur. J. of Oper. Res.* **171**(1) 123-134.