Critical Chain: A New Project Management Paradigm or Old Wine in New Bottles?

Thomas G. Lechler, Stevens Institute of Technology
Boaz Ronen, Tel Aviv University
Edward A. Stohr, Stevens Institute of Technology

Abstract: In this paper we analyze the Critical Chain (CC) approach to managing projects. Is CC as some authors assert, one of the most important breakthrough for project management since the introduction of the Critical Path concept (CP) or does CC merely consist of known concepts presented in a different way? Our discourse compares systematically CC and CPM on three conceptual levels to reveal the differences between the two approaches. We conclude that the philosophy behind the CP and CC approaches is remarkably different resulting in a different mindset for managers and a different set of management practices. The main difference is the application of the Theory of Constraints (TOC) in the CC case. As a result, CC focuses at improving the systems performance by laying out specific policies many of which are focused on resource management especially in multiproject environments that are not explicitly addressed by CP. We conclude that while the application of CC is complex, many of its ideas can be easily adapted by practicing managers.

Keywords: Critical Chain, Theory of Constraints, Buffer Management, Critical Path Method

EMJ Focus Areas: Program & Project Management

Ever since Goldratt introduced critical chain (CC) in his book of the same name in 1997, the concept has been widely discussed in the project management literature and project management community. Some authors see CC as the most important breakthrough for project management since the introduction of the critical path method and refer to CC as the direction for project management in the 21st century (Steyn, 2002; Newbold, 1998). Others question its innovativeness and argue that it consists of known concepts presented in a different way (Maylor, 2000; Raz et al., 2003; McKay and Morton, 1998).

In the last few years, several books have been published explaining the concepts underlying CC (Newbold, 1998; Leach, 2000) and a number of software packages based on CC scheduling concepts have been developed (Prochain, 1999; Scitor, 2000). Many examples of successful applications of CC have been cited in the literature (Leach, 1999) and on websites (Product Development Institute, 2005). A number of researchers have discussed the concepts underlying CC and the differences between CC and CP at a conceptual level (Raz et al., 2003; Globerson, 2000). Other researchers have focused on the technical aspects of CC scheduling using simulation analyses (Herroelen and Leus, 2001; Cohen et al., 2004). Although these studies are helpful, we share the view of Herroelen and Leus (2001 and 2002) that the discussions on both sides of the CC debate are often too general to offer guidance on CC's advantages and disadvantages relative to established CP

About the Authors

Thomas G. Lechler is associate professor at the Wesley J. Howe School, Stevens Institute of Technology. His research focuses on the early development stages of new ventures and the success factors of project management to understand the dynamics and interactions between decisions, structures, and behaviors on innovation success. He has published articles in leading international and German journals and authored two books in the fields of project management and entrepreneurship and reviews regularly for several technology and innovation management journals. He holds a PhD from the University of Karlsruhe, Germany.

Edward A. Stohr is associate dean for research at the Wesley J. Howe School, Stevens Institute of Technology. Prior to joining Stevens, Professor Stohr was a faculty member at NYU's Stern School of Business for more than 20 years. His research focuses on the problems of developing computer systems to support work and decision-making in organizations. He has published articles in many leading journals. He is the co-editor of three books in the field of information systems and is on the editorial boards of a number of leading journals. He holds a PhD from the University of California, Berkeley.

Boaz Ronen is a professor of technology management and information systems in Tel Aviv University, the Leon Recanati Graduate School of Business Administration. He holds a BSc in electronics engineering from the Technion, Haifa, Israel, and an MSc and PhD in business administration from Tel Aviv University, Faculty of Management. His main areas of interest are focused on increasing shareholders' value and the application of focused management techniques and philosophies. He has published more than 100 papers in leading academic and professional journals, and co-authored a book on Value Creation, Managerial Decision Making and Cost Accounting.

Contact: Thomas G. Lechler, Stevens Institute of Technology, Castle Point on Hudson, Hoboken, NJ 07030; phone: 201-216-8174; fax: 201-216-5385; tlechler@stevens.edu

Refereed management tool manuscript. Accepted by Special Issue Editor.

concepts; thus, the conflicting opinions on the relative merits of the CC approach are not surprising.

The failure to understand lean manufacturing (TQM, JIT, Kanban, etc.) as a coordinated system of management practices led to many problems when western manufacturers tried to emulate Japanese techniques in the early 1980s (Ronen and Starr, 1990; Womack et al., 1991). We believe that the inability to conceive CC as a coordinated set of ideas and practices leads to similar problems in evaluating and understanding CC. The goal in this article is to introduce CC to practitioners through the development of a framework for understanding and evaluating the two competing approaches to project management.

Our discussion is structured in five main sections. First, we derive the basic framework of our analysis. The philosophical level is then covered, followed by the single project and multiproject cases, respectively. Next, we summarize the discussion by comparing the management practices associated with CP and CC and examine the problem of transitioning from one set of management practices to the other. This leads to the identification of important management concepts from CC that we believe can be applied independently of whether CC or CP is chosen as the underlying methodology. We conclude the article with a brief overview of future research questions.

Conceptual Framework

The first efforts to define a project management methodology were based on CP networks applied to unique technical tasks such as the construction of bridges, tunnels, buildings, etc. during the 1950s (Wiest, 1969; Pinto, 1999). The principles of network techniques were further developed, and management practices and standards were added during the next few decades to provide an organizational environment to improve the execution of projects. The Apollo program in the 1960s and 70s was perhaps the first to define and standardize the organizational configuration and leadership side of managing projects (Morris and Pinto, 2004). Since its introduction, CP has not been significantly modified (Shou and Yeo, 2000). The need for a new approach to project management is motivated by the fact that CP frequently fails and that even expensive software is not able to improve the situation (Rand, 2000).

To date, most project management theory and practice has focused on single projects in which it has been assumed that the main goal of responsible management is to implement each project within given budget, time, and scope constraints. This single-project focus has been criticized by several authors because projects are now more pervasive within organizations and the problem of simultaneously managing multiple projects is a major concern (Pinto, 1999; Morris and Pinto; 2004). Another weakness of current project management theory and practice is in the area of resource management. This is true despite the fact that the issues of resource leveling and resource conflicts are dealt with every day by managers and have been extensively researched since they were first discussed by Wiest (1969). A key challenge for project managers is to cope with the complexity of resource management, especially in multiproject environments where contention for scarce resources can also be a major political concern.

We recognize the complex relations between project management concepts and their implications for the day-to-day management of projects. Following the approach of Ronen and Starr (1990), who analyzed, among other issues, the fundamental differences between Just in Time (JIT) and Optimized Production Technology (OPT) management on a philosophical and tactical level, our analysis of CP and CC is conducted on two levels: the philosophical and the operational. On the philosophical level, the related theories and their implications for CP and CC are differentiated and compared. The operational level is divided into two areas of discussion: on the single project level, issues related to the planning and controlling of a single project are analyzed; on the multiproject/program level, we focus on the relations between a single project and other projects. The identification of the conceptual differences at different levels of abstraction allows practitioners as well as researchers to evaluate the strengths and weaknesses of the two approaches to project management.

Each level is analyzed using the following basic perspectives:

- Theory (philosophical level only)
- Goals
- Focus of Attention
- Uncertainty
- Resource Management
- Behavioral Issues
- Metrics (operational level only)
- Execution (operational level only)

Note that our framework includes separate perspectives for *Goals* and *Focus* of Attention. We do this because the CP and CC paradigms suggest that managers focus their attention on different aspects of projects in order to achieve somewhat different project goals.

We confine our discussion to situations where available resources are limited and the demand for them could exceed their availability (Patrick, 1998). To avoid bias, we assume the best possible implementation of each paradigm and an environment where sound project management practices are possible.

Philosophical Level

At the philosophical level, we compare the theoretical basis and underlying assumptions of CP and CC. This level is often not explicitly considered in the literature but it is crucial to understanding the methodologies at the operational level. Exhibit 1 compares different aspects of the philosophies of CP and CC that will guide our discussion in the remainder of this section.

Theory. Both CP and CC rely on systems and graph theory. Traditional CP and CC differ primarily because the latter applies TOC concepts to project management. TOC requires first that the goal of the entire system be identified. Applied to a single project, CC identifies on-time performance as the primary goal; applied to a multiproject environment, total systems throughput is identified as the goal. There are five focusing steps in TOC as developed by Goldratt and Cox (1986) and Goldratt (1990) and applied to project management (Goldratt, 1997; 1998). These are as follows:

Identify: Find the constraint that limits system performance. In the case of production management, this means finding the weakest link in the chain—the resource or workstation that is the bottleneck. Applied to a single project, this means identifying the critical chain: the critical chain is defined as the longest chain of tasks that satisfies both precedence and resource constraints. Applied to a multiproject environment, this means identifying the bottleneck resource(s) that involve most cross-project utilization.

Exploit: Improve systems' performance using existing resources. In the single project case, this means focusing on the activities in

Exhibit 1. Philosophical Differences Between CP and CC

Perspective	CPM/PERT	Critical Chain
Theory	• Systems Theory, Graph Theory	• Systems Theory, Graph Theory, Theory of Constraints (TOC)
Goals	• Minimize duration of single project under resource constraints • Satisfy the triple constraints of time, cost, and scope	• Minimize duration of single project under resource constraints • Maximize project throughput in multiproject environments • Satisfy the triple constraints of time, cost, and scope with special emphasis on meeting the due date • Adopt a satisficing approach
Focus of Attention	• Single project perspective (primarily) • Set a project completion time and determine which activities require particular attention to avoid delaying project completion • Local systems perspective	• Systems perspective—both single and multiple project environments • Set a project completion time and determine, under explicit consideration of uncertainty, which activities require particular attention to avoid delaying project completion • Global systems perspective
Uncertainty	• Contingency plans to protect against external events based on risk analysis and Monte Carlo simulation • Local protection against uncertainty • Tradeoffs between the triple constraints	• Contingency plans to protect against external events based on risk analysis and Monte Carlo simulation • Global protection against uncertainty • Tradeoffs between the triple constraints are not emphasized; CC attempts to avoid the need for tradeoffs
Resource Management	• Solve the Resource-Constrained Scheduling Problem (RCSP) to develop a baseline schedule • Maximize utilization of all resources	• Solve the RCSP to develop a baseline schedule (as for CP but including buffers) • Maximize utilization of the bottleneck resource(s)
Behavioral Issues	• The human-side of project management only implicitly addressed	• Reduce activity times to counteract individual tendencies to delay task execution (Parkinson's Law and Student Syndrome)

the critical chain to ensure that work is performed efficiently and without delays. In the multiproject case, this means managing the deployment of the critical resources—first, by prioritizing projects and, second, by avoiding multitasking so that a bottleneck resource completes all of its work on one project before moving on to the next lower-priority project.

Subordinate: Use slack or overcapacity in non-bottleneck resources (i.e., subordinate them) in order to improve the performance of the bottleneck resource. In CC, the emphasis is on reducing the uncertainty in due date performance. Applied to a single project, this means that non-critical activities must not interfere with or delay work on critical activities. Subordination in the multiproject case means that non-critical resources may, at times, be left idle to ensure high utilization of the bottleneck resources across the projects.

Elevate: If system performance is unsatisfactory after taking the above steps, increase the capacity of the total system focusing first on the bottleneck constraint. In both the single and multiproject cases, this might mean investment in additional resources. Naturally, the focus will be on increasing the capacity of resources that most impact the critical chain or total systems throughput. Alternatively, elevating system capacity might mean investing in IT infrastructure, additional management training, etc. In certain cases, elevating the system constraints may be carried out by the offloading mechanism, i.e., assigning some of the CC tasks to non-CC resources/activities.

Unlike CP, CC makes a distinction between critical and non-critical resources. CC puts a lot of attention on managing the critical resources and planning mainly according to these resources. CP treats the resources as a less important issue that should be subordinated to the critical path planning, without an explicit distinction between a bottleneck and a non-bottleneck resource.

Goals. In the CP world, the initial project schedule is designed to minimize project duration under resource constraints. A second important goal is to satisfy the "triple constraints" of time, cost, and performance on a single project (Umble and Umble, 2000). It is recognized that tradeoffs between these three project objectives are often made—for example, on-time performance might be achieved by reducing the scope of a project. It is noteworthy that more general objective functions that take into account the net present value of completing projects or that explicitly take risk into account have not found much acceptance in practice despite active research in both areas (Vanhouke, Demeulemeester, and Herroelen, 2001).

In contrast to CP, CC directly addresses the multiproject case as well as the single project case. In the CC world, the emphasis is on initially reducing the scope of the projects as part of a focused management approach (Pass and Ronen, 2003). Once the scope has been refined to only essential elements, the emphasis shifts to on-time performance and throughput in the scheduling and execution phases of project management. Satisfying the triple constraints is as important in CC as in CP. To some extent, the scope constraint is addressed by the initial focusing step. While cost is, of course, important, good cost performance is thought of in the CC world as a corollary of high throughput performance.

Acknowledging the inherent complexity of project management, CC takes a "satisficing" (Simon, 1956) approach both to the development of the baseline schedule and the management of projects during the execution phase as explained in the next section (Goldratt, 1997). This satisficing approach, it is argued, is the best one can do in the face of the enormous complexity and uncertainty of project management in real environments. The satisficing approach is evident in the recommendation by CC proponents that it does not matter if there is more than one critical chain—just choose one and then protect it from being superseded by another critical chain during execution (Goldratt, 1997). It is also evident in the recommendation to focus on managing the one bottleneck resource in multiproject situations. These are simple remedies that have the advantage of helping managers focus on essentials even when the real world becomes overwhelmingly complex. The focusing and simplifying perspective of CC may provide real advantages; however, this assertion should be tested at both the theoretical and practical level.

Focus of Attention. In conventional CP, management attention is primarily focused on the performance of single projects to meet the triple project goals of time, cost, and scope. Management focus is directed to managing the activities on the critical path—the longest path though the project network. The focus in CP on efficiency of single projects leads to local, rather than global, optimization in multiproject situations.

In contrast to CP, CC focuses explicitly on both the single project and the system as a whole—i.e., on global efficiency, that is more than the sum of local efficiencies. In the case of a single project, management focus is directed to managing activities on the critical chain, in which both resource and precedence constraints are considered important. A unique contribution of CC is the guidance it provides for improving performance in situations where multiple projects share scarce resources.

In the multiproject case, an attempt is made to maximize throughput by imposing a "throughput" metric, by managing the interaction of multiple projects, by managing system-wide critical resources, and through the management discipline involved in prioritizing projects.

Uncertainty. The uncertainty and risk inherent in projects has been a major issue throughout the history of project management. To estimate risk, Monte Carlo simulations of project networks were developed in the 1970s and stochastic network analysis software such as the Graphical Evaluation Review Techniques (GERT) was introduced (Taylor and Moore, 1980). Because estimating activity probability distributions is conceptually difficult, these concepts did not find general acceptance. Currently, in traditional project management, uncertainty and risk are recognized by the development of contingency plans and risk analyses (PMI, 2004). The safety margins built into individual activity estimates and the float in non-critical individual project activities can be used to buffer the project against variation in non-critical path activities (Globerson, 2000). Uncertainty can also be managed by tradeoff decisions between the three fundamental project goals of time, cost, and scope.

The above approaches to handling risk and uncertainty are also valid in CC; however, a fundamentally different approach is also introduced. CC proponents argue that individual activity estimates are almost always padded by the introduction of a safety margin that will give the activity duration a high probability of being met. Goldratt therefore proposes that the safety margin be removed from the individual activities and pooled in global buffers (Goldratt, 1997).

Resource Management. Resource management is fundamentally important in both the CP and CC approaches. While CP focused initially on resolving precedence constraints, the need to recognize and avoid resource conflicts was recognized early (Wiest, 1969). The Resource-Constrained Scheduling Problem (RCSP) (see Herroelen et al., 1998) is essentially the same for both CP and CC; however, CC's explicit focus on resource management is a key difference between the two approaches to project management. In particular, consistent with its foundation in TOC, CC urges managers to identify and manage the system's "bottleneck resource" in multiproject environments.

Behavioral Issues. A growing literature addresses the problems of poor project performance by investigating the human side of project management (House, 1988; Lynn and Reilly, 2004). This area of research applies equally to CP and CC; however, CC attempts to remove some of the sources of human conflict by designing the management system to perform more efficiently and by avoiding conflicts over resources. To achieve this, CC adds several new behavioral concepts. The first behavioral concept was mentioned above—namely, the replacement of local safety by global buffers and drastically cutting activity time estimates to achieve better on-time performance and throughput; however, this part of recommended CC management practice is very controversial. Will workers "game" by doubling the initial size of their estimates (McKay and Morton, 1998)? Shouldn't smart project managers insist on "crashed" project activity times regardless of whether they are in a CP or CC environment?

Other recommendations of the CC approach also have behavioral implications. As mentioned above, a key challenge is to avoid pressures on resources to multitask. This is particularly true in multiproject environments where different project owners exert pressure to have their project executed first (Patrick, 1998a, 1998b). Another behavioral implication of CC is its throughput orientation, which supposedly encourages managers to think globally rather than locally (Rand, 2000.)

A final behavioral issue is the accountability for the various activities. CP focuses on meeting due dates of local activities. This enables meeting due dates and controlling the schedule. On the other hand, CC focuses on the whole project's due date, and manages the schedule by monitoring the project buffers. This requires a huge behavioral change and a paradigm shift from a local to a global perspective, and from one's own accountability to common goal accountability.

Operational Level: Single-Project Case

In this section, we compare CP and CC practices in the planning and execution phases for the single project. Before we discuss the conceptual differences between CP and CC, we provide a concrete example of the development of a baseline schedule using both approaches.

Example: Developing a CC Baseline Schedule. Exhibits 2 and 3 illustrate the differences between the CP and CC approaches to developing the baseline schedule for the same project (based on an example in Herroelen and Leus, 2002). The development of a CC schedule follows the five steps of the TOC.

In the first step, the longest path is identified as the critical chain (CC) after resource conflicts are solved. This path is equivalent to the resource dependent CP and describes the constraint of the project.

The second step exploits the system's bottleneck by removing safety margins from individual activities on the CC and adding a project buffer at the end of the critical chain to provide a global safety margin. A project buffer is essentially a period of time by which the estimated project duration is extended to allow for uncertainty. The total project safety time can be reduced relative to the CP approach because of risk pooling effects as in insurance (Steyn, 2000). Thus, CC proponents argue that there is ample safety margin built into projects; however, it is in the wrong place—at the activity level rather than the project level (Steyn, 2000). In addition, CC attempts to build stability into project execution by protecting the critical chain from change using feeding and resource buffers in individual projects and drum and capacity buffers between projects as explained more fully below. In our example the durations of the activities in Exhibit 2 have been reduced in Exhibit 3 to 50% of the original size rounded up. In Exhibit 3 a project buffer equal to one half of the length of the critical chain has been added at the end of the CC baseline schedule.

In the third step, the remaining paths are subordinated to the constraint. In Exhibit 3, feeding buffers equal to one half of the associated non-critical path length are introduced and non-critical chain activities have been shifted to their latest start date.

The fourth step, requiring the elevation of the constraint, is not directly shown in the baseline and depends on the decision-makers to add more capacity to the systems constraint.

Exhibit 2. Baseline Schedule Using the Critical Path Approach

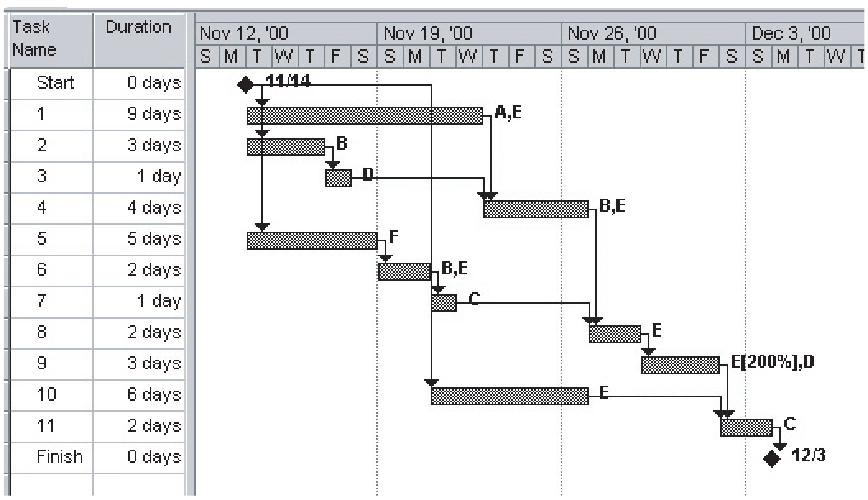
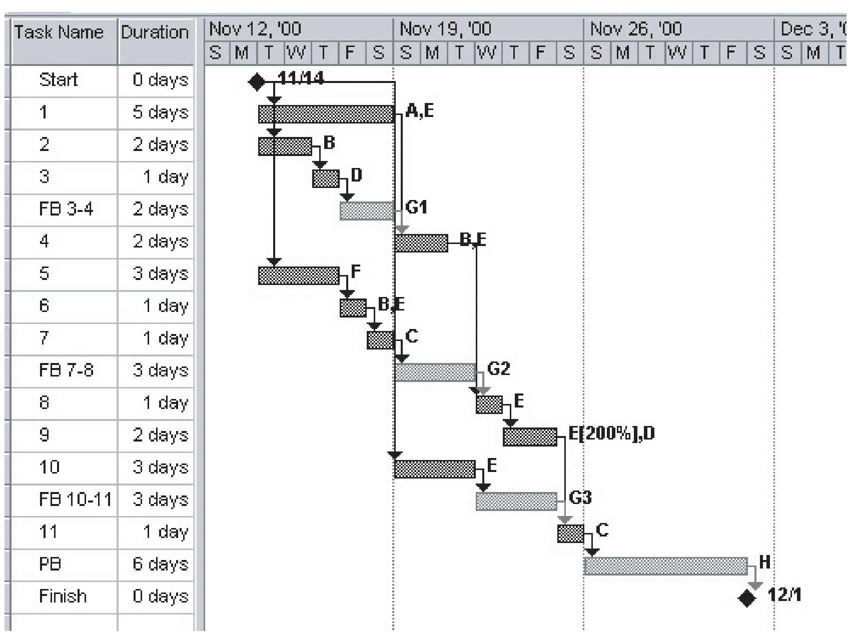


Exhibit 3. Baseline Schedule Using the Critical Chain Approach



The critical path in Exhibit 2 consists of activities 1, 4, 8, 9, and 11 with a planned project duration of 21 days. In Exhibit 3, by chance, the critical chain consists of the same activities but the planned project duration is now 18 days. Our example is illustrative only; we do not assume any particular activity probability distribution and make no choice between the average and median activity estimates.

Planning Phase: Scheduling Single Project. Exhibit 4 compares the CP and CC approaches to developing the initial baseline schedule for a single project.

Goals. In the planning phase, both CP and CC attempt to provide a precedence and resource feasible schedule of minimum duration. It is also the goal of both CP and CC to protect the target date for the project; however, as explained below in the "Scheduling and Rescheduling" section, CC has a distinct approach to achieving this objective.

A second goal of the CC approach is to reduce work-in-process (WIP) defined as the amount of work currently in progress on the project. According to TOC, WIP can be reduced by reducing lot sizes in manufacturing environments. Applied to project management, this means reducing the size of the scheduled activities. In one company studied by the authors, the rule was to reduce the work assignments to no more than 200-300 hours (Lechler, Ronen, and Stohr, 2005). Another way to reduce WIP is to schedule "gating activities," those succeeding project milestones or having no predecessor activity, at their latest start date after insertion of appropriate downstream feeding buffers (Goldratt, 1997). The latest start date approach has two further effects: it reduces uncertainty and controls behaviors because, once a gating activity is started, there is no opportunity for procrastination.

Focus of Attention. While management attention in CP focuses on finishing the activities on the CP in the allotted time in order to meet the project due date, CC requires managers to focus on the critical chain with its emphasis on resource interactions.

In CP, calendar dates for project milestones are identified in the planning phase and achieving the milestones is an important goal during the execution phase. CC avoids the use of calendar-bound project milestones except for the project due date and other milestone dates that might be required to coordinate with external contactors. The major reasons for avoiding calendar-bound milestones in CC are related to behavioral issues and the fact that calendar bound milestones need extra buffers. The planning focus is to float milestones whenever possible.

Uncertainty. Because projects are unique and innovative undertakings, it is an important management challenge to keep the due date once it has been defined. In the CP baseline schedule, the project due date can only be protected against uncertainties by including safety margins in individual activity duration estimates. In CP schedules it is possible that the CP changes. Project planners are aware of this and try to identify the criticality of network activities (Bowers, 1995). Activities with high criticality could be protected by extending them with a safety margin. Another possibility to protect the CP against changes is to create float by starting activities as early as possible. The problem is that float cannot be intentionally positioned within the network.

In CC, as mentioned previously, safety margins are aggregated into a "project buffer" placed at the end of the project, that protects the promised due date against uncertainty. The critical chain is protected by feeding buffers (the third step of TOC) that

Exhibit 4. Differences Between CP and CC Planning at the Single Project Level

Perspective	Critical Path	Critical Chain
Goals	• Minimize project duration • Protect the due date	• Minimize project duration • Use buffers to protect the due date • Minimize work-in-process (WIP)
Focus of Attention	• Critical Path • Identify calendar dates for project milestones	• Critical Chain • No project milestone calendar dates except where externally imposed
Uncertainty	• Activity estimates might contain safety margins • No project buffer • CP protected to some extend by float • Schedule activities at their early start time	• Remove safety margins from activity estimates • Aggregate safety margins on the critical chain into a project buffer • Add feeding buffers where non-critical paths join the critical chain • Schedule activities at their latest start times to reduce WIP
Resource Management	• Determine a precedence and resource feasible baseline schedule	• Determine a precedence and resource feasible baseline schedule
Scheduling	• Solve the RCSP problem to resolve resource conflicts and estimate the Critical Path	• Solve the RCSP problem to resolve resource conflicts and estimate the critical chain • Use as late as possible start dates for the activities • Introduce project buffers and feeding buffers
Behavioral Issues	• Activity estimates might contain safety margins	• Avoid the student syndrome and Parkinson's law

are placed wherever a non-critical chain activity leads into a CC activity. Float is minimized by starting non-critical path activities at their late start date. This delay has a number of implications besides reducing WIP in the early stages of the project. The additional time to start the gating activities and their successors may be useful if there are initial uncertainties associated with the project. Of course, the project management maxim that risky activities should be scheduled for early starts is in contradiction to this recommendation—instead, buffers are added to protect the schedule against possible risk.

The above measures to protect the CC may make it more stable than the CP but this requires empirical verification.

Resource Management. The first step in resource management for both CP and CC is to determine a baseline schedule that is both precedence and resource feasible as explained above. Resource buffers are introduced in CC in order to ensure that resources are available for activities on the critical chain. These are usually just warning signals—resources are warned well in advance of the time that they will be needed to work on an activity in the critical chain. The same warnings would be sent in a well-run CP system so, although resource buffers are emphasized in CC, they are not a new idea. Alternatively, resource buffers can be introduced through explicit time delays similar to feeding buffers (Leach, 2000). Another problem is that the duration of a buffer is tied to the resource capacities on the path leading to the buffer. For example, if the activities on a critical chain are assigned to different resources, it is not clear which part of the buffer duration covers variability in each resource. This has implications for the calculation of required resource capacities.

Scheduling. An enormous body of research addresses the problem of obtaining a minimal baseline schedule by taking both precedence and resource constraints into account—the so-called resource-constrained scheduling problem (RCSP) (Herroelen et al., 1998). Computer packages, such as MS Project for CP and Prochain or PS8 for CC, routinely produce schedules that are "resource leveled." MS Project will provide an optimized baseline schedule in the CC case if the feeding buffers are entered as activities requiring zero resources (Herroelen et al., 2002); however, as the authors point out, these packages sometimes produce non-minimal initial schedules for even small problems. As the project network size increases, or if we move from the single project to the multiproject case, RCSP becomes less tractable (Demeulemeester and Herroelen, 1997, 1998). Consistent with CC's satisficing approach, Goldratt (1997) offers a simple, but poorly specified, manual heuristic to solve the RCSP problem in which non-critical paths are pushed back until resource conflicts are avoided (Leach, 2000).

There is always a possibility that the insertion of a feeding buffer into a non-critical path will make the resulting path longer than the critical chain (Leach, 2000). This means that non-critical activities have to be started before the first activity on the critical chain or that time gaps need to be introduced into the critical chain. This leads to a violation of the definition of the critical path, which requires that those activities that are started first have to be on the critical path.

One of the most controversial issues in CC is the buffer calculation. The project buffer is conventionally set at one half of the length of the critical chain. This is based on the rough approximation that activity estimates normally have about 50% safety margin included in them and that the safety in each activity should simply be aggregated and transferred into the project buffer (Leach, 2000). Goldratt (1997) derives this calculation from the fact that in a beta-distribution the difference between the 90% probability and the 50% duration estimate probability is approximately 50% of the duration that represents one standard deviation.

The implications of this discussion are quite profound. The CC method results in a shorter baseline schedule and an organized approach to protecting the schedule against uncertainty. Depending on the assumptions made, the CC baseline schedule duration is on average between 10% and up to 30% shorter than the corresponding CP baseline schedule as we also demonstrated in our example in Exhibit 3. This seems to be the best of both worlds; however, the assumption that all activity estimates follow the same probability distribution is crucial. Also, a number of authors assert that the 50% estimate for the project buffer overstates the required buffer size.

Another criticism is that the critical chain may not be stable during execution. While the critical chain takes both resource and precedence dependencies into account, the feeding buffers that are supposed to protect the critical chain are based only on the network topology (precedence relationships). This may be a problem as a delay in the execution of the non-critical chain activity may set off a cascading effect that delays the start of a critical chain activity (see Herroelen and Leus, 2001).

Behavioral Issues. The planning tactics of CC assume that behaviors can be modified. Goldratt (1997) addresses two specific behaviors that increase the lead times of projects. They are the *student syndrome*, meaning that humans with time buffers start their tasks later and waste safety margins, and *Parkinson's Law*, meaning that humans tend not to finish their tasks ahead of time even though they have the chance to do so. According to Goldratt, activity estimates are often padded to increase the likelihood of on-time performance to 90% or more (Goldratt, 1997). To avoid both behavioral problems, Goldratt recommends that activity times should be reduced to their median estimates or 50% likelihood of successful completion. To make 50% activity estimates more acceptable, a further tenet of CC is that the actual start and finish times of individual activities are not monitored during project execution. This is designed to relieve the pressure on individuals performing activities and to promote acceptance of the idea that one half of the time activities will overrun their estimated durations. Furthermore, since activity start and finish times are not adhered to, activity performers do not wait for the scheduled activity start times; rather, the "next" activity is begun as soon as the previous one is finished. Also, to avoid the student syndrome, slack time is minimized by using as late as possible activity starts. The CC requirement that activity times should be drastically reduced is controversial. Cases available to the authors document the behavioral problems with the rigorous reduction of activity time estimates. It has yet to be proven that Parkinson's Law and the student syndrome have a strong influence on the lead times of projects.

In CC, calendar-based milestones are also avoided. It is contended that Parkinson's Law and the student syndrome tend to make milestones "self-fulfilling prophecies." This leads to late start of activities resulting in higher risk of delays and to a loss of time advantages because early finish dates are not reported.

Execution Phase: Monitoring and Controlling a Single Project. Most theoretical work has focused on the planning phase and the development of what we have called the baseline schedule. The differences between CC and CP with regard to the execution phase of a project are shown in Exhibit 5. The goals are the same for the planning and execution phases and will not be discussed further here.

Focus of Attention. In CP, the focus is on expediting activities on the critical path in order to meet the estimated calendar dates and to meet the calendar dates of the project milestones. Delays of single critical path activities or milestones have to be avoided and specific action has to be taken to compensate for these delays. In CC, activities are not planned to start and finish on specific calendar dates. Instead, the focus is on the penetration of the project and feeding buffers as discussed next. Specific decisions are only necessary for those activities with over proportional buffer consumption rates. These activities could be on the critical chain or on feeding paths since an important focus is to avoid a change of the critical chain.

Uncertainty. Uncertainties occurring during project execution are treated in CP by exploiting the available float of non-critical activities and by making trade-off decisions between budget, scope, and schedule for critical activities. In CC, uncertainties are directly covered by buffers. Rescheduling decisions have to be made only if one or more of the buffers are exhausted. Also resource buffers are used to allow a direct continuation of succeeding activities without any delay; this practice reduces the uncertainty involved in WIP estimation.

Resource Management. In CP, resources are coordinated along the critical path. When critical activities are delayed and impact the critical path, more resources can be assigned directly to these activities or to other succeeding activities. This is also called activity crashing. In CC, the resources are coordinated using the status of buffers. The resource buffer warning mechanism explicitly introduced in the CC baseline schedule is not a new idea as practitioners also make use of this idea in CP. CC adds a further stricture—avoid multitasking or switching resources from one task to another. CP does not explicitly address the issue of multitasking.

Execution and Rescheduling. There is an interesting gap between the baseline schedule plans and what actually happens once a project begins execution. In the CP world, it is often the case that the formal plans are not updated because this can be a time-consuming task. In the CC world, calendar dates of activity start and finish times are monitored but, due to the buffers, updating the plans would seem irrelevant. Monitoring the buffers should be sufficient—as long as the critical chain does not change! If everything went according to plan, the baseline schedule would be the only schedule that is needed. In practice, at regular intervals (e.g., weekly project meetings) and as activities are completed and resources released, resource allocation decisions have to be made that react to the current situation. In essence, the project is either implicitly or explicitly rescheduled. By the former, we mean that some heuristic such as management judgment, min-slack, or earliest due date is used to choose the next activity to be executed, which activities need to be expedited, and so on. In the CP world, ceteris paribus, critical path activities are given the higher priority and in the CC world, activities on the critical chain have higher priority. Both approaches provide good heuristics (Cohen et al., 2004).

By explicit rescheduling, we mean that an "optimal" project plan going forward is computed (Herroelen and Leus, 2001, 2004). Of course, completely recalculating an optimal plan periodically is optimal in a theoretical sense; however, the transaction costs—the costs of communication, coordination, and renegotiating

Exhibit 5. Differences Between CP and CC: Single Project Execution and Controlling

Perspective	Critical Path	Critical Chain
Focus of Attention	• Manage to the calendar dates of the critical path activities • Meet project milestones	• Keep the baseline schedule and critical chain fixed during execution
Uncertainty	• Use available float • Trade-off decisions between budget, scope, and schedule	• Buffer management
Resource Management	• Coordinate resources along the CP • No explicit position on multitasking	• Coordinate resources by heeding buffer warnings • Avoid multitasking
Execution and Rescheduling	• No single guideline—many heuristics	• Use road-runner paradigm—execute activities as soon as feasible—except for gating activities
Monitoring Metrics	• Monitor and report activity start and finish times • Monitor progress towards project milestones • Earned value (EV) reporting	• No activity due dates • Report penetration of buffers • No project milestones except where externally imposed • EV reporting difficult but not excluded
Behavioral Issues	• Activity performers are held responsible for timely activity completion	• Responsibility for activity delays not clarified

with suppliers when there are frequent changes in plans—can be prohibitive. These latter costs should be taken into account in the planning problem formulation, but this does not seem to occur in practice.

As shown in Exhibit 3, CC schedules place gating activities at their latest start dates. During the execution phase, an attempt is made to maintain the planned start dates for these gating activities. For all other activities, the planned start dates are ignored. Instead, a "road-runner" (Herroelen and Leus, 2002) or relay race strategy is followed. Under this execution strategy, as activities are completed, handoffs are immediately made to eligible succeeding activities. If things go well, the project may then be completed ahead of schedule because activity performers do not wait to start an activity until its planned start date.

Monitoring Metrics. Setting performance objectives, monitoring performance, and providing feedback to activity performers and project members is always important. The dynamic nature of projects makes it particularly important in project management. Probably the most dramatic difference between the two approaches is that CP monitors and reports activity start and finish times and performance against calendar fixed due dates while CC does not. Instead, CC monitors the project and feeding buffers. A simple "green-yellow-red" warning system is recommended (Goldratt, 1997). If overruns on activities leading up to a buffer cause a buffer penetration such that the ratio between the available buffer and the minimum required buffer drops by more than 20%, a serious effort must be made to correct the problem to preserve due date performance.

In concept, this is an attractive monitoring system. The number of buffers is less than the number of activities, thus simplifying the management system; however, some of our case studies indicate that tracking the buffers is complex and time consuming. This has led the authors to speculate that it might be sufficient simply to monitor the project buffer (Lechler et al., 2005); however, this cannot be confirmed without extensive research.

Another difference between CP and CC is that project performance is monitored in CP by achieved milestones. Because milestones are fixed calendar dates, CC only introduces them if they are externally imposed. Because of the buffers, milestones are not used to monitor project progress in CC.

In CP, the earned value (EV) method provides metrics to monitor the progress of projects (PMI, 2004). EV requires that progress on individual activities be tracked. The EV is then computed as the sum of the originally estimated costs (value) of the completed activities and pro-rated costs of partially completed activities divided by the total estimated cost of the project. EV provides a useful metric but is essentially backward looking: there are possibilities to make forecasts but these are extrapolations of past progress. In addition, EV does little to pin-point the project activities that need attention.

The EV method is not excluded from the CC approach but, because in "pure" CC the activity finish times are not planned on a calendar basis, the necessary reference points for the EV approach are missing. Instead, in CC, project performance is monitored and controlled by observing the buffer consumption and the ratio between the currently available buffer and the minimum required buffer. Both metrics help to manage the project implementation toward due date performance. Note that estimating the buffer penetration at each time period involves forecasting future activities along the critical chain and the paths to the feeding buffers. The buffer system thus provides both a forward-looking measure of the likelihood that the due date will be met and more specificity as to the areas that need management attention.

Operational Level: Multiproject Case

Many organizations manage multiple projects simultaneously and face a continuing demand to execute new projects. Examples include software development organizations, repair shops, and maintenance facilities. The major problem in these situations is to allocate and coordinate resources across multiple projects.

Planning Phase: Scheduling Multiple Projects. The differences between CP and CC in developing baseline schedules in a multiproject environment with shared resources are summarized in Exhibit 6. The CC case is directly derived from the application of the TOC steps described earlier.

Exhibit 6. Differences between CP and CC Planning at the Multiproject Level

Perspective	Critical Path	Critical Chain
Goals	• Minimize project duration	• Maximize systems throughput.
Focus of Attention	• Performance of individual projects	• Performance of multiple project system constraint resource • Reduce WIP
Uncertainty	• Not explicitly addressed	• Introduce drum and capacity buffers
Resource Management	• Maximize resource utilization of all resources • Multitasking not explicitly addressed	• Maximize resource utilization of constraint resources • Do not allow multitasking
Scheduling	• Several project prioritization rules	• Stagger projects along the systems constraint using drum and capacity buffers • Prioritize projects • Resolve resource conflicts on the systems level
Behavioral Issues	• Not explicitly addressed	• Avoid multitasking

Goals. In the CP case, multiproject environments are not explicitly addressed. Instead, the goal is to minimize the duration of each project under consideration of shared resources. The CC approach directly addresses the system's level and its goal is to maximize system throughput, that could be defined as the number of projects completed per unit of time, or, preferably, the value created per unit of time.

Focus of Attention. Most attention in traditional project management has focused on managing single projects to meet time and cost objectives while fulfilling scope requirements. The management of multiple projects to maximize throughput has been studied by a number of researchers including Cohen et al. (2004), but this is a complex, computationally difficult problem. Using a satisficing approach, CC therefore focuses on the "bottleneck constraint"—the component of the system that limits throughput. This is also called the "drum" resource because it dictates the pace of work. Operationally, the drum resource is identified in CC as the resource that is most in demand across all of the projects. For example, in one of the cases studied by the authors, the bottleneck resource was the database design function. Alternative interpretations are possible: the bottleneck resource could be the shared resource with the greatest risk, variability, or expense; however, to the knowledge of the authors, these interpretations have not been studied. CC then proposes that managers focus on "exploiting" the constraint by making sure the bottleneck resource is used continuously without interruption.

In our interviews, CC consultants mentioned that on average between one or two constraint resources exist. It is worth noting that a bottleneck resource may not exist if each project team has its own resources in a pure project organization (PMI, 2004). Even if this is not the case, the bottleneck may be hard to identify. In one successful application of CC by a large military contractor, no bottleneck resource was identified and CC was applied independently to each of the individual projects (Lechler et al., 2005). This concept is also based on the assumption that the bottleneck resource remains stable within a multiproject system, but the demand for resources could change with every new project entering the system. This issue needs further investigation.

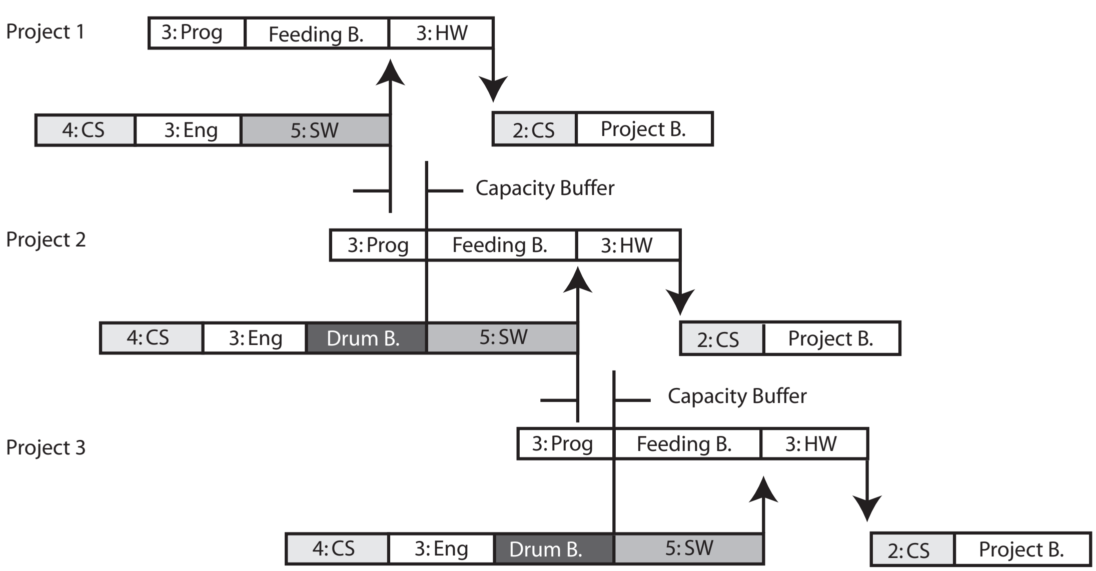
Another important focusing activity in CC involves selecting the projects based on some criterion such as expected net present value divided by expected project duration. This prioritization requires management discipline to overcome political pressures that might favor certain projects over other projects. The selection of projects is also discussed in the CP case but, without the focus on the capacity of the bottleneck constraint, there is a tendency to overload the system because the focus remains on the single project level. In CC, it is not possible to overload the system with too many projects because of the focus on the bottleneck.

Uncertainty. The start of work by a bottleneck resource could be delayed by preceding activities leading to idle time of the bottleneck and to lower the system's throughput. To ensure that the bottleneck resource is never idle, "drum" buffers (see Exhibit 7) are introduced. A drum buffer does not represent resource capacity—rather it involves starting a preceding activity that is not assigned to a bottleneck resource earlier so that the bottleneck resourse never needs to wait for a preceding activity to be finished.

Capacity buffers are introduced to ensure that performance on one project does not delay the promised due date on another project. As shown in Exhibit 7, a capacity buffer represents a possible "time delay" between the completion of work by the bottleneck resource on one project and the beginning of its work on the succeeding project.

Resource Management. The focus in managing multiple resources in CP, although not directly stated, is to achieve maximal resource utilization of all system's resources. The insight in CC is that a system's performance can only be maximized if the performance of the system's constraint is maximized. Thus, the

Exhibit 7. Drum and Capacity Buffers in Multiproject Scheduling



Notes:
SW is the bottleneck resource
Activity Notation - Time Units: Resource
Critical Chain activities shown in bold

system is synchronized by giving individual projects time slots for utilization of the bottleneck resource. Resource conflicts are solved by prioritizing the constraint resource.

CC strongly advises project managers to avoid "bad multitasking." Multitasking occurs when a resource continuously switches from activity to activity and project to project. It can be shown that such switching leads in a mechanical way to lower throughput as well as to increased set up and coordination costs (Leach, 2000). Multitasking is particularly prevalent and detrimental in the multiproject case. Goldratt (1999) showed in simulations that multitasking of resources has a significant negative impact on the due date performance of a multiproject system. This effect is also confirmed by several case studies. The avoidance of multitasking also has the effect that WIP is significantly reduced.

Scheduling. The scheduling of multiple projects under resource constraints is extremely difficult computationally. In practice, only heuristics exist to minimize multiple project durations under resource conflicts. There are two aspects to this problem. First, one has to determine priority rules for introducing new projects into the system; second, one has to allocate resources across projects.

In CP, several project prioritization rules are possible and many queuing disciplines such as highest value, earliest due date, and first-come-first-served, can be applied to determine the priorities by which projects should be accepted into the system. The problem of determining which heuristic to apply was studied by Cohen et al. (2004) who showed that system performance was not particularly sensitive to the chosen prioritization rule.

Depending on the bottleneck resource, the individual projects are ordered (staggered) in time as shown in Exhibit 7. The first project in the system has the highest priority, etc. The individual start date of a project depends on the availability of the bottleneck resource. This is a very simple heuristic that does not require complex calculations. The problem is that resource conflicts between non-critical resources are not directly addressed. This could lead to an earlier start of a second prioritized project. Also the synchronization schedule must take into consideration the fact that not all projects are consistent in the use of the synchronizing resource. This may result in occasional windows of time when the stagger is insufficient to protect other resources from peak loading and to pressures to multitask (Patrick, 1998a).

Other features of the CC schedule are the drum and capacity buffers introduced above. The calculation of the appropriate sizes for these buffers is a matter for investigation. Note that the drum has the same effect as the feeding buffers: it could push a non-critical chain of activities to an earlier start than the first activity of the critical chain. Simulation studies are needed to test the impact and need for these buffers.

Behavioral Issues. A key difference between CP and CC is the latter's identification of multitasking as a source of project inefficiency (Steyn, 2000). The pressure on individuals to multitask, that is, to simultaneously work on multiple projects is immense (Patrick, 1998b). There are a number of reasons. In the first place, project owners will, quite naturally, demand that some progress be made on their project in each time period. Second, individuals may take on multiple tasks out of enthusiasm to contribute to the organization or to impress their superiors; however, multitasking is a source of inefficiency because it delays the completion (and benefits) of some projects in its attempt to provide equal treatment for all projects. In addition, the cost and time involved in setting up to perform tasks is needlessly multiplied with a negative impact on project performance (Pinto, 1999).

To avoid multitasking, CC proponents recommend that individuals complete their work on one project before moving on to the next. This is facilitated by prioritizing the projects; however, top management leadership is needed to develop a culture in which project priorities are accepted and multitasking is seen as a source of inefficiency rather than as a corollary of keeping all resources busy. For this reason, Pinto (1999) argues that, despite its merits, avoidance of multitasking may ultimately be impossible. In our case studies, several organizations used weekly project meetings to identify and eliminate multitasking beyond about four or five tasks per individual (Lechler et al., 2005). In one of these studies, the amount of multitasking before the weekly meetings devoted attention to the problem was between 15 and 20 identifiable tasks per individual! In a study devoted to new product development projects, Clark and Wheelright (1993) argue that the optimum number of development projects assigned concurrently to a single engineer is two. Whether or not there is an optimal level of multitasking will probably be hard to establish; however, our studies indicate that an informed management can, at least, move in the right direction.

Finally, there may be a relationship between the number of the tasks that are assigned to individuals and the pressure to multitask in the disruptive sense that we have discussed in this section. This is because more activities will be completed in any time period and more project owners satisfied; however, this supposition needs further study.

Execution Phase: Monitoring and Controlling Multiple Projects. In a multiproject environment with shared resources, it is quite complex to manage projects toward due dates. All projects are connected via the available resources and delays in one project could cascade to following projects. Exhibit 8 compares the CP and CC approaches for the execution phase.

Focus of Attention. During execution, the focus of attention in CP is to avoid variation on the critical path of each individual project to maximize systems performance. In CC, the focus is on maximizing the performance of the whole system. The single project remains important, but the focus during the project implementation is on supporting and scheduling the bottleneck resource to achieve maximal throughput.

Uncertainty. The uncertainty of achieving high performance on the system's level is addressed in CC with the drum and capacity buffer concepts. These buffers are not directly controlled by the project manager—they are specified by the plan. The two buffers are designed to compensate for uncertainty while maintaining maximum possible systems performance.

Resource Management. On the CP side, concrete guidelines on how to manage resources within a multiproject environment are missing. Indirectly, the goal is to maximize the resource utilization across the whole multiproject system. On the multiproject level, CC addresses this issue by allocating additional resources to support the bottleneck resource and to maximize its performance. Any idle time of the bottleneck resource has to be avoided, but, these decisions and measures are only helpful if the bottleneck does not change over time.

Exhibit 8. Differences Between CP and CC: Multiproject Execution and Controlling

Perspective	Critical Path	Critical Chain
Focus of Attention	• Avoid variation on critical path	• Support the bottleneck resource
Uncertainty	• As for single project case	• Drum and capacity buffers
Resource Management	• Maximize utilization of all available resources	• Maximize utilization of bottleneck resource
Execution and Rescheduling	• Different prioritization rules	• Manage the total system using drum buffers • Drum buffer rope to control for new entering projects
Monitoring Metrics	• Earned value (EV) reporting No explicit multiproject metrics	• EV reporting • Systems Metrics: Number of projects finished (throughput), WIP
Behavioral Issues	• Not explicitly addressed • Accountability for due dates clearly regulated	• Avoid multitasking • Accountability not clear

Execution and Rescheduling. In situations where projects are continuously being completed and replaced by new projects, the prioritization and staggering (scheduling) must be done dynamically. This gives rise to a higher order scheduling problem involving projects rather than activities within projects. It is common practice to re-order projects in case of delays. Changes in the project priorities are critical and can lead to even further delays and lower systems throughput. In CP, incoming projects may be prioritized by different rules. Herroelen et al. (1998) showed that continuous rescheduling of the projects within the system maximizes systems performance. The problem is that rescheduling on a systems level requires the effort and input of many resources. The transaction costs for these rescheduling efforts could easily overcome the benefits of the adaptations. Furthermore, these costs could increase dramatically if the priority rules are changed.

Project prioritization in CC uses the first-in, first-served rule. All evolving resource conflicts are solved under this rule. Also, the number of incoming new projects is regulated using a concept called the drum buffer rope. This control mechanism basically connects the available capacity of the constraint resource with a stage gate controlling for the entrance of new projects. A new project is only entered if the constraint resource has or will have free capacity. In practice these coordination decisions are made by committees.

Monitoring Metrics. As stated earlier, the CP concept does not really address the multiproject level, and the EV method does not make any assertion about the performance of a multi-project system. In CC, individual projects are monitored and controlled using the feeding and project buffers as discussed for the single project case. The drum buffers are not really monitored in the same way as the project completion buffers. To control system performance, the basic metrics are WIP and throughput. Deviations in these metrics could require rescheduling the multiproject system. The CC concept does not give any guidelines as to what percentage of WIP should be allowed.

Behavioral Issues. As discussed above in the planning section, a major difference between CP and CC is that the latter clearly restricts multitasking of resources. A CP schedule ties activity due dates to definite calendar dates. This helps assign accountability to the activity performers. In the CC schedule, accountabilities are not clearly regulated. Which activity owner is responsible if buffer time is exhausted?

Conclusions and Future Directions

We have analyzed and compared the traditional CP with the newer CC approach to project management. In this analysis, we viewed each approach as an internally consistent set of management practices and beliefs. The philosophy behind the CP and CC approaches is remarkably different resulting in a different mindset for managers and a different set of management practices. The main difference is the application of TOC in the CC case. TOC focuses at improving the system's performance by laying out specific policies many of which are focused on resource management (Shou and Yeo, 2000).

Perhaps the greatest advantage of the CP approach is that it is well established. The training and infrastructure investment costs needed to change to a CC approach are considerable. On the other hand, numerous successful applications of CC testify to its value. To maintain a balanced perspective, however, we must also point out that a number of firms failed to implement CC and complained about the complexity of the CC approach in changing behaviors and expectations and managing the extra complexity of buffer management. In particular, users point to the difficulty of convincing managers of the efficacy of the approach and getting them to impose the necessary management discipline—for example, to insist on activity times with no safety margin, to impose priorities on projects, and to develop an environment that eliminates bad multitasking. Furthermore, our case studies suggest that project managers find it difficult to manage multiple buffers. For this reason we suggest elsewhere a simplified version of CC that we call CC-Lite, in which the feeding buffers are eliminated (Lechler et al., 2005).

From the analysis in this article and case studies conducted by the authors (Lechler et al., 2005), it seems that a number of TOC ideas are highly beneficial for managing projects and can be used without implementing the whole concept of CC. These are summarized in Exhibit 9.

Finally, we agree with McKay and Morton (1998) who state that a series of empirical studies are needed to clearly identify the

Exhibit 9. Simple Management Practices Implied by CC

CC Management Practice	Implementation
Manage constraint resources to avoid or solve resource conflicts	Monitor closely the constraint resource and solve resource conflicts dependent on the constraint resource
Reduce WIP	Reduce number and size of active work packets
Reduce multitasking	Hold regular project meetings, prioritize activities and ensure that no person has more than (say) three or four concurrent tasks. Avoid preemption of active tasks
Focus on total systems throughput rather than individual projects	Prioritize projects. Recognize the bottleneck resource and synchronize its use by different projects based on their priority

necessary and sufficient conditions for the TOC concept to work in project management and to test its robustness. In the discussion we identified a number of specific questions for future research:

- Is it reasonable to ask that activity durations be estimated with no included safety margin? Can this practice be sustained?
- Can the tendency for people to multitask be controlled?
- Can project managers handle the complexity of buffer management?
- Can the additional discipline and knowledge required to successfully implement CC be found in the majority of organizations?
- Is the critical chain more stable than the critical path?
- What is the best method for rescheduling projects in dynamic environments?
- What is the best way to identify the bottleneck resource in multiproject environments?
- Can the implementation of CC be simplified by eliminating the feeding buffers?

Of course, the broader and more important question is whether or not CC concepts and practices will eventually replace CP as the main paradigm for project management. Because it is difficult to change longstanding management practices, this question may take years to answer.

References

Bowers, J.A., "Criticality in Resource Constrained Networks," *Journal of the Operational Research Society 46*:1 (1995), pp. 80–91.

Clark, K.B., and S.C. Wheelwright, *Managing New Product Development—Text and Cases*, The Free Press (1993).

Cohen, I., A. Mandelbaum, et al., "Multi-Project Scheduling and Control: A Process-based Comparative Study of the Critical Chain Methodology and Some Alternatives," *Project Management Journal, 35*:2 (2004), pp. 39–50.

Demeulemeester, E.L., and W.S. Herroelen, "New Benchmark Results for the Resource-Constrained Project Scheduling Problem," *Management Science 43*:11 (1997), pp. 1485–1492.

Forsberg, K., H. Mooz, and H. Cotterman, *Visualizing Project Management, 2nd ed*., J. Wiley and Sons (2000).

Globerson, S., "PMBOK and the Critical Chain," *PM Network, 14*:5 (2000), pp. 63–66.

Goldratt, E.M., and J. Cox, *The Goal, revised 2nd ed.* North River Press (1986).

Goldratt, E.M., *Theory of Constraints*, North River Press (1990).

Goldratt, E.M., *Critical Chain*, North River Press (1997).

Goldratt, E.M., "Critical Chain, *IIE Transactions, 30* (1998), pp. 759–763.

Goldratt, E.M., *Goldratt Satellite Program*, North River Press (1990).

Gutierrez, G., and P. Kouvelis, "Parkinson's Law and its Implications for Project Management," *Management Science, 37*:8 (1991), pp. 990–1001.

Heroelen, W., R. Leus, and E. Demeulemeester, "Critical Chain Project Scheduling: Do Not Oversimplify," *Project Management Journal, 33*:4 (2002), pp. 48–60.

Herroelen, W., and R. Leus, "On the Merits and Pitfalls of Critical Chain Scheduling," *Journal of Operations Management, 19* (2001), pp. 559–577.

Herroelen, W., B.D. Reyck, et al., "Resource-Constrained Project Scheduling: A Survey of Recent Developments," *Computers and Operations Research, 25*:4 (1998), pp. 279–302.

House, R.S., *The Human Side of Project Management*, Addison-Wesley (1998).

Leach, L.P., "Critical Chain Project Management Improves Project Performance," *Project Management Journal, 30*:2 (1999), pp. 39–51.

Leach, L.P., *Critical Chain Project Management*, Artech House Proffessional Developement Library (2000).

Lechler, T., B. Ronen, and E.A. Stohr, Final Report, "NASA Strategic Multi-project Resource Management 'CC-Lite'," *NASA CPMR Project Report, Phase 1* (2005).

Lynn, Gary S., and Richard R. Reilly, *Blockbusters: The Five Keys to Developing GREAT New Products*, Harper Collins (2002).

Mabin, V.J., and S.J. Balderstone, *The World of the Theory of Constraints*, St. Lucie Press (2000).

Maylor, H., "Another Silver Bullet? A Review of the TOC Approach to Project Management," *Proceedings of the Paper Presented at the 7th International annual EurOMA Conference* (2000).

McKay, K.N., and T.E. Morton, "Critical Chain," *IIE Transactions, 30*:8 (1998), pp. 759–763.

Morris, Peter, and Jeffrey Pinto, "Introduction" in *The Wiley Guide to Managing Projects*, Morris, Peter and Jeffrey Pinto (eds.), John Wiley & Sons (2004).

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

Pass, S., and B. Ronen, "Management by the Market Constraint in the Hi Tech Industry," *International Journal of Production Research, 41*:4 (2003), pp. 713–724.

Patrick, F., "Program Management—Turning Many Projects into Few Priorities with TOC," *Project Management Institute Symposium* (1998a).

Patrick, F.S., "Critical Chain Scheduling and Buffer Management—Getting Out From Between Parkinson's Rock and Murphy's Hard Place," *PM Network, 13* (1998b), pp. 57–62.
Pinto, J.K., "Some Constraints on the Theory of Constraints—Taking a Critical Look at the Critical Chain," *PM Network, 13*:8 (1999), pp. 49–51.
Project Managemet Institute, *A Guide to Project Management Body of Knowledge* (*PMBOK® Guide*)*, 3rd ed.,* Newtown Square, PA: Project Management Institute (2004).
Prochain Solutions, Inc., *Prochain Plus Project Sceduling* (1999), http://www.prochain.com.
Product Development Institute, *Tutorial: Goldratt's Critical Chain Method: A One Project Solution* (1999), http://www.pdinstitute.com.
Rand, G.K., "Critical Chain," *Journal of the Operational Research Society, 49*:2 (1998), p. 181.
Rand, G.K., "Critical Chain: The Theory of Constraints Applied to Project Management," *International Journal of Project Management, 18*:3 (2000), pp. 173–177.
Raz, T., R. Barnes, et al., "A Critical Look at Critical Chain Project Management," *Project Management Journal, 34*:4 (2003), pp. 24–32.
Ronen, B., and M.K. Starr, "Synchronized Manufacturing as in OPT: From Practice to Theory," *Computers and Industrial Engineering, 18*:8 (August 1990), pp. 585–600.
Scitor, "Critical Chain Concepts," http://www.scitor.com/products/ps_suite/ccintro.htm (2000).
Shou, Y., and K.T. Yeo, "Estimation of Project Buffers in Critical Chain Project Management," *ICMIT* (2000), pp. 162–167.
Simon, H.A., "Rational Choice and the Structure of the Environment," *Psychological Review, 63* (1956), pp. 129–138.
Steyn, H., "An Investigation into the Fundamentals of Critical Chain Project Scheduling," *International Journal of Project Management, 19* (2000), pp. 363–369.
Steyn, H., "Project Management Applications of the Theory of Constraints Beyond Critical Chain Scheduling," *International Journal of Project Management, 20* (2002), pp. 75–80.
Taylor, B.W., and L.J. Moore, "R&D Project Planning with Q-GERT Network Modeling and Simulation," *Management Science, 26*:1 (1980), pp. 44–59.
Umble, M., and E. Umble, "Manage Your Projects for Success: An Application of the Theory of Constraints," *Production and Inventory Management Journal, 41*:2 (2000), pp. 27–32.
Vanhouke, M., E. Demeulemeester, and W. Herroelen, "Maximizing the Present Value of a Project with Linearly Time-Dependent Cash Flows," *International Journal of Production Research, 39*:14 (2001), pp. 3159–3181.
Wiest, Jerome D., *A Management Guide to PERT/CPM*, Englewood Cliffs (1969).
Womack, James P., Daniel T. Jones, and Daniel Roos, *The Machine that Changed the World*, HarperCollins (1991).

Acknowledgment
This work was supported by a grant from NASA's Center for Project Management Research (CPMR).