Mario Vanhoucke

# Integrated Project Management Sourcebook

## A Technical Guide to Project Scheduling, Risk and Control

Springer

Integrated Project Management Sourcebook

Mario Vanhoucke

# Integrated Project Management Sourcebook

## A Technical Guide to Project Scheduling, Risk and Control

Springer

Mario Vanhoucke
Fac. Economics & Business Administration
Ghent University
Gent, Belgium

ISBN 978-3-319-27372-3 ISBN 978-3-319-27373-0 (eBook)
DOI 10.1007/978-3-319-27373-0

Library of Congress Control Number: 2015960270

Springer Cham Heidelberg New York Dordrecht London


Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

*Knowledge is of two kinds. We know a subject ourselves, or we know where we can find information upon it.*

Samuel Johnson

# Preface

This book is intended to be an Integrated Project Management Sourcebook for students of any project management (PM) course focusing on the integration between baseline scheduling, schedule risk analysis, and project control, known as *Dynamic Scheduling* or *Integrated Project Management and Control*. It contains a set of +70 articles that are also available online at www.pmknowledgecenter.com. The introduction of the book contains an overview article of the Project Management Knowledge Center with references to a PM bookstore, software tools, research results, and much more material relevant to the reader. The main body of this book contains articles on *baseline scheduling*, *risk analysis*, and *project control*. Each individual article focuses on one particular topic, and links are provided to the other articles (chapters) in this book. Almost all articles are accompanied with a set of questions (unlike the articles, these questions cannot be found online), for which the answers are provided at the end of this book.

This book has been written in the sunlight of Lisbon during my 4-month stay at the city of light. While artists say that light is all important to creating a masterpiece, I just think back on it as a period where I enjoyed writing in my apartment at Beco da Boavista and on the terraces of Jardim da Praça Dom Luís I (my favorite one, I called it the red terrace), Praça do Comércio, and Portas do Sol but also on the Miradouro de Santa Catarina, the city beach of Cais do Sodré, and of course at Universidade Aberta de Lisboa. In fact, it is my stay at the city that has become the masterpiece, while the book is simply the result of hard work in complete isolation from all Belgian distractions.

It goes without saying that the writing of such a manuscript is not an individual work, but is done in collaboration with people willing to help in many ways. Thank you to friend and colleague José Coelho for the many work meetings with fruitful and enriching discussions at various places in Lisbon. Thank you to Jordy Batselier, Jeroen Burgelman, Danica D'hont, Louis-Philippe Kerkhove, Pieter Leyman, Annelies Martens, and Vincent Van Peteghem for helping me with providing a set of questions and for checking the calculations throughout the many examples given in each chapter. Thank you to Mathieu Wauters for proofreading most of the articles. Thank you Louis-Philippe Kerkhove once again for setting up

a shared online correction system for our research group and for double-checking the questions of all the articles over and over again. Thank you to Tom Van Acker for providing the IT technology to put all the articles online. Thank you to Gaëtane Beernaert for supporting me in extending this work from an online learning tool to a complete integrated manuscript.

Lisbon, Portugal Mario Vanhoucke
August 2015

# Contents

# Chapter 1
# Introduction

## Welcome to PM Knowledge Center[1]

Project baseline scheduling, risk analysis and project control are crucial steps in the life of a project. The project manager uses the project schedule to help planning, executing and controlling project activities and to track and monitor the progress of the project. A major component of a project schedule is a work breakdown structure (WBS). However, the basic critical path method (CPM) schedules, or its often more sophisticated extensions, are nothing more than the starting point for schedule management. Information about the sensitivity of the various parts of the schedule, quantified in schedule risk numbers or of a more qualitative nature, offers an extra opportunity to increase the accuracy of the schedules and might serve as an additional tool to improve project monitoring and tracking. Consequently, project scheduling and monitoring/control tools and techniques should give project managers access to real-time data including activity sensitivity, project completion percentages, actuals and forecasts on time and cost in order to gain a better understanding of the overall project performance and to be able to make faster and more effective corrective decisions. All this requires understandable project performance dashboards that visualize important key project metrics that quickly reveal information on time and cost deviations at the project level or the activity level. During monitoring and tracking, the project manager should use all this information and should set thresholds on the project level or on lower WBS levels to receive warning signals during project execution. These thresholds serve as triggers to take, when exceeded, corrective actions.

This triangular role of a project schedule is often labeled as *dynamic scheduling* (see Fig. 1.1) to highlight the need and ability of project scheduling software

[1]This section is also available as an article entitled "Dynamic scheduling: Welcome to PM Knowledge Center" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_1

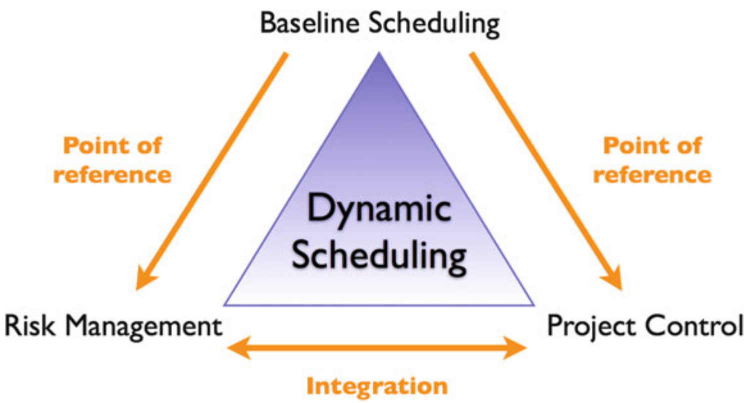



**Fig. 1.1** Dynamic scheduling: the baseline schedule, risk management and project control triangle

to dynamically create a baseline schedule environment that provides information during project execution and that can be easily adapted using the new information during project monitoring and tracking. Consequently, the three dimensions of dynamic scheduling can be summarized as follows:

- Baseline schedule construction: A project baseline schedule visualized in a Gantt chart acts as a point of reference in the project life cycle. It should especially be considered as nothing more than a predictive model that can be used for resource efficiency calculations, time and cost risk analysis, project tracking and performance measurement, and so on (see section "BS1: An Introduction to Baseline Scheduling" on page 9).
- Schedule risk analysis: When management has a certain feeling of the relative sensitivity of the various project activities on the project objective, a better management's focus and a more accurate response during project tracking should positively contribute to the overall performance of the project. Through the use of buffers inserted into the baseline schedule, the project is better protected against unexpected delays and corrective actions can be restricted to a minimum (see section "RA1: An Introduction to Risk Analysis" on page 109).
- Project control: Using dynamic information during project progress to improve corrective action decisions is the key target of project monitoring and control. The performance information obtained through EVM will be dynamically used to steer the corrective action decision making process and improve the overall success of the project (see section "PC1: An Introduction to Project Control" on page 197).

This book acts as an Integrated Project Management Sourcebook on dynamic scheduling, integrating these three dimensions in three different parts of the book. It is considered to be part of the Project Management Knowledge Center (further abbreviated as PM Knowledge Center or PMKC) that is the topic of this chapter.

The purpose of PM Knowledge Center is to act as a Project Management guide for students, lecturers and professionals interested in the field of Dynamic Scheduling. All topics described in the articles are based on research done at Ghent University (Belgium). Additionally, the aim of PMKC is to share knowledge and invoke interest in Project Management. To that purpose, a number of tools are available, that are summarized along the following lines.

- ORASTalks app: Stimulate interaction
- ProTrack: Dynamic scheduling on your desktop
- Business Game: Learning by doing
- Bookstore: Literature for students and professionals
- P2 Engine: Advancing the state-of-the-art knowledge
- Research: Project Management research

## *ORASTalks*

The main purpose of PM Knowledge Center is to interact with our audience consisting of undergraduate and graduate students, MBAs and practitioners. All summary articles of PM Knowledge Center in this book are therefore also freely available from the website www.pmknowledgecenter.com. In order to get and stay in contact with our PMKC audience, a free mobile app has been developed. ORASTalks is an app that aims at bringing students together to offer them a central place for their course content, to provide them with additional background information and to bring them in contact with interested professionals. OR-AS is an acronym for "Operations Research—Applications and Solutions" and develops software Applications and Solutions for academia and business based on a well-balanced combination between academic knowledge and practical experience. It serves as a bridge between the academic environment of our university and MBA students and the business world that they will soon (re-)discover after their graduation. The specific approach to improve and optimise business processes consists of data analysis, simulation and optimisation using state-of-the-art tools and techniques, followed by the implementation and validation. The field of Operations Research is applicable to many complex business processes. Special attention will be devoted to Integrated Project Management and Control using well-known as well as novel project management tools and techniques. More information can be found at www.or-as.be/orastalks.

## *ProTrack*

ProTrack 3.0 is a complete redesigned version of the smart version of ProTrack 2.0. Its integration with PM Knowledge Center and its strong focus on the integration of baseline scheduling, risk analysis and project control makes it yet a stronger learning tool to stimulate interaction between researchers, students and practitioners in the field of project management and dynamic scheduling. Go to www.protrack.be, buy, interact and…enjoy! More information can be found at www.protrack.be.

## *Business Game*

The Project Scheduling Game (PSG) is an IT-supported simulation game to get acquainted with dynamic project scheduling using the critical path method (CPM). The critical path method involves a time/cost trade-off in project activities and require the construction of a project baseline schedule within a predefined project deadline and budget. The uncertainty during project progress disturbs the original baseline schedule and requires interventions to bring the project back on track. More information can be found at www.protrack.be/psg.

## *Bookstore*

The themes discussed in PM Knowledge Center are the result of research projects at Ghent University and Vlerick Business School and the development of a commercial software tool ProTrack at OR-AS. Currently, three books published by Springer are available (see Fig. 1.2 or www.or-as.be/books):

- Integrated Project Management and Control: First comes the theory, then the practice: A summary book on Earned Value Management and Schedule Risk


**Fig. 1.2** PMKC bookstore: three books published by Springer


**Fig. 1.3** PMKC bookstore: the three editions of "The Art of Project Management"

Analysis, containing example projects and reports, as well as an overview of the P2 Engine tool.

- Project Management with Dynamic Scheduling: Baseline scheduling, risk analysis and project control: An overview book on the three main themes of dynamic scheduling, containing overview chapters, cases studies and a tutorial for the ProTrack software tool.
- Measuring Time: Improving project performance using Earned Value Management: A project control research study awarded by the Belgian chapter of the Project Management Institute (PMI-Belgium) and the International Project Management Association (IPMA).

A fourth book is published as a free online pdf at www.or-as.be/books/work_and_passion. The first edition is published in 2014 and ever since a yearly update has been put online, resulting in the third edition in 2015, as shown in Fig. 1.3:

- The Art of Project Management: A Story about Work and Passion: This book gives you an overview of the OR-AS endeavors done in the past and the ideas that will be done in the future. It tells about the products and ideas of OR-AS and gives you a brief overview of the most important people who inspired us and the OR-AS products. It tells about work, and the passion that has led to the results of the hard work. It's not a scientific book. It isn't a managerial book either. It's just a story . . . about work and passion

## *P2 Engine*

P2 Engine is a command line utility tool based on the LUA scripting language to generate gigabytes of project data. It generates project baseline scheduling data and risk analysis metrics as well as dynamic project progress data that can be used for testing and validating novel research ideas. P2 Engine gives the user access to the complexity of various project analysis algorithms incorporated in ProTrack 3.0. The researcher can solve difficult and critical dynamic project scheduling optimization problems using ProTrack's intelligent algorithms. It can easily produce a enormous database of optimization results for a wide range of project management problems faster than ever before and advance the state-of-the-art knowledge available today. More information on P2 Engine can be found www.p2engine.com.

## Research

All articles, books and the software tool ProTrack are the result of years of academic research. Most research done before and during the continuous development of ProTrack can be situated in the so-called Project Life Cycle (PLC, see Fig. 1.4). This cycle defines all phases between the start and end of the life of a project, and has been extensively described in various sources.

The aim of the research is threefold. First, the research goal is to search for determinants that influence the accuracy of earned value based predictive methods to forecast a project's final duration. A distinction is made between static determinants, which can be calculated before the start of the project (i.e. during the definition and scheduling phases, see study 1 of the Fig. 1.4) and dynamic determinants, which can be calculated during the project's execution and control phases (see study 2). Obviously, the ultimate goal is not the accuracy for the sake of accuracy, but rather to use this static and dynamic information to guide and improve the corrective action decision making process (see study 3). More information on the latest obtained research funding can be found at www.or-as.be or at the research site of the Operations Research and Scheduling group at www.projectmanagement.ugent.be.

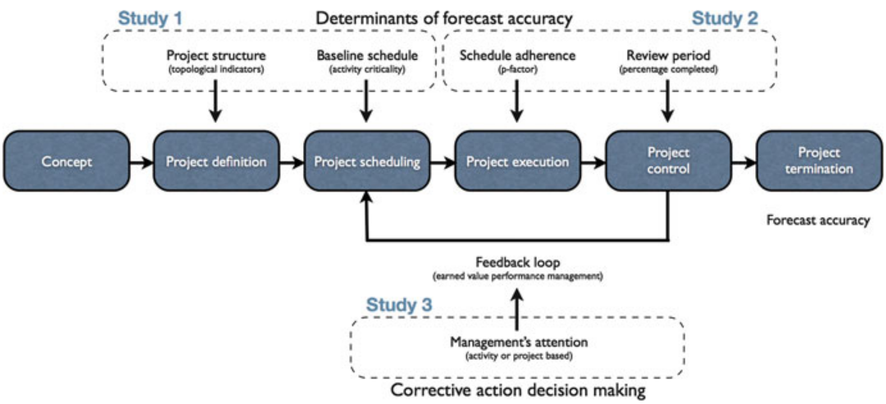



**Fig. 1.4** The project life cycle and the three integrative studies used in all research studies

# Part I
# Baseline Scheduling

# Chapter 2
# Preface

## BS1: An Introduction to Baseline Scheduling[1]

Baseline scheduling can be defined as the act of constructing a timetable to provide a start and end date for each project activity, taking activity relations, resource constraints and other project characteristics into account and aiming at reaching a certain scheduling objective.

The construction of a project baseline schedule is often a time-consuming and cumbersome task. However, the central role of the baseline schedule in a schedule risk analysis (see section "RA1: An Introduction to Risk Analysis" on page 109) and in the project control phase (see section "PC1: An Introduction to Project Control" on page 197) cannot be underestimated. It should indeed be generally accepted that the usability of a project baseline schedule is to act as a point of reference in the project life cycle, and hence, a project schedule should especially be considered as nothing more than a predictive model that can be used for resource efficiency calculations, time and cost risk analyses, project tracking and performance measurement, and so on.

The baseline scheduling topics of this book have been classified in the following categories:

- Network analysis
- Resource analysis
- Scheduling techniques

For an overview of the three dynamic scheduling dimensions, see Fig. 1.1 on page 2 or section "Welcome to PM Knowledge Center" on page 1. The different categories are briefly explained below.

[1]This section is also available as an article entitled "Dynamic scheduling: An introduction to baseline scheduling" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_2

### *Network Analysis*

Network analysis involves the construction of a project network containing activities and links between these activities to model the project network logic.

### *Resource Analysis*

The presence of resources in project scheduling increases the complexity of finding an acceptable baseline schedule. Due to this inherent complexity, software scheduling tools are necessary to construct a resource feasible schedule without resource overallocations.

### *Scheduling Techniques*

In order to have an idea about the underlying mechanism used by software tools, various scheduling techniques are discussed. These techniques are classified as follows:

- Critical path scheduling: Easy and straightforward scheduling techniques where it is assumed that no resource constraints are imposed.
- Resource scheduling: Complex scheduling techniques for projects where the use of renewable resources is restricted.
- Scheduling objectives: Information on the use of objectives that can be set during the construction of a project's baseline schedule.

# Chapter 3
# Network Analysis

## BS2: Activity Networks[1]

A project network consists of a set of nodes and arcs. A project contains activities and precedence relations to model technological relations between pairs of activities. A project network can be represented in two formats, which is the topic of this article, as follows:

- Activity-on-the-node: Activities are represented by nodes and precedence relations by arcs.
- Activity-on-the-arc: Activities are represented by arcs and the precedence relations are implicitly embedded in the network nodes.

### *Activity-on-the-Node (AoN)*

In an activity-on-the-node network format, project activities are represented by nodes and precedence relations by arcs between the nodes. Figure 3.1 displays a precedence relation between two activities in an activity-on-the-node format. It is said that activity 2 is a successor of activity 1 and activity 1 is a predecessor of activity 2.

This project network format is the default format in most commercial project management software tools such as MS Project, Primavera, ProTrack and many others. This format is easy to use in combination with different types of precedence relations (start-start, start-finish, finish-start and finish-finish relation, see article "*BS3: Precedence relations*" on page 16).

[1]This section is also available as an article entitled "Project networks: Nodes and arcs or arcs and nodes?" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_3

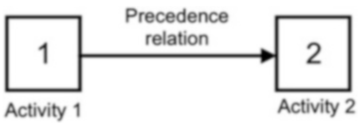



**Fig. 3.1** An example activity link in activity-on-the-node format

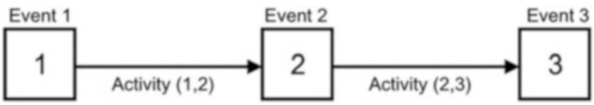



**Fig. 3.2** An example activity link in activity-on-the-arc format

**Table 3.1** A comparison between activity-on-the-node and activity-on-the-arc format

| Activity-on-the-arc | Activity-on-the-node |
|---|---|
| Originally used for PERT and CPM | Used by most commercial software tools |
| Network representation is not unique due to dummy activities | Unique network representation (no dummy arcs) |
| Restricted to finish-start with zero time-lags | Can be easily extended to SS, SF, FS and FF relations with nonzero time-lags |

## *Activity-on-the-Arc (AoA)*

In an activity-on-the-arc network format, project activities are represented by arcs, as shown in Fig. 3.2. The nodes are events (or milestones) denoting the start and/or finish of a set of activities of the project and implicitly model the precedence relations between the nodes.

It is said that activity (2,3) is a successor of activity (1,2) and activity (1,2) is a predecessor of activity (2,3).

Unlike the activity-on-the-node format, the activity-on-the-arc network representation requires some rules to follow, which can be summarized in the following lines:

- Unique representation: Each activity can be a uniquely identified by its start and end node.
- Single start/end event: Each project network starts and ends with a single event (representing the start and end of the project).
- Dummy activities: Arcs to model extra precedence relations or to fulfill the two requirements written above.

## *Comparison*

Table 3.1 displays a brief comparison between the two project network formats.

An illustrative example is given by the project data of Table 3.2 and the project networks in Fig. 3.3. While the AoA project networks are not unique due to the incorporation of one (top) or two (middle) dummy activities, the AoN network never contains dummy activities and is therefore always unique.

**Table 3.2** Project data for an example project

| Activity | Predecessors |
|---|---|
| A | None |
| B | A |
| C | A |
| D | B |
| E | B and C |
| F | D and E |

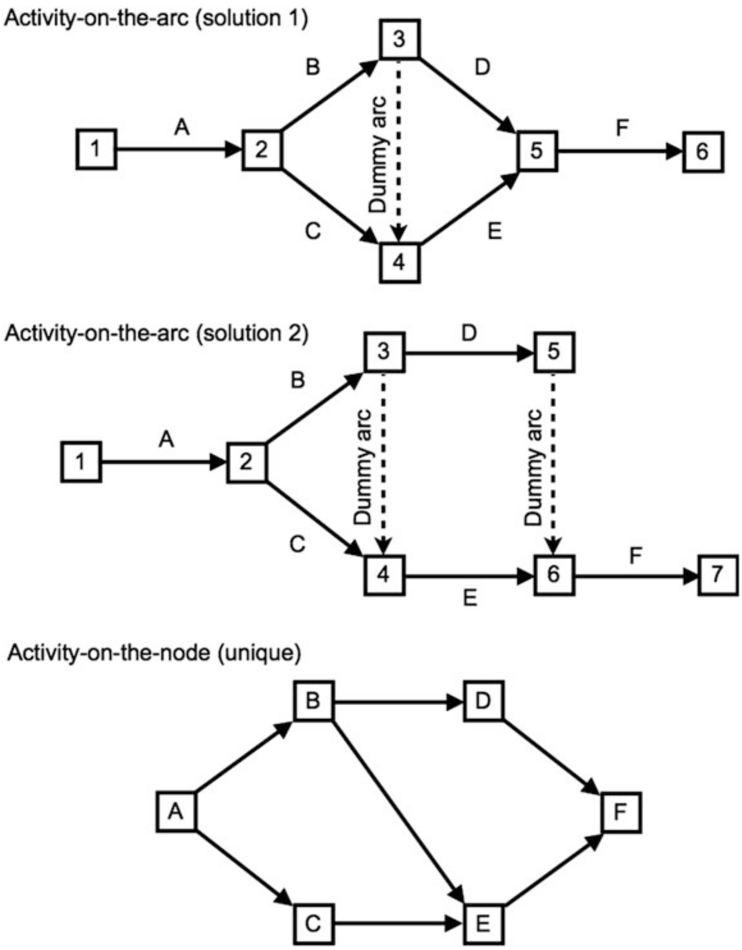



**Fig. 3.3** An AoA (not unique) and AoN (unique) network for the project data of Table 3.2

## Questions

1. The activity-on-the-arc representation is used as default format in most commercial project management software tools.
   (a) True
   (b) False
2. The activity-on-the-arc representation is restricted to finish-start relations with zero time-lags.
   (a) True
   (b) False
3. Dummy activities can be added to an activity-on-the-arc network in order to insert extra precedence relations to the network.
   (a) True
   (b) False
4. Assume the project data given in Table 3.3. The project data can be represented by one of the activity-on-the-node networks of Fig. 3.4 on page 15. Which one?
   (a) Network (a)
   (b) Network (b)
   (c) Network (c)
   (d) Network (d)
5. Based on the activity-on-the-arc project network of Fig. 3.5 on page 15, which of the following propositions is not correct? (The dashed lines indicate dummy arcs)
   (a) Activity G is a successor of activity C
   (b) Activity B can be scheduled in parallel with activity E
   (c) Activity G has no successors
   (d) Activity C can be scheduled in parallel with activity D
6. Both project networks in Fig. 3.6 on page 16 represent the same project.
   (a) True
   (b) False

**Table 3.3** Example project data

| Activity | Successors |
|---|---|
| A | B and C |
| B | D |
| C | E |
| D | F |
| E | B and F |
| F | None |

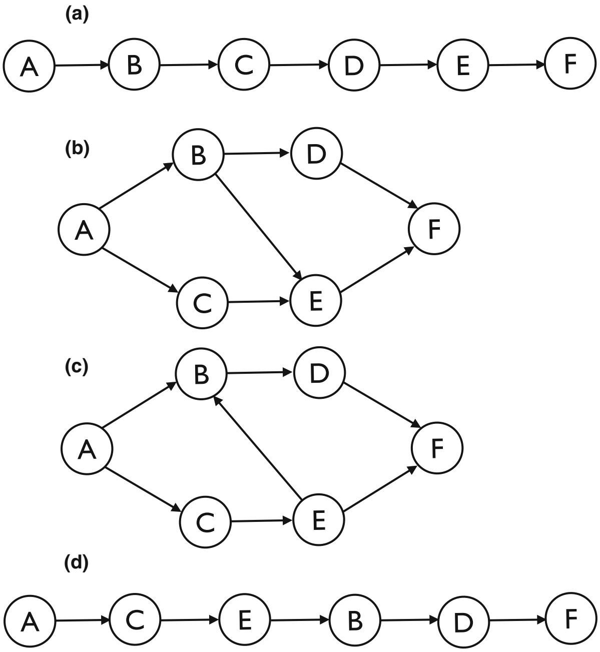



**Fig. 3.4** Four example project networks

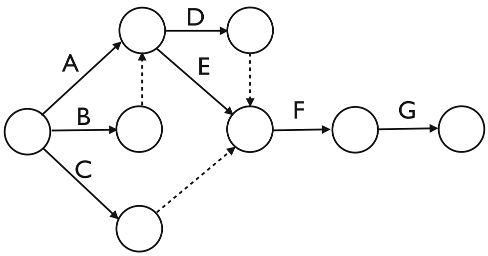



**Fig. 3.5** Example project network

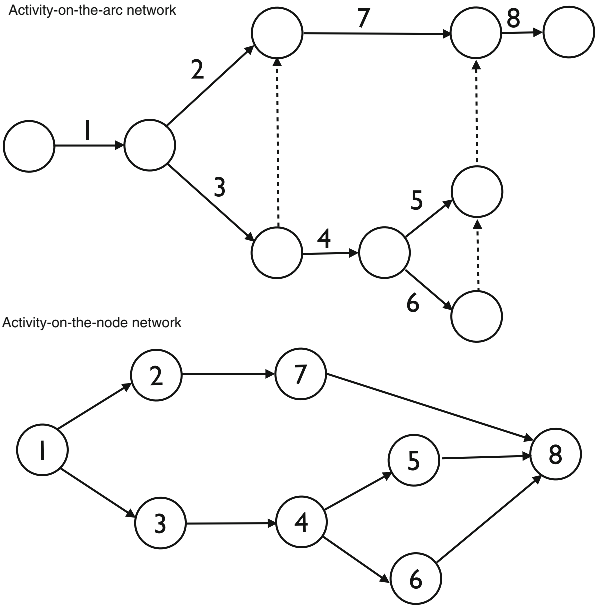



**Fig. 3.6** Two example project networks

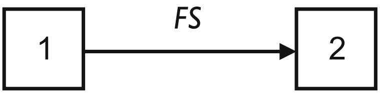



**Fig. 3.7** An example activity link between activity 1 and 2

## BS3: Precedence Relations[2]

A project network consists of a set of activities, represented as nodes in a network between which links are drawn to represent the technological precedence relations between these project activities. In Fig. 3.7, a finish-start precedence relation between activities 1 and 2 is used to imply that activity 2 cannot start earlier than the finish of activity 1.

[2]This section is also available as an article entitled "Activity links: How to add precedence relations between activities?" published online at PM Knowledge Center.

In general, three inputs are required from a project manager to define the precedence relations between activities, as given along the following lines:

- Time-lag of precedence relations: Zero or nonzero.
- Type of precedence relation: Finish-start, finish-finish, start-start and start-finish.
- Time-lag requirement of a precedence relation: Minimal or maximal.

## *Time-Lag*

Each precedence relation has a time-lag to denote a minimal time-span between the two activities. Time-lags can be positive, zero or negative, as shown in Fig. 3.8.

A finish-start relation with a zero time-lag is used to imply that activity 2 can only start after the finish of activity 1. A zero time-lag implies that the second activity can start immediately after the finish of the first activity, or later. It does not force the immediate start after the finish of the first activity, but only describes the technological requirements and limitations that exist between two activities.

Similarly, a finish-start relation with a nonzero time-lag $n \neq 0$ can be used to imply that activity 2 can only start $n$ time periods after the finish of activity 1, with $n$ denoting a positive or negative number.

## *Types of Activity Link*

While the default type of precedence relation between activities is a FS (finish-start) type, there are four different types of precedence relations between two activities, as graphically displayed in the figure below.

The four types of precedence relations of Fig. 3.9 can be summarized as follows:

- FS $= n$: Activity 2 can only start $n$ time periods after the finish of activity 1.
- SS $= n$: Activity 2 can only start $n$ time periods after the start of activity 1.
- FF $= n$: Activity 2 can only finish $n$ time periods after the finish of activity 1.
- SF $= n$: Activity 2 can only finish $n$ time periods after the start of activity 1.

with $n$ the time-lag between activities 1 and 2.

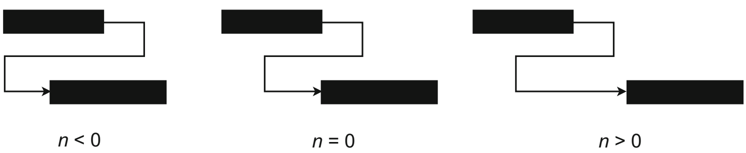



**Fig. 3.8** Negative, zero and positive time-lags between two activities

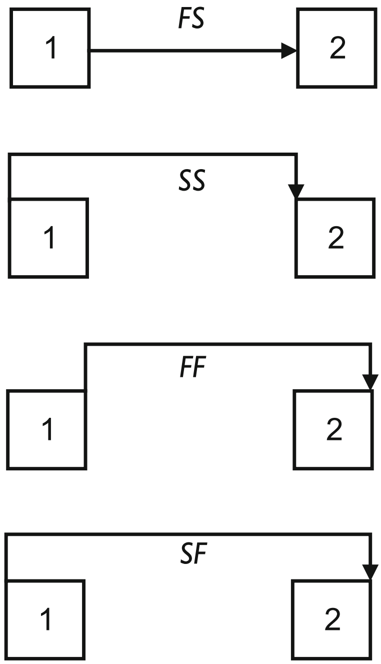



**Fig. 3.9** Four types of precedence relations between activities 1 and 2

## *Minimal/Maximal*

While it is assumed that precedence relations imply minimal requirements between two activities, they can be easily extended to maximal requirements. Although most commercial software tools do not incorporate the possibility of maximal time-lags, they are briefly discussed here.

- A finish-start relation with a minimal time-lag of $n$ can be used to imply that activity 2 can only start $n$ or more time periods after the finish of activity 1.
- A finish-start relation with a maximal time-lag of $n$ can be used to imply that activity 2 can only start $n$ or less time periods after the finish of activity 1.

Logically, the extension from a minimal to a maximal time-lag also holds for start-start, finish-finish and start-finish precedence relations. However, they often lead to an increased complexity of the project network. Minimal and maximal precedence relations can and often will be used in combination.

## Questions

1. Assume a finish-start relation with a nonnegative time-lag between activities A and B. Activity B can start before the finish of activity A.
   (a) True
   (b) False
2. A start-finish relationship with a time-lag of T periods between activities A and B means that activity B can only start T periods after the finish of activity A.
   (a) True
   (b) False
3. A finish-start relation between activities A and B with a maximal time-lag of T periods means that activity B can only start T (or less) periods after the finish of activity A.
   (a) True
   (b) False
4. Assume a start-start relationship between activities A and B, with a maximal time-lag of 3. Activity A has a duration of 4 weeks and starts at the beginning of week 18, while activity B has a duration of 5 weeks. What is the latest possible finish time of activity B?
   (a) End of week 27.
   (b) End of week 21.
   (c) End of week 26.
   (d) End of week 25.
5. Assume a finish-start relation with a maximal time-lag of 3 between activity A and B and a start-start relation with a minimal time-lag of $-3$ between activity B and A. What is the earliest possible start time of activity A (duration $= 3$) if activity B (duration $= 2$) starts at time 15.
   (a) Week 9
   (b) Week 12
   (c) Week 15
   (d) Week 11
6. A finish-start relation with a minimal time-lag of 4 is the same as a finish-start relation with a maximal time-lag of $-4$.
   (a) True
   (b) False

7. An activity A with a duration of 3 days has a start-start relation (minimal time-lag of 3 days) with activity B (duration of 3 days). Activity B has a finish-finish relation with A with a minimal time-lag of −4 days. For a given start of activity B, activity A has 3 possible start dates.

   (a) True
   (b) False

## BS4: Minimal and Maximal Time-Lags[3]

A project network consists of a set of activities, represented as nodes in a network between which links are drawn to represent the technological precedence relations between these project activities (see article "*BS3: Precedence relations*" on page 16). These precedence relations can be one out of four types (start-start (SS), start-finish (SF), finish-start (FS) and finish-finish (FF)), with a positive, zero or negative time-lag to express a minimal or a maximal time window between two activities.

In Fig. 3.10, a finish-start precedence relation between activities 1 and 2 with a minimal time-lag of 3 days is used to imply that activity 2 cannot start earlier than 3 days after the finish of activity 1.

In this article, it will be shown that each of the four types of precedence relations can be transformed to any other type, and that the equivalence between minimal and maximal time-lags can be easily shown by adding reverse arcs in a project network, as given along the following lines:

- SS, SF and FF to FS transformations
- Maximal to minimal transformations

### *Minimal Time-Lag Transformations*

Each precedence relation has a time-lag to denote a minimal time-span between the two activities. Time-lags can be positive, zero or negative, and can be used in combination with each of the four precedence relation types. Figure 3.11 shows

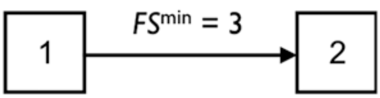



**Fig. 3.10** An example activity link between activity 1 and 2

[3]This section is also available as an article entitled "Activity links: The equivalence of minimal and maximal time-lags" published online at PM Knowledge Center.

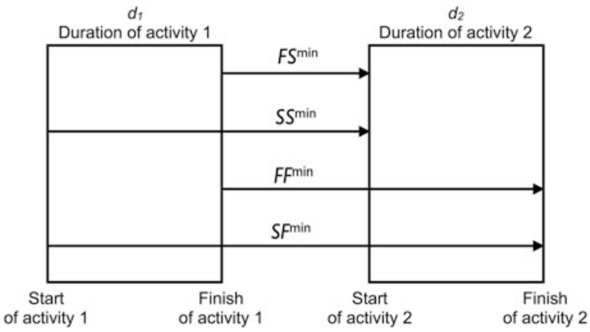



**Fig. 3.11** Four different types of minimal precedence relations

the four types of relations between activities 1 and 2, where the width of the bar represents the duration of both activities.

Since a finish-start (FS) relation can be considered as the default precedence relation, Fig. 3.11 will be used to illustrate the transformation of a SS, SF and FF relation to a FS relation, using the following simple formulas:

- $\text{SS} = \text{FS} + d_1$
- $\text{FS} = \text{FS}$
- $\text{SF} = \text{FS} + d_1 + d_2$
- $\text{FF} = \text{FS} + d_2$

with $d_x$ the duration of activity $x$.

As an example, a finish-start relation between two activities, with durations equal to 5 and 7 for activities 1 and 2, respectively, and a minimal time-lag of 4 days is identical to a start-start relation with a time-lag of 9 days, a start-finish relation with a time-lag of 16 days and a finish-finish relation with a time-lag of 11 days.

## *Minimal and Maximal Time-Lag Equivalence*

The minimal time-lag transformations discussed previously can be extended to maximal time-lags, where similar equations can be used. Moreover, the equivalence between minimal and maximal time-lags can be easily obtained by reversing project network arcs. Indeed, a maximal time-lag can be represented by a negative minimal time-lag with an arc between the two activities in the opposite direction. Consequently, project networks with generalized precedence relations can be represented by cyclic networks, which often lead to an increased complexity or are sometimes completely unacceptable in project management software tools.

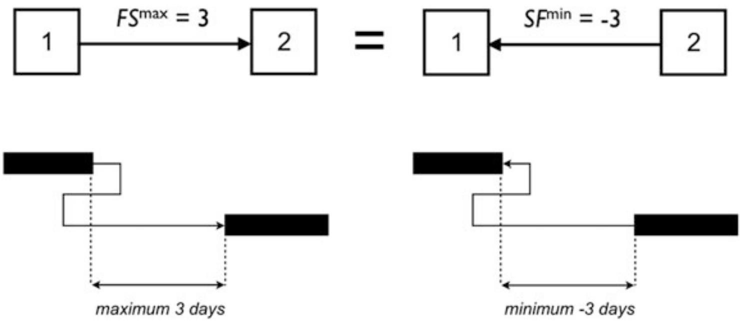



**Fig. 3.12** The equivalence of minimal and maximal time lags

Figure 3.12 shows such a transformation from a maximal finish-start relation ($FS^{max} = 3$) between activities 1 and 2 to a minimal start-finish relation ($SF^{min} = 3$) between activities 2 and 1. Activity 2 is allowed to start maximum 3 time periods after the finish of activity 1, which is exactly the same as specifying that activity 1 can only finish minimum $-3$ time periods after the start of activity 2.

Obviously, this equivalence also holds for the other three types (SS, SF and FF).

## Questions

1. In Fig. 3.13, there is a start-start relation between activities 1 and 2 with a maximal time-lag of 2 days. This means that activity 2 can start at least 2 days after the start of activity 1.

   (a) True
   (b) False

2. If there is a minimal start-finish relation of 10 days between activities 1 and 2, the relation between both activities can also be written as a finish-start relation of $10 - d_1 - d_2$ days, with $d_1$ and $d_2$ the duration of activities 1 and 2, respectively.

   (a) True
   (b) False

3. A start-start relation between two activities with a minimal time lag of 2 can also be written as $FF^{min} = -2$.

   (a) True
   (b) False

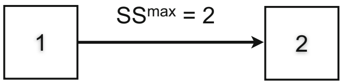



**Fig. 3.13** Example with two activities (SS)

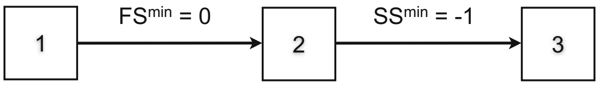



**Fig. 3.14** Example with three activities (FS and SS)

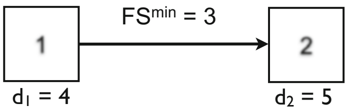



**Fig. 3.15** Example with two activities (FS)

4. Assume a project network with two activities (activity A with a duration of 4 and activity B with a duration of 2), between which a start-start relation with a minimal time-lag of 3 days exists. What is the correct transformation of this relation to a FS relation?
   (a) $FS^{min} = 3$
   (b) $FS^{max} = -1$
   (c) $FS^{min} = -3$
   (d) $FS^{min} = -1$
5. Based on the precedence relations between each of the three activities as shown in Fig. 3.14, what is the earliest start time of activity 3, assuming that activity 1 starts at time 0. Activity 1 has a duration of 5, activity 2 a duration of 3 and activity 3 a duration of 2.
   (a) 9
   (b) 7
   (c) 8
   (d) 4
6. In Fig. 3.15, a precedence relation is given between activities 1 and 2. Which of the following transformations is correct? (multiple answers possible)
   (a) $FF^{min} = 8$
   (b) $SF^{min} = 9$
   (c) $SS^{max} = 7$
   (d) $SS^{min} = 7$

## BS5: Activity Constraints[4]

Activity constraints can be imposed when there is a need to control the start or finish of an activity in a project schedule. In this article, three types of commonly used activity constraints will be discussed along the following dimensions:

- Constraint types
- Constraint hardness
- Switching from constraint hardness

### *Constraint Types*

Activity constraints can be used to control the degrees of freedom in a project schedule and can be generally classified in three classes. Each type has a start and a finish version, as follows:

- Ready dates imply earliest start or finish times on activities, and hence, force the activity to start/finish no earlier than the defined time instance. These constraints are known as ready start time (RST) or ready finish time (RFT).
- Due dates imply latest start/finish times on activities and force activities to start/finish no later than a predefined time instance. These constraints are referred to as due start time (DST) or due finish time (DFT).
- Locked dates imply a fixed time instance and force the activity to start/finish on a predefined time instance, known as locked start time (LST) or locked finish time (LFT).

Figure 3.16 displays a project with three activities and with ready times and a due date. This picture will be used to illustrate the effect of shifting activity 1 further in time. This effect will depend on the hardness of the three constraints, as explained below.

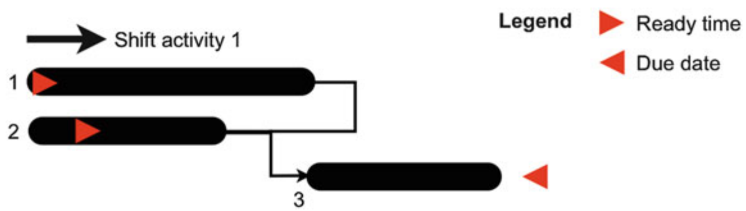



**Fig. 3.16** An example project with activity constraints and a manual forward shift

[4]This section is also available as an article entitled "Activity constraints: Limiting the degrees of freedom of project activities" published online at PM Knowledge Center.

### *Constraint Hardness*

The use of activity constraints increases the project manager's control of the project schedule but leads to a flexibility decrease for the project scheduling algorithm of the software tool. Although it can be generally recommended to restrict the use of activity constraints to prevent the construction of a rigid project schedule, there are four ways to handle constraints in a project schedule, varying the degree of constraint hardness. These hardness options influence the result of user interventions (e.g. a manual activity shift in time, adding a constraint or precedence relation, changing an activity duration, etc.) or software interventions (e.g. rescheduling the baseline schedule, update of tracking information, etc.) on the project schedule. The four constraint hardness modes are as follows:

- Hard constraint mode: All activity constraints and the precedence relations between activities need to be satisfied at all times. When manual activity shifts lead to constraint and/or precedence violations, the software tool will automatically return to the previous schedule and undo the infeasible user intervention.
- Moderate constraint mode: All activity constraints need to be satisfied at all times. When user interventions lead to constraint violations, the precedence relations will be overruled by allowing a certain degree of overlap between project activities. Consequently, a moderate activity constraint has a higher priority than a precedence relation between two activities.
- Soft constraint mode: Activity constraints can be violated due to user interventions at any time. However, the software tool will try to prevent the total number of constraint violations by searching for the best possible schedule to satisfy constraints to the best possible extent.
- Forward constraint mode: Activity constraints are only satisfied in one direction and are treated as forward activity constraints. Consequently, all activity ready times are explicitly taken into account, while locked times and due dates are often ignored: locked times are treated as ready times, which will only be satisfied unless it is not possible due to predecessor activities while due dates are completely ignored and will possibly be violated by user or software interventions at any time.

Figure 3.17 shows that the impact of a shift in activity 1 of Fig. 3.16 is different for each of the four types of the hardness of the activity constraints. Note that the activity shift for both the moderate and the soft constraint modes result in a violation of the original project schedule logic. More precisely, while the moderate constraint mode resolves conflicts between activity constraints by allowing overlaps between activities (i.e. violation of the original precedence relations), the soft mode tries to construct a project schedule where the constraint violations are minimized. In the example of Fig. 3.17, the soft mode constructs a baseline schedule without activity overlaps but leads to a violation of the due date constraint of the third activity.

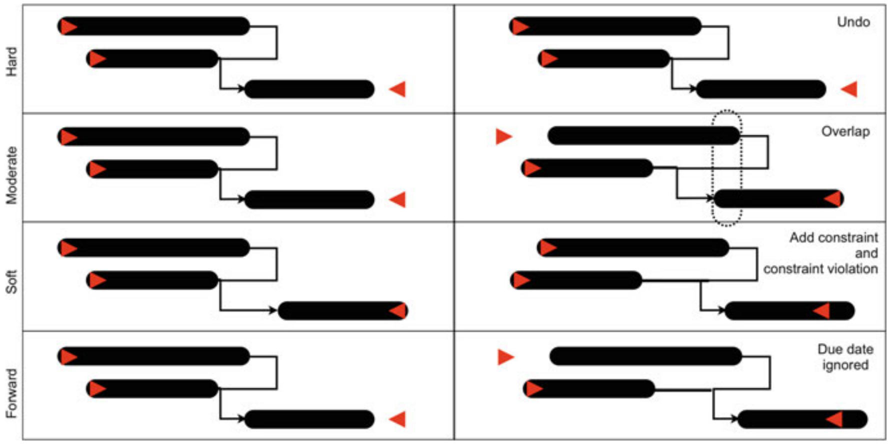



**Fig. 3.17** The impact of a shift in activity 1 of Fig. 3.16 depends on the hardness of the activity constraints

## *Switching from Constraint Hardness*

Changing the constraint hardness for a current schedule might lead to unexpected activity shifts, complete schedule changes or infeasible schedule solutions and therefore need to be done with care. Since the hard constraint mode is the most strict constraint hardness, switching to this mode can lead to three possibilities:

- The switch can be done without any changes and the constraint hardness is set to the hard mode.
- The switch is not possible with the start times of the current schedule due to constraint conflicts, but leads to no constraint conflicts when every activity is set to its earliest possible start (known as an Earliest Start Schedule (ESS)). In this case, the user has the choice to either undo the constraint hardness mode switch or to switch to an ESS.
- The switch is neither possible with the start times of the current schedule nor with the ESS: Undo the constraint mode switch.

### Questions

1. The degrees of freedom in a project can be limited by imposing additional constraints on the activity start times only, and not on the activity finish times.

   (a) True
   (b) False

2. Three main types of activity constraints can be distinguished.
   (a) True
   (b) False
3. The constraint hardness implies a trade-off between the project manager's control and the flexibility of the scheduling algorithm.
   (a) True
   (b) False
4. Changing the constraint hardness of a schedule always leads to unexpected activity shifts.
   (a) True
   (b) False
5. The constraint "an activity can start or finish no later than the predefined time instance" belongs to the following type:
   (a) Locked dates
   (b) Due dates
   (c) Ready dates
6. Rank the three types of constraint hardness in the correct order from the least to most restrictive:
   (a) Hard, soft, moderate constraint modes
   (b) Soft, moderate, hard constraint modes
   (c) Hard, moderate, forward constraint modes

# Chapter 4
# Resource Analysis

## BS6: Resource Types[1]

In the PERT/CPM scheduling techniques (see articles "*BS15: PERT*" on page 59 and "*BS11: CPM*" on page 46), project activities are characterized by their estimated duration(s) and project networks are constructed by adding precedence relations between these activities. It is therefore implicitly assumed that these activities do not require resources during their execution (or alternatively, the assumption is that the resources are unlimited in availability). In practice, activities need resources during their execution that are often limited in availability. These resources have been classified in two basic categories, as follows:

- Renewable resources
- Consumable resources

After the presentation of the two commonly used resource types, some special cases are briefly presented.

### *Renewable Resources*

Renewable resources are available on a period-by-period basis, i.e. the available amount is renewed from period to period (e.g. per hour, per day, ...). Only the total resource use at every time instant is constrained. Typical examples are manpower, machines, tools, equipment and space. As an example, skilled laborers are available to work each day on a project, although their availability is limited each day and

[1]This section is also available as an article entitled "Resource types: Renewable and consumable resources" published online at PM Knowledge Center.

 
M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_4

may vary over time due to absence, sickness, vacations, etc. Consequently, there is no general constraint placed on the number of days skilled labor may be used but instead they are renewed each period of the project.

The use of renewable resources with limited periodic availability constitutes the heart of most project scheduling tools, and are the subject of various scheduling algorithms to schedule so-called resource constrained project scheduling problems (see article "*BS22: What is my scheduling objective?*" on page 88).

## *Consumable Resources*

Consumable resources (or nonrenewable) are not constrained on a periodic basis but rather have a limited consumption availability for the entire project. Typical examples are money, raw materials and energy. Usually, the overall project costs are limited and pre-defined in a total contract price. Their consumption is not renewed as is the case with renewable resources, but instead, these resources are consumed when used.

## *Special Cases*

Renewable and consumable resources are the two commonly used resource classes in software tools, although many other extensions have been proposed in the academic literature. These special cases can often not be incorporated in software tools, although they can sometimes be modeled as a combination of renewable and consumable resources. A nonexhaustive list is given along the following lines.

Spatial resources are required by a group of activities, rather than a single activity as renewable resources. The spatial resource is occupied from the first moment an activity from the group starts until the finish of all activities from that group. Examples are dry docks in a ship yard or a freezing machine in the Westerscheldetunnel (see www.westerscheldetunnel.nl).

Doubly constrained resources are constrained on a periodic basis, similar to renewable resources, as well as for the total project duration, as with the consumable resources. An example is a total budget with an extra restriction of a maximum limit per period.

Partially renewable resources assume for each resource a capacity restriction on a subset of periods. A resource type is characterized by a number of subsets of periods and a certain total availability. A typical example is a worker who is allowed to work during all days of the week and only one day (i.e. Saturday or Sunday) during the weekend.

**Questions**

1. Consumable resources have a limited consumption availability for the entire project.
   (a) True
   (b) False
2. A total budget with an extra restriction of a maximum limit per period is an example of a partially renewable resource.
   (a) True
   (b) False
3. The availability of renewable resources decreases on a period-by-period basis depending on their consumption in each period.
   (a) True
   (b) False
4. Which of the following statements is not applicable to spatial resources?
   (a) Spatial resources are required by a group of activities.
   (b) The capacity of spatial resources is not restricted for all periods.
   (c) Spatial resources are occupied from the first moment an activity from the group starts until the finish of all activities from that group.
   (d) An example of a spatial resource is the dry dock in a ship yard.
5. Which of the following examples is not a nonrenewable resource?
   (a) Money
   (b) Coal
   (c) Oil
   (d) People

## BS7: Critical Path/Chain[2]

It is a wise and generally accepted management principle to put a focus on the constraining or limiting factor of a system that determines the system's goal. In project management and scheduling, the scheduling objective is the objective during the construction of a project baseline schedule.

In this article, it is assumed that the scheduling objective is the minimization of the total project duration. A distinction is made between the construction of a

[2]This section is also available as an article entitled "The critical path or the critical chain? The difference caused by resources" published online at PM Knowledge Center.

schedule with and without renewable resource constraints, as summarized along the following lines:

- The critical path: No renewable resources limit the scheduling degrees of freedom.
- The critical chain: Renewable resources limit the scheduling degrees of freedom.

## *Critical Path*

Figure 4.1 displays a project network with 11 activities and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity. In the PERT/CPM scheduling approach, projects are not subject to limited renewable resources, and hence, the construction of a baseline schedule boils down to sequencing all activities according to their precedence relations. The minimal duration of the project is determined by the length of the critical path, which is a sequence of activities in the project network, as shown in Fig. 4.1 in bold red or grey. The attentive reader has noticed that a second path (1-2-5-8-10-11) is also equally critical since it has the same total duration of 22 time periods. Indeed, in a network, multiple critical paths can occur.

## *Critical Chain*

Scheduling projects within the presence of resources with a limited availability might lead to resource conflicts. In Fig. 4.2, it is assumed that the availability of the resource is restricted to six units for all periods of the project. The earliest start schedule displayed at the top of the figure shows a resource conflict between period 8 and 13 since the total use of the renewable resource by activities 3, 4, 5, 6 and 7

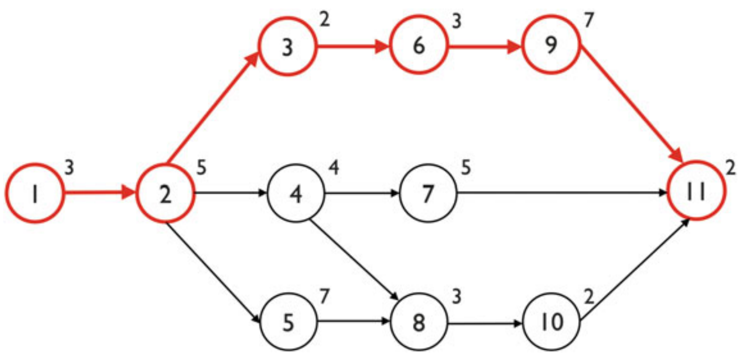



**Fig. 4.1** An example project network with the critical path equal to 22 time periods

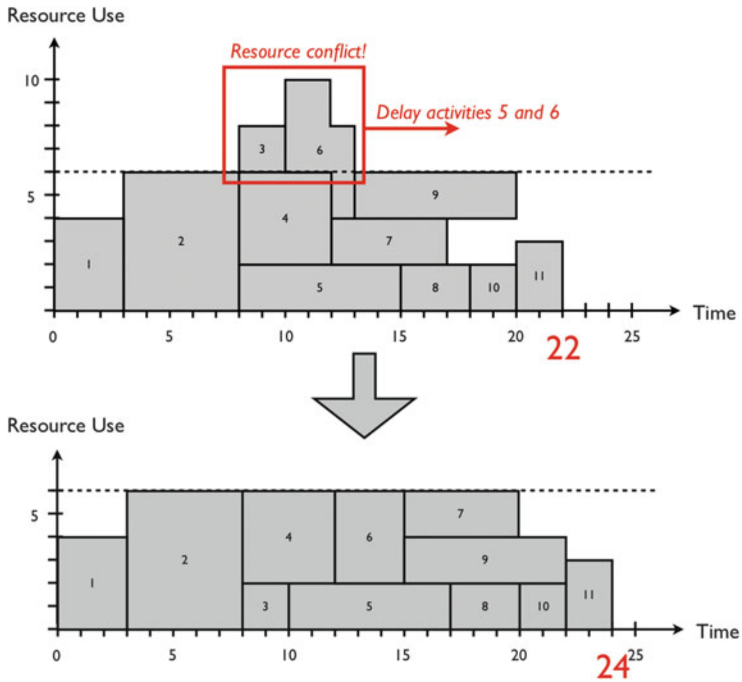



**Fig. 4.2** The resource profiles for the critical path schedule and a resource feasible schedule

exceeds its limited availability of 6. In order to solve this resource conflict, activities need to be shifted further in time to time periods where resources are still available (see article "*BS8: Linking resources*" on page 36). The way these resource conflicts have to be solved depends on the scheduling objective (see article "*BS22: What is my scheduling objective?*" on page 88). The bottom schedule of Fig. 4.2 shows a resource feasible schedule with a minimal project duration of 24 time periods as a result of shifts in activities 5 and 6.

A comparison between the resource unconstrained critical path schedule (length = 22) and the resource feasible and resource constrained schedule (length = 24) leads to the following two conclusions:

- The critical path duration is always smaller than or equal to the resource feasible project schedule duration. Indeed, since the limited availability of renewable resources puts an extra constraint on the degrees of freedom during the construction of a baseline schedule, the critical path length is a lower bound on the length of the resource feasible schedule duration.
- In analogy to the critical path, which determines the subset of project activities that are responsible for the length of the project schedule, a similar concept, known as the critical chain, holds for a resource feasible schedule. However, while the critical path is the result of a simple sequencing approach, the critical

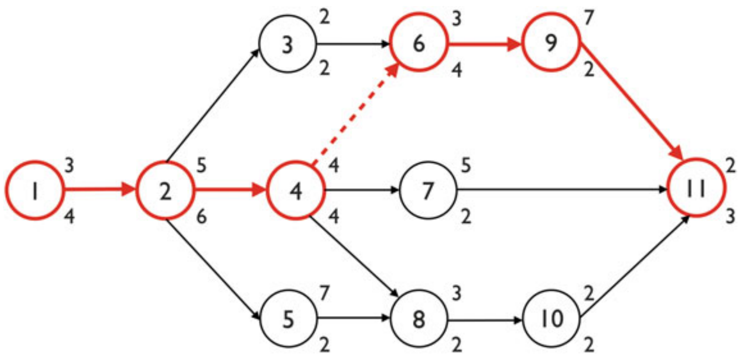



**Fig. 4.3** An example project network with the critical chain equal to 24 time periods due to the incorporation of a resource arc between 4 and 6

chain depends on the scheduling objective, and hence, on the choice of activities that have been delayed. In the example, activities 5 and 6 have been delayed to solve the resource conflict, but another scheduling objective could result in other activities to delay and hence, another critical chain. The critical chain in the example is equal to the activity sequence 1-2-4-6-9-11 and consists of precedence relations as well as, unlike the critical path, resource relations. Figure 4.3 shows the resource link between activities 4 and 6 that has been added to the project network (the number below the node is used to refer to the resource demand). The extra resource link is added to prevent that activities 4 and 6 will be scheduled simultaneously.

## Questions

1. The resource feasible project schedule duration can be smaller than the critical path duration.
   (a) True
   (b) False
2. The critical chain consists not only precedence relations, but also resource relations.
   (a) True
   (b) False
3. The critical path length of the project network in Fig. 4.4 is 11.
   (a) True
   (b) False

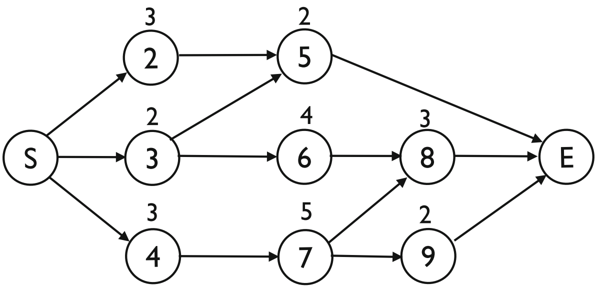



**Fig. 4.4** An example project network

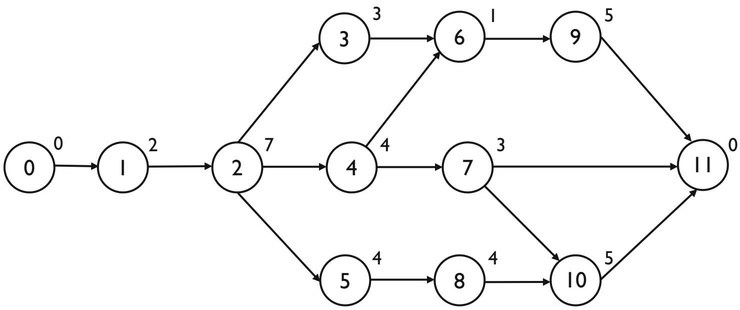



**Fig. 4.5** An example project network

4. How long is the critical path of the project network in Fig. 4.5? (The number above each node represents the activity duration.)
   (a) 16
   (b) 18
   (c) 20
   (d) 22
5. What is the critical chain of the schedule in Fig. 4.6? (Each number above the node represents the activity duration and each number below the node represents the renewable resource demand. The availability of the renewable resource is equal to 5.)
   (a) 0-1-2-4-7-6-8-10-11
   (b) 0-1-2-5-3-6-9-11
   (c) 0-1-2-5-7-6-8-9-11
   (d) 0-1-2-4-7-10-11

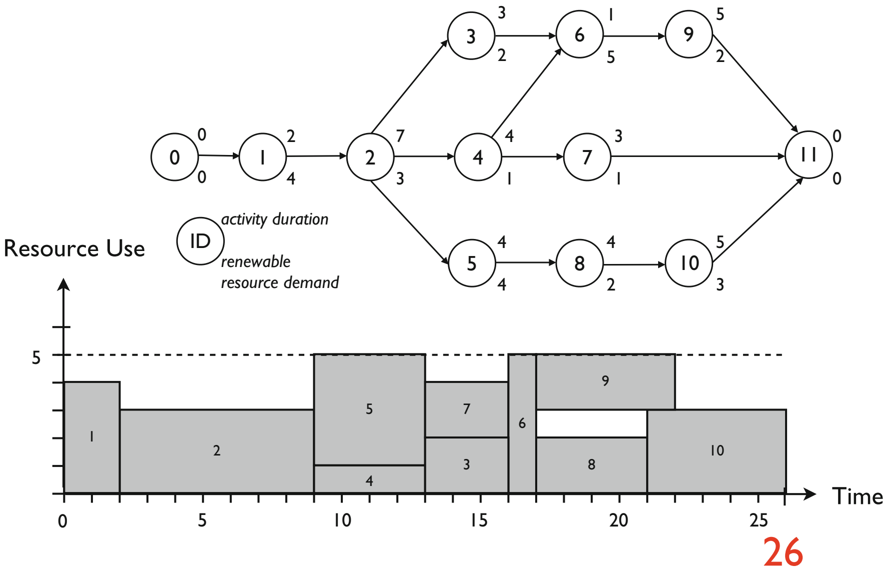



**Fig. 4.6** An example project network and resource feasible schedule

## BS8: Linking Resources[3]

Activities require renewable resources during their execution, which are limited on a periodic basis (see article "*BS6: Resource types*" on page 29). In practice, activities need resources during their execution that are often limited in availability. In this article, the connection between the resource requirements of activities and the limited availability of these resources is discussed along the following topics:

- Resource availability: Renewable resources.
- Resource demand: Activity resource requirements.
- Resource conflicts.

Figure 4.7 displays a project network with 11 activities and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand.

[3]This section is also available as an article entitled "Linking resources to activities: Resource availability and resource demand" published online at PM Knowledge Center.

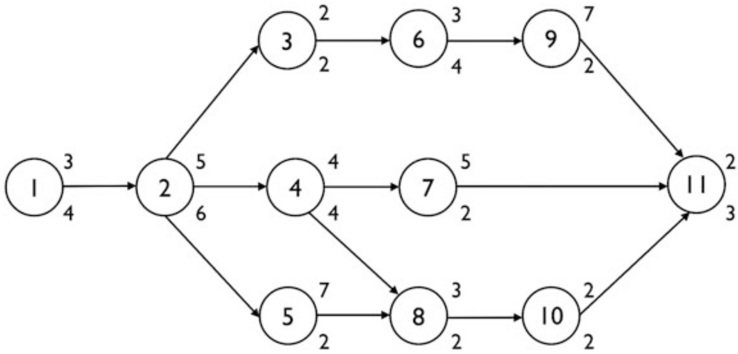



Fig. 4.7 Project network and activity data

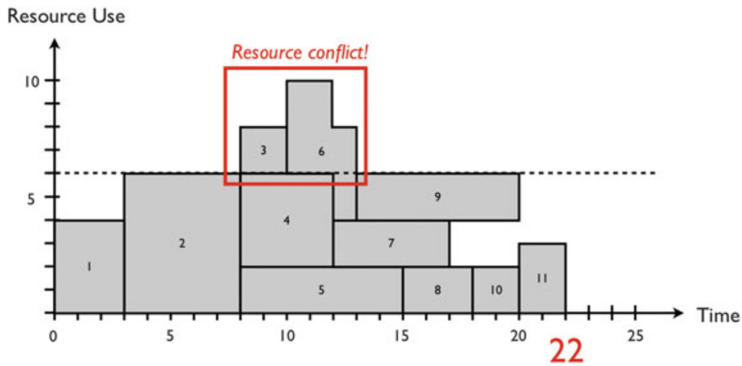



Fig. 4.8 Renewable resource availability and resource demand

## *Resource Availability*

Renewable resources have a limited availability for each period, which can be fixed over the complete scheduling horizon but also variable with upward or downward jumps (e.g. people are unavailable at some days). In Fig. 4.8, the fixed periodic availability of one renewable resource is given by the straight dotted line to show that maximum six resources (e.g. people) are available all days of the project.

## *Resource Demand*

Activities require renewable resources in order to be executed, such as a team of skilled workers, machines or tools. Allocating resources to activities links these resources and their corresponding costs to individual activities and creates value for the activity's resource demand. In Fig. 4.8, each activity is represented by a rectangle with the length indicating the activity duration and the height denoting the renewable resource requirement/demand. The work content is defined as the product of the two, as follows:

$$\text{Work} = \text{Duration} * \text{Demand}$$

Each activity is scheduled as soon as possible (i.e. an earliest start schedule, see article "*BS10: Activity slack*" on page 43) while respecting the precedence relations of the project network of Fig. 4.7.

## *Resource Conflict*

When the demand for a resource is higher than its availability (since the resource is allocated to various activities scheduled at the same time), it is said that this resource is over-allocated and the schedule has a resource conflict. Figure 4.8 shows a resource conflict between period 8 and 13 due to over-allocations of resources caused by activities 3, 4, 5, 6 and 7.

**Questions**

1. If the availability of a certain resource type is fixed, it is assumed that the resources are available over the complete scheduling horizon.

   (a) True
   (b) False

2. If a resource is over-allocated, this means that there is a resource conflict in the schedule.

   (a) True
   (b) False

3. If none of the resource types is limited in availability, all activities can be scheduled at their earliest start time.

   (a) True
   (b) False

## BS9: Activity Costs[4]

The calculation of costs is an important aspect of project scheduling and control. In this section, two main types of costs are discussed, each consisting of two subtypes. The total activity cost consists of the sum of all subtype costs, as will be explained in this article. The two main types are as follows:

- Activity cost: A fixed or variable cost that is set for an activity and not for a resource.
- Resource cost: A fixed or variable cost for the use of resources when demanded by the various project activities.

### *Activity Cost*

A fixed cost can be assigned to an activity without having any link with its resource assignments. It is a fixed amount of money, independent of the duration of the activity and the work content for one or more resources assigned to this activity.

A variable cost per time unit can be assigned to an activity regardless of its use of renewable and consumable resources. It is a variable amount of money that is dependent on the activity duration and can be used to describe resource independent costs such as activity overhead.

### *Resource Cost*

The cost per use is the cost for the use of a resource that can be considered as a one-time cost incurred every time the resource is used by the activity. The per use cost for a consumable resource is applied only once from the moment the resource is used. The per use cost for a renewable resource, however, depends on the resource demand of the activity (i.e. its resource requirement and not its total work content). As an example, the cost per use for a consumable resource like gallons of gasoline needed to feed excavators necessary during a construction project is € 250 per delivery, regardless of the amount of gallons brought per delivery per truck (obviously, this is cost of delivery and not the cost for the gasoline itself, which will be measured by the cost per unit discussed hereafter). The excavator itself, on the contrary, is a renewable resource and has a per use cost of € 1,000. Working with three excavators in parallel to finish the activity makes the total per use cost equal to € 3,000.

[4]This section is also available as an article entitled "The cost of a project activity: Calculating the activity and/or resource costs" published online at PM Knowledge Center.

Unlike the one-time cost per use, the cost per unit is a cost that typically depends on the amount of the resource demanded by the activity and its duration. The calculation of the total activity per unit cost differs according to the resource type, renewable or consumable, as explained along the following lines:

- Renewable resources: The costs per unit are cost rates calculated per time unit (hours, days, weeks, etc.) and per resource unit, and hence, are based on the total work content (= activity duration * resource demand) of the activity.
- Consumable resources: Normally, the use of consumable resources by project activities is expressed in units that are typically different from time units. Consequently, the costs per unit are monetary rates typically not calculated per hour but expressed in other units (per weight, per length, per pallet, etc.). However, there are examples where consumable resource use is expressed as a time dimension. To that purpose, the assignment of a consumable resource to an activity can be done in two alternative ways:

  – Fixed use: The unit for the cost/unit calculation of a consumable resource is anything but a time dimension. It is an indication that the quantity of the resource used by an activity is independent from its duration. A typical example is the cost per unit for materials, such as bricks needed to build a wall, which is equal to € 100 per pallet, regardless how much time it takes to build that wall. The gasoline example for the excavator could also be considered as a consumable resource with fixed use, since its cost completely depends on the amount used, expressed in gallons, to finish a certain activity.
  – Variable use: The unit for the calculation of the cost/unit for a consumable resource can be expressed in a time dimension (hours, days, …). It is an indication that the quantity of resources used by an activity changes proportionally as its duration changes. In the gasoline example, the resource use could have been specified as the number of gallons needed per day. In this case, it is implicitly assumed that the daily occupation of the excavator is known and more or less stable, and the consumable resource of gasoline then depends on the daily gasoline consumption of the excavator, and hence, on the number of days it takes to finish that activity with the help of that excavator.

The distinction between fixed and variable use is often a matter of choice made by the project scheduler to express the consumable resource consumption needed to finish that activity. For example, if the assignment units value for a consumable resource is, e.g. 100 tons for an activity with a duration of 10 days, the total resource consumption is 100 tons. However, if the assignment units value for that resource is 100 tons/day (variable), the total resource consumption is 1,000 tons (fixed).

The resource cost options are displayed in Table 4.1.

**Table 4.1** Resource cost options (Source: ProTrack 3.0)

| | Renewable | Consumable |
|---|---|---|
| | Limited availability during scheduling | No limited availability during scheduling |
| Cost/use | Cost/use * resource demand | Cost/use |
| Cost/unit | Cost/unit * activity duration * resource demand | Variable: cost/unit * activity duration * resource demand<br>Fixed: cost/unit * resource demand |

**Table 4.2** Summary table for the "total activity cost" calculations

| |
|---|
| Total activity cost |
| = |
| fixed activity cost |
| + |
| variable activity cost * duration |
| + |
| cost/use * resource demand (renewable resources) |
| + |
| cost/unit * work content (renewable resources) |
| + |
| cost/use (consumable resources) |
| + |
| cost/unit * resource demand * duration |
| (variable consumable resource with units expressed in time units (h)) |
| + |
| cost/unit * #units |
| (fixed consumable resource with units not expressed in time units) |

## *Total Cost*

The total activity cost is equal to the sum of these cost factors as shown in the formula of Table 4.2.

### Questions

1. The variable cost of an activity never depends on the duration of that activity.
   (a) True
   (b) False

2. The total renewable resource cost for an activity is based on the total work content of the activity.
   (a) True
   (b) False
3. A fixed activity cost is independent of the duration of the activity or the number of resources it uses.
   (a) True
   (b) False
4. Mr. Surkamp is the project manager of a construction project. The following information about the project is known:
   - Administration cost for the project is € 5,000.
   - A crane is hired for 2 months from the beginning of month 2 till the end of month 3 (€ 2,000 per month). The crane is only used in the first week of month 2 and the last 2 weeks of month 3 (4 weeks in 1 month).
   - Three people are working on the project (one works on a full-time basis (€ 1,750 per month) and two on an interim basis (€ 100 per day; 20 days in a month)).
   - The total project duration is 4 months.
   - During the last 2 weeks of the project (weeks 3 and 4 of month 4), only one of the two interims is still working on the project. The other interim and full time worker are no longer needed for the project.

   The total activity cost is equal to ...
   (a) € 28,500
   (b) € 29,500
   (c) € 30,125
   (d) € 31,000
5. Which of the following factors is not part of the total activity cost?
   (a) Cost/unit * resource demand * duration (variable consumable resource with units expressed in time units)
   (b) Cost/unit * work content (renewable resources)
   (c) Cost/unit * duration * #units
   (d) Fixed activity cost

# Chapter 5
# Scheduling Techniques

## Critical Path Scheduling

### *BS10: Activity Slack*[1]

Scheduling a project without resources boils down to a sequencing problem where activities are iteratively scheduled while respecting the precedence relations between them. It results in the detection of the critical path, which refers to a subpart of the project network containing the activities that are critical to the project objective. In this article, the scheduling objective is assumed to be the minimization of the total project duration. Figure 5.1 displays a project network with six activities (A–F) and an activity duration estimate displayed above each node. This example will be used throughout this article to calculate the critical path.

The determination of the critical path of a project requires three steps that are summarized along the following lines and discussed in the remainder of this article:

- Construct an earliest start schedule (ESS)
- Construct a latest start schedule (LSS)
- Calculate the activity slack

#### Earliest Start Schedule (ESS)

The earliest start of each activity can be calculated using forward calculations in the project network and is equal to the maximum of the earliest finish times of all its predecessor activities. The earliest finish of an activity is defined as its earliest

[1]This section is also available as an article entitled "Scheduling projects: How to determine the critical path using activity slack calculations?" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_5

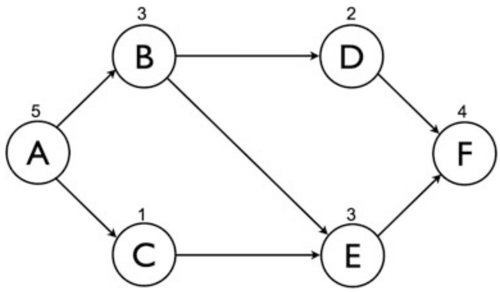



**Fig. 5.1** An example project network

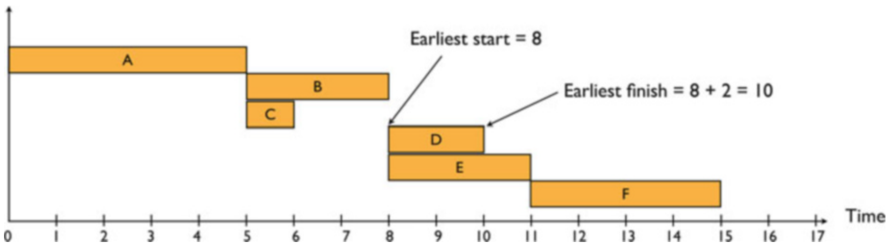



**Fig. 5.2** An earliest start schedule for the example project

start time increased with its duration estimate. Figure 5.2 displays the ESS for the example project of Fig. 5.1, starting from the first activity A and working forwards to the last activity F, resulting in a total project duration of 15 time units.

### Latest Start Schedule (LSS)

The latest finish of each activity can be calculated in an analogous way, using backward calculations, starting from the project deadline at the last activity of the project found by the ESS (that is equal to 15). It is equal to or less than the latest start of all its successor activities. The latest start of an activity is defined as its latest finish time decreased by its duration estimate. Figure 5.3 displays THE LSS, starting from the last activity F and working backwards to the first activity A, resulting in a total project duration of 15 time units.

### Activity Slack

The amount of slack (or float) associated with each activity is used to denote the free time of each activity within the ESS and LSS. It denotes the amount of time each activity can be delayed without violating the entire project duration. The slack of an

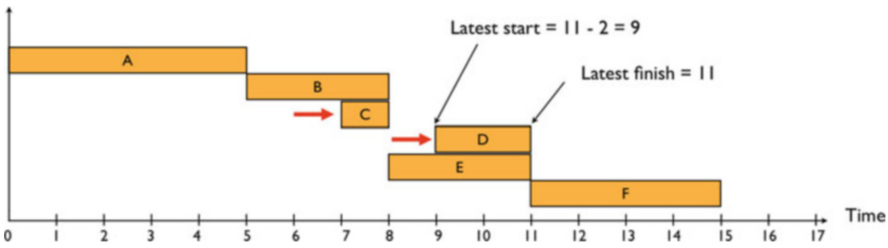



**Fig. 5.3** A latest start schedule for the example project

activity can be calculated as the difference between its latest start and earliest start time, or alternatively, as the difference between its latest and earliest finish time.

Activities with zero slack cannot be delayed without affecting the entire project duration and are called critical activities. The critical path consists of a path of critical activities and is given by activities A, B, E and F.

Activities that lie on the critical path cannot be delayed without delaying the entire project duration. Since time is an important objective in scheduling, the critical path is where the project manager has to focus on. It helps the manager to calculate the minimum length of time in which the project can be completed, and which activities should be prioritized to complete by that project deadline. In order to finish a project on time, the critical path calculations help the project manager to focus on the essential activities to which attention and resources should be devoted. It provides an effective basis for the scheduling and monitoring of project progress.

## Questions

1. The critical path of a project network is the longest path in the network between the start and end node.

   (a) True
   (b) False

2. The earliest start of an activity is equal to the minimum of the earliest finish times of all its predecessors.

   (a) True
   (b) False

3. The larger the slack of an activity, the more time this activity can be delayed without violating the entire project duration.

   (a) True
   (b) False

4. If an activity is critical, the start time of that activity in the earliest start schedule is equal to the finish time in the latest start schedule minus the duration of that activity.
   (a) True
   (b) False

## *BS11: CPM*[2]

The Critical Path Method (CPM) is a project scheduling technique to analyze and represent the tasks involved in completing a given project. It incorporates a trade-off between an activity's duration and cost and relies on concepts similar to the program evaluation and review technique (PERT, see section "*BS15: PERT*" on page 59).

In this article, three CPM related topics will be discussed:

- Time/cost trade-offs in activities
- Scheduling objectives of CPM
- The Project Scheduling Game (PSG)

### Time/Cost Trade-Offs

The critical path method assumes that the duration of an activity depends on the amount of resources assigned to it, and incorporates a trade-off between its duration and the cost of the assigned resources. More precisely, the more resources have been assigned to the activity, leading to an overall increase in the cost, the lower the expected duration of the activity.

This time/cost trade-off is given in Fig. 5.4. The CPM assumes four pieces of information for each project activity, as follows:

- Normal duration: The maximum duration of the activity
- Crash duration: The minimum duration of the activity
- Normal cost: The cost associated with the normal duration
- Crash cost: The cost associated with the crash duration

The normal duration is equal to the minimal expected duration when the activity is performed with the lowest possible amount of resources, leading to the lowest total activity cost (normal cost). The crash duration is the shortest activity duration, when the maximum amount of resources has been assigned to the activity,

[2]This section is also available as an article entitled "The Critical Path Method (CPM): Incorporating activity time/cost trade-offs in a project schedule" published online at PM Knowledge Center.

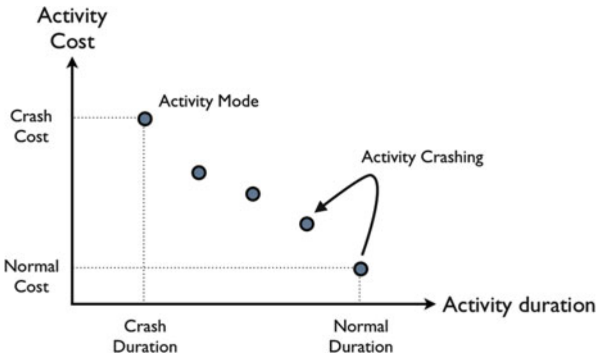



**Fig. 5.4** Time/cost trade-offs for a project activity

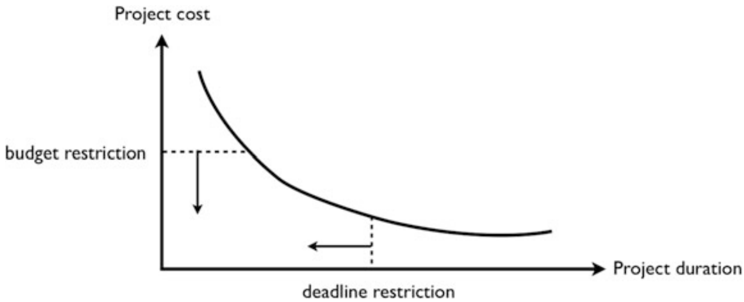



**Fig. 5.5** The scheduling objective of the CPM: time or cost

leading to the highest activity cost (crash cost). Each dot in Fig. 5.4 represents a so-called activity mode to represent the list of time/cost combinations for this activity.

The aim of the critical path method is to schedule the project under a pre-defined scheduling objective, i.e. the choice of an activity mode for each activity to optimize a scheduling objective. A decrease in an activity's duration and the corresponding increase in the activity cost is known as activity crashing.

### Scheduling Objectives

Scheduling a project requires a scheduling objective in order to optimize the project schedule according to the wishes and needs of the project manager (see section "*BS22: What Is My Scheduling Objective?*" on page 88). The CPM has a dual view on the project schedule, i.e. a time or cost point-of-view, resulting in two possible scheduling objectives. Figure 5.5 illustrates the time/cost profile on the project level as a result of selected time/cost combinations (i.e. modes) for each project activity. The two scheduling objectives are given along the following lines:

- Deadline objective: The objective is to determine the activity durations and to schedule the activities in order to minimize the project duration, given a predefined total budget restriction.
- Budget objective: The objective is to determine the activity durations and to schedule the activities in order to minimize the project costs, given a predefined project deadline restriction.

## Project Scheduling Game (PSG)

The Project Scheduling Game (PSG) is an IT-supported simulation game that illustrates the characteristics of scheduling a real-life project with discrete time/cost trade-offs in the project activities as described in this article. The project is based on a sequence of activities for a large real-life project at a Belgian company. The participant (manager) of the game has to construct a dynamic project schedule for the discrete time/cost trade-off critical path method. By allocating nonrenewable resources (i.e. money) to a particular activity, the manager decides about the duration and corresponding cost of each network activity. The manager schedules the project with the negotiated project deadline in mind, focusing on the minimization of the total project cost. For more information, see section "*BS14: The Project Scheduling Game*" on page 56.

## Questions

1. In the well-known time/cost trade-off scheduling problem, the minimization of the total project duration is the only existing objective.

   (a) True
   (b) False

2. In the well-known time/cost trade-off scheduling problem, a lower activity duration corresponds with a lower activity cost.

   (a) True
   (b) False

3. In the well-known time/cost trade-off scheduling problem, the crash cost is the highest possible cost for an activity.

   (a) True
   (b) False

**Table 5.1** Example project data

| Network | | Normal | | Crash | | Crashing cost |
|---|---|---|---|---|---|---|
| Activity | Successors | Time | Cost (€) | Time | Cost (€) | per time unit (€) |
| A | B, C, D | 3 | 6,000 | 2 | 8,000 | 2,000 |
| B | F | 5 | 12,000 | 4 | 13,500 | 1,500 |
| C | E | 5 | 16,000 | 3 | 22,000 | 3,000 |
| D | E | 4 | 8,000 | 2 | 10,000 | 1,000 |
| E | F | 2 | 6,000 | 1 | 7,500 | 1,500 |
| F | – | 3 | 14,000 | 1 | 20,000 | 3,000 |

4. The time/cost trade-off scheduling problem with a deadline objective aims at finding activity durations and costs that minimize the total project duration subject to a given predefined total cost constraint.
   (a) True
   (b) False
5. Consider the project data in Table 5.1 with data for normal and crash times and costs for each activity. Calculate the critical path length for both the normal and crash data.
   (a) The critical path length is equal to 13 for the normal and 7 for the crash data.
   (b) The critical path length is equal to 13 for the normal and 6 for the crash data.
   (c) The critical path length is equal to 11 for the normal and 7 for the crash data.
   (d) The critical path length is equal to 11 for the normal and 6 for the crash data.
6. Consider the project data in Table 5.1 with data for normal and crash times and costs for each activity. Calculate the minimum total project cost for a maximum critical path length and the maximum total project cost for a minimum critical path length.
   (a) The minimum cost is € 62,000 and the maximum cost is € 81,000.
   (b) The minimum cost is € 60,000 and the maximum cost is € 81,000.
   (c) The minimum cost is € 62,000 and the maximum cost is € 80,000.
   (d) The minimum cost is € 60,000 and the maximum cost is € 80,000.

## *BS12: Slack Definitions*[3]

The determination of the critical path of a project requires forward and backward calculation steps (see section "*BS10: Activity Slack*" on page 43) and allows for

[3]This section is also available as an article entitled "Activity slack: Total, safety and free slack definitions" published online at PM Knowledge Center.

the calculation of three different slack values. In this article, the following three definitions of activity slack will be discussed:

- Total slack
- Safety slack
- Free slack

The three definitions will be illustrated on the example project depicted in Fig. 5.1. The values for the activity slack depend on the predefined project deadline and therefore, a distinction is made between the following two situations:

- When no project buffer is added, D(ESS) = D(LSS): The project deadline is equal to the minimal critical path length.
- When a project buffer is added on top of the minimum critical path length, D(ESS) $\neq$ D(LSS): The project deadline is set to a later time instance than the minimal critical path length.

With D(ESS) the project duration of an earliest start schedule (every activity as soon as possible) and D(LSS) the project duration of a latest start schedule (every activity as late as possible).

Figure 5.6 displays an ESS with a critical path length of 15 time periods, and hence, D(ESS) = 15. Figure 5.7 displays a LSS with a predefined project deadline of 19 time periods (D(LSS) = 19)). In this case, a project buffer of 4 time units is added to the project as an extra safety time margin.

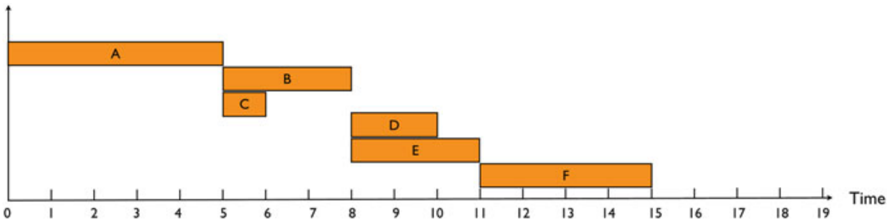



**Fig. 5.6** An earliest start schedule for the example project (critical path length = 15 time units)

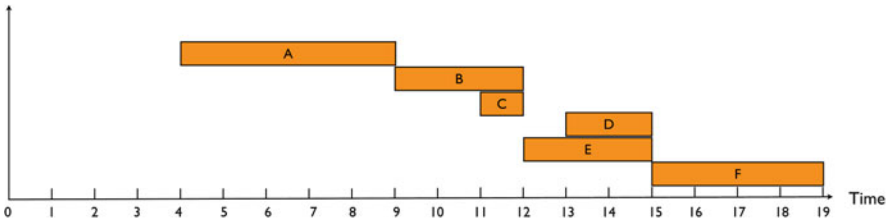



**Fig. 5.7** A latest start schedule for the example project (within a predefined project deadline of 19 time periods)

**Table 5.2** The earliest/latest start/finish times for two different project schedules

| | D(ESS) = D(LSS) = 15 | | | | D(ESS) = 15 ≠ D(LSS) = 19 | | | |
|---|---|---|---|---|---|---|---|---|
| | ES | LS | EF | LF | ES | LS | EF | LF |
| A | 0 | 0 | 5 | 5 | 0 | 4 | 5 | 9 |
| B | 5 | 5 | 8 | 8 | 5 | 9 | 8 | 12 |
| C | 5 | 7 | 6 | 8 | 5 | 11 | 6 | 12 |
| D | 8 | 9 | 10 | 11 | 8 | 13 | 10 | 15 |
| E | 8 | 8 | 11 | 11 | 8 | 12 | 11 | 15 |
| F | 11 | 11 | 15 | 15 | 11 | 15 | 15 | 19 |

### Slack Calculations

In Table 5.2, the earliest and latest start and finish times are displayed for the project network of Fig. 5.1 under two scenarios. In a first scenario, the predefined project deadline is equal to 15, i.e. D(ESS) = D(LSS) = 15 while in a second scenario, the project deadline is set equal to 19 time periods, and hence, D(ESS) = 15 ≠ D(LSS) = 19. The values in the table will be used to define and calculate the three different slack values.

### Total Slack

The total slack is equal to the allowable delay of an activity *i* without causing a violation of the project deadline and can be calculated as

$$\mathrm{LF}_i - \mathrm{EF}_i = \mathrm{LS}_i - \mathrm{ES}_i$$

with $\mathrm{ES}_i$ and $\mathrm{EF}_i$ the earliest start and earliest finish of each activity *i* using forward calculations (i.e. obtained from the ESS). Similarly, the $\mathrm{LS}_i$ and $\mathrm{LF}_i$ are used to denote the latest start and latest finish of activity *i* using backward calculations (i.e. obtained from the LSS).

The total slack is given in Table 5.3 for a project deadline of 15 and 19. Note that in case D(ESS) = D(LSS), the activities with a total slack value equal to zero belong to the critical path.

### Safety Slack

The safety slack is equal to the allowable delay of an activity *i* when all predecessors finish as late as possible and can be calculated as

$$\mathrm{LS}_i - \max_{\text{all predecessors } j}(\mathrm{LF}_j)$$

**Table 5.3** The three slack versions for two different project schedules

| | D(ESS) = D(LSS) = 15 | | | D(ESS) = 15 ≠ D(LSS) = 19 | | |
|---|---|---|---|---|---|---|
| | Total slack | Safety slack | Free slack | Total slack | Safety slack | Free slack |
| A | 0 | 0 | 0 | 4 | n.a. | 0 |
| B | 0 | 0 | 0 | 4 | 0 | 0 |
| C | 2 | 2 | 2 | 6 | 2 | 2 |
| D | 1 | 1 | 1 | 5 | 1 | 1 |
| E | 0 | 0 | 0 | 4 | 0 | 0 |
| F | 0 | 0 | 0 | 4 | 0 | n.a. |

*n.a.* not available

with $LF_j$ the latest finish time of an activity $j$ that precedes activity $i$. The safety slack values are given in Table 5.3 for the two project deadlines.

### Free Slack

The free slack of an activity $i$ is equal to the allowable delay of an activity that has no effect on the earliest start ES of a successor activity and can be calculated as

$$\min_{\text{all successors } j}(ES_j) - EF_i$$

with $ES_j$ the earliest start time of an activity $j$ that succeeds activity $i$. The free slack values are given in Table 5.3 for the two project deadlines.

### Questions

1. Three types of slack are discussed in the article.
   (a) True
   (b) False
2. When a buffer is added to the project, the deadline decreases.
   (a) True
   (b) False
3. Consider the project network in Fig. 5.8 with activity duration displayed above each node. What is the total slack of activity 5, assuming that the project deadline equals the critical path length?
   (a) 2
   (b) 3
   (c) 4

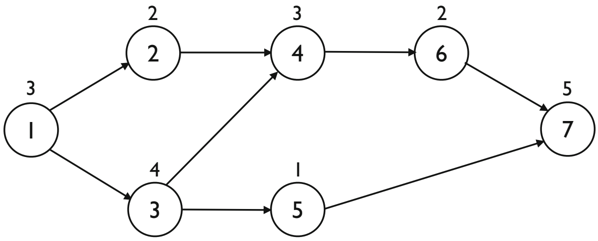



**Fig. 5.8** Example project network

4. What is the safety slack of activity 5, assuming that the project deadline equals the critical path length for the project of Fig. 5.8?
   (a) 1
   (b) 2
   (c) 4
5. What is the total slack of activity 3, assuming that a project buffer of 3 time units is added for the project of Fig. 5.8?
   (a) 0
   (b) 1
   (c) 3
   (d) 5
6. What is the free slack of activity 2, assuming that a project buffer of 3 time units is added for the project of Fig. 5.8?
   (a) 0
   (b) 1
   (c) 2
   (d) 3
   (e) 5
7. The total, safety and free slack of an activity are …
   (a) never the same.
   (b) always the same if the project deadline equals the critical path duration.
   (c) always the same if the project deadline is larger than the critical path duration.
   (d) always the same.
   (e) sometimes the same, but they can be different too.

## *BS13: Anomalies*[4]

The critical path is a series of project activities with a zero slack value connected by precedence relations (see section "*BS10: Activity Slack*" on page 43). It contains all activities that are critical for the project duration and results in the generally accepted critical path rule:

> **Critical path rule**
> "Increasing (decreasing) the duration of a critical activity leads to an increase (decrease) in the critical path length and the project duration"

However, in case more general precedence relations are used in a project network (the four types of relation are start-start, start-finish, finish-start and finish-finish, see section "*BS3: Precedence Relations*" on page 16), a discrepancy or deviation from the established critical path rule might occur, leading to counterintuitive and often conflicting results.

This anomaly is illustrated by the three-activity project network as displayed in Fig. 5.9. The project network on top contains three activities with a duration equal to 6, 6 and 7 for activities 1, 2 and 3, respectively. Since the critical path length is equal to 10 days, as displayed in the Gantt chart at the top of Fig. 5.10, the project manager can decide to reduce the duration of an activity by 50 % in order to decrease the total project duration. The new project network is shown in the bottom network of Fig. 5.9.

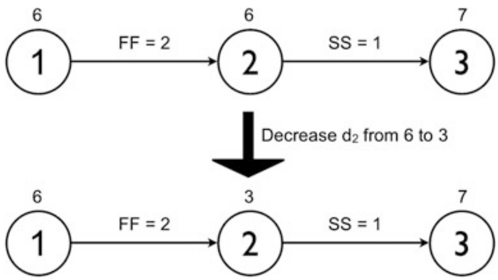



**Fig. 5.9** An example project network where the duration of activity 2 will be decreased by 50 %

[4]This section is also available as an article entitled "The critical path with nonzero time-lags: An anomaly or just a mistake?" published online at PM Knowledge Center.

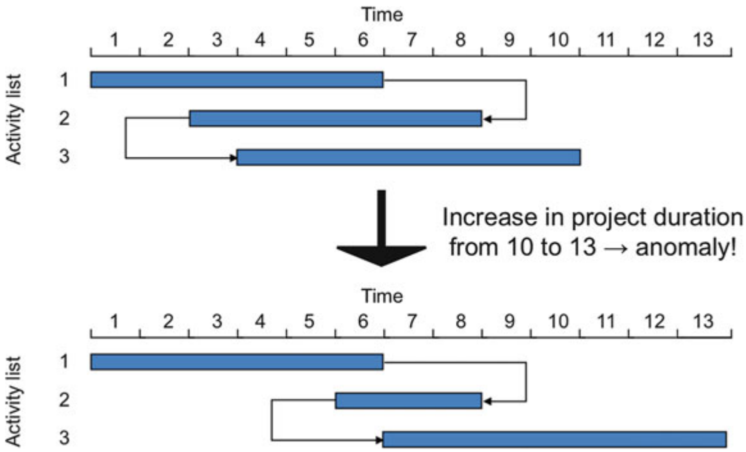



**Fig. 5.10** An earliest start schedule for the example project

The bottom Gantt chart of Fig. 5.10 shows the counterintuitive result of the decrease in the duration of activity 2 that can be summarized in the following anomaly:

> **Anomaly in the critical path rule**
> "Increasing (decreasing) the duration of a critical activity leads to a decrease (increase) in the critical path length and the project duration"

In the example of Fig. 5.9, increasing the duration of a critical activity results in a decrease of the project duration, as follows:

- If $d_2 = 6$ then the project duration $= 10$
- If $d_2 = 3$ then the project duration $= 13$

## Questions

1. Increasing the duration of a critical activity always leads to a project duration increase with minimal finish-start precedence relations.

   (a) True
   (b) False

2. Increasing the duration of a critical path activity always leads to a project duration increase with generalized precedence relations.

   (a) True
   (b) False

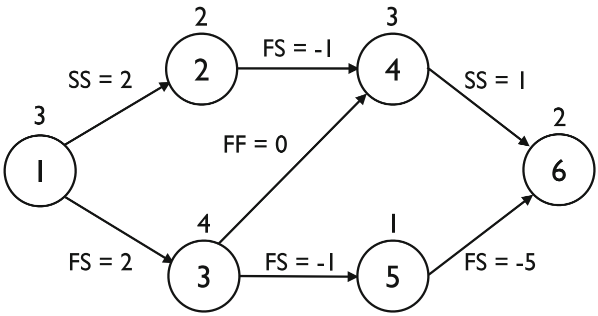



**Fig. 5.11** An example activity-on-the-node project network

3. Consider the project network in Fig. 5.11 with the durations above each node expressed in days. The minimal project duration is equal to . . .
   (a) 9 days
   (b) 10 days
   (c) 11 days
   (d) 12 days
4. When decreasing the duration of activity x in Fig. 5.11 by 1 day, the project duration will increase by 1 day, which is clearly an anomaly. The activity x is equal to . . . (multiple answers possible)
   (a) activity 1
   (b) activity 2
   (c) activity 3
   (d) activity 4
   (e) activity 5
   (f) activity 6

## *BS14: The Project Scheduling Game*[5]

The Project Scheduling Game (PSG) is an IT-supported simulation game that illustrates the characteristics of scheduling a real-life project with discrete time/cost trade-offs in the project activities as described in this article. The project is based on a sequence of activities for a large real-life project at a Belgian company. The participant (manager) of the game has to construct a dynamic project schedule for the discrete time/cost trade-off critical path method. By allocating nonrenewable

[5]This section is also available as an article entitled "The Project Scheduling Game (PSG): Time/cost trade-offs on a computer screen" published online at PM Knowledge Center.

resources (i.e. money) to a particular activity, the manager decides about the duration and corresponding cost of each network activity. The manager schedules the project with the negotiated project deadline in mind, focusing on the minimization of the total project cost.

In this article, three CPM and PSG related topics will be discussed:

- Critical path method
- Game settings
- Learning objectives

## Critical Path Method

The Critical Path Method (CPM) is a project scheduling technique to analyze and represent the tasks involved in completing a given project. It assumes that the duration of an activity depends on the amount of resources assigned to it, and incorporates a trade-off between its duration and the cost of the assigned resources (see section *"BS11: CPM"* on page 46).

## Game Settings

The focus of the game lies on the scheduling and control phases of the project life cycle, as illustrated in Fig. 5.12. More precisely, it is the aim of the player to follow an iterative approach, known as reactive scheduling, that compares the project baseline schedule with the current project performance (simulated during the execution phase) in order to control the project and take corrective actions in case the project objective is in danger.

The game consists of several phases that require periodic inputs from the game player.

The initial project baseline schedule is proposed by the game developer and is based on an activity network that is accessible by the player of the game. A change

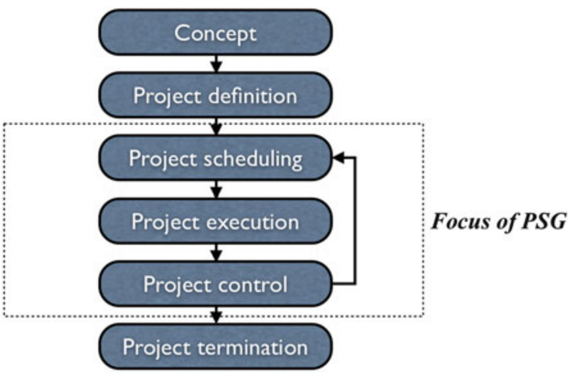



**Fig. 5.12** The project life cycle used in the Project Scheduling Game

in the project deadline requires an update in the baseline schedule, which is the task of the game player. The actions that can be taken are known as activity crashing and lie in the heart of the CPM principle.

The game simulates periodic project progress in which uncertain events might occur. Changes to the original activity durations and costs lead to deviations from the initial baseline schedule and endanger the project objective. The player has to evaluate periodic review reports and re-baseline the unfinished activities of the project schedule in order to bring the project back on track.

After a predefined number of iterative runs, the game reports the final project status, in terms of a total project duration and cost, and the project with the lowest total cost can be considered as the best project.

## Learning Objectives

Since PSG is used as an educational tool for project management students, project managers and/or project management software users, it can be used for a wide variety of project management topics, both on a basic as well as a more advanced level.

Topics which can be discussed as a result of playing the game are, amongst others:

- Network analysis: Use of precedence relations, activity networks, and project network related topics.
- Discussion of the meaning of critical path, including the calculation of slack, project deadlines and total costs.
- The need for risk management and the trade-off between project scheduling efficiency and incorporation of buffers against risk.
- The concept of activity crashing in the critical path method.

The game is a part of the ProTrack software (www.protrack.be) and contains both a teacher version (with access to the project data) and a player version (with limited access to the original project data).

## Questions

1. The Project Scheduling Game (PSG) has a clear focus on defining the project network structure.

   (a) True
   (b) False

2. The aim of the PSG player is to proactively reschedule activities.
   (a) True
   (b) False
3. Uncertainty is included in the PSG.
   (a) True
   (b) False
4. Activity crashing is used to influence the project execution.
   (a) True
   (b) False

## *BS15: PERT*[6]

The Program Evaluation and Review Technique (PERT) is a project scheduling technique to analyze and represent the tasks involved in completing a given project. It incorporates activity duration variability and relies on similar concepts as the critical path method (CPM, see section "*BS11: CPM*" on page 46).

In this article, the technique will be explained by means of the example project data of Table 5.4. More precisely, the following elements, necessary to analyze a PERT project network, will be discussed:

- The calculation of the activity duration estimates based on three input measures
- The calculation of the activity average and variance
- The calculation of the average critical path
- The use of the central limit theorem
- The presence of the normal distribution

**Table 5.4** An example project for PERT calculations

| Task | Predecessors | a | m | b |
|---|---|---|---|---|
| A | | 2 | 5 | 8 |
| B | A | 1 | 2 | 9 |
| C | A | 0.25 | 0.5 | 3.75 |
| D | B | 1 | 1 | 7 |
| E | B and C | 1 | 2 | 9 |
| F | D and E | 1 | 3 | 11 |

[6]This section is also available as an article entitled "The Program Evaluation and Review Technique (PERT): Incorporating activity time variability in a project schedule" published online at PM Knowledge Center.

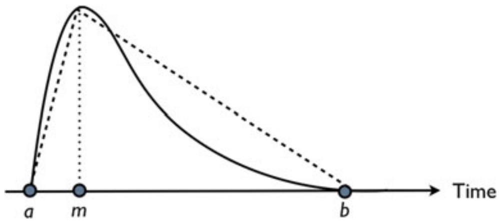



**Fig. 5.13** A PERT (*solid line*) and triangular (*dotted line*) distribution

## Activity Duration Estimates

Activity durations are based on estimates made by human beings and are therefore error-prone. In PERT, the technique requires three duration estimates for each individual activity, as follows (Fig. 5.13):

- Optimistic time estimate ($a$): This is the shortest possible time in which the activity can be completed, and assumes that everything has to go perfect
- Realistic time estimate ($m$): This is the most likely time in which the activity can be completed under normal circumstances
- Pessimistic time estimate ($b$): This is the longest possible time the activity might require, and assumes a worst-case scenario

## Activity Average and Variance

Based on the three estimates, a weighted average and variance is calculated for each activity duration as a measure of the average duration and the corresponding variability, respectively. The weighted average is equal to $(a + 4m + b)/6$ to express that the likeliness that the real activity duration lies close to the realistic estimate ($m$) is larger than the likeliness that it lies closer to the two extreme values $a$ or $b$.

The standard deviation is equal to $(b - a)/6$ and is based on the principles of a three-sigma interval that states that 99.73 % (i.e. almost all) of the observations lie in that interval when the variable is normally distributed. Although it is assumed that the activity duration is beta distributed, the general principle is that the standard deviation assumes that almost all observations (i.e. real durations) will lie between the extreme values $a$ and $b$.

The average durations and their standard deviations are given in the second line above each node of Fig. 5.14.

## The Average Critical Path

The expected duration D of a critical path is equal to the sum of the expected durations of the critical activities, i.e. E(D) $= 5 + 3 + 3 + 4 = 15$. The expected variance V of a critical path is equal to the sum of the variances of the critical

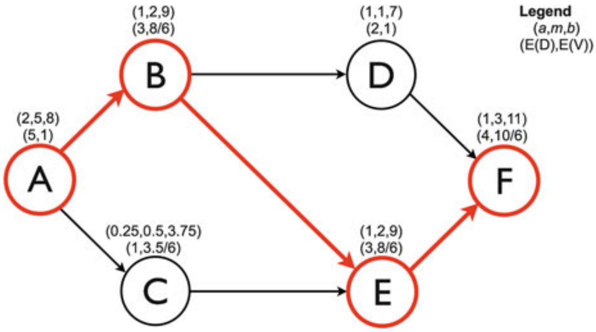



**Fig. 5.14** The project network and the (average) critical path

activities, i.e. $E(V) = 1^2 + (8/6)^2 + (8/6)^2 + (10/6)^2 = 7.33$. Note that only the activities on the (average) critical path are taken into account for the calculation of the variance.

## The Central Limit Theorem

The main idea of the central limit theorem (CLT) is that the average of a sample of observations drawn from a population with any distribution shape is approximately distributed as a normal distribution if certain conditions are met. More precisely, the central limit theorem states that given a distribution with a mean $\mu$ and variance $\sigma^2$, the sampling distribution of the mean approaches a normal distribution with a mean $\mu$ and a variance $\sigma^2/n$ as $n$, the sample size, increases. The amazing and counter-intuitive thing about the central limit theorem is that no matter what the shape of the original distribution is, the sampling distribution of the mean approaches a normal distribution.

In PERT, the project network of Fig. 5.14 represents the population, while the sample that is drawn from that population is equal to the average critical path (highlighted in bold red or grey).

Consequently, the project duration follows a normal distribution with the following parameters:

- Average = E(D) = expected project duration (based on critical path)
- Variance = E(V) = expected variance (of the critical path)

## The Normal Distribution

Using the characteristics of the normal distribution (N(15,7.33)) of the total project duration, basic statistical calculations can be applied to give answers to questions such as:

- What is the probability that the project will be finished before ... ?

- What is the expected project deadline?
- What is a reasonable project duration such that it can be met with a probability of ... %?

As an example, the P(project duration critical path length E(D)) = 50 % (symmetrical normal distribution), which clearly shows that the deterministic critical path underestimates the likely project duration.

Another example shows that the P(project duration $\leq 17$) = P($z \leq (17 - 15)/2.71$) = P($z \leq 0.74$) = 77 % with E(D) = 15 and E(V) = 7.33 using the transformations from the normal distribution N(15,7.33) to a standardized normal distribution N(0,1). This value can be found using standardized normal tables or by the NORMDIST(17,15,$\sqrt{7.33}$,1) function in MS Excel.

## Questions

1. How many duration estimates are used by PERT?
   (a) 2
   (b) 3
   (c) 4
2. The central limit theorem states that the sampling distribution of the average project duration in PERT approaches a normal distribution.
   (a) True
   (b) False
   (c) True, if the critical path consists of enough activities
3. What is the expected project duration of the project data given in Table 5.5?
   (a) 19
   (b) 21
   (c) 24.25
   (d) 26.67
4. What is the expected project standard deviation of the project data given in Table 5.5?
   (a) 2.35
   (b) 2.50
   (c) 2.71
   (d) 3.35
5. What is the probability that the project is completed in 21 time units, using the project data given in Table 5.5?
   (a) 100 %
   (b) 95.48 %
   (c) 80.31 %
   (d) 77.91 %

**Table 5.5** Example project data

| Task | Predecessors | *a* | *m* | *b* |
|---|---|---|---|---|
| 1 | None | 1 | 4 | 7 |
| 2 | 1 | 4 | 5 | 7 |
| 3 | 1 | 1.5 | 2.5 | 3 |
| 4 | 2 and 3 | 1 | 6 | 13 |
| 5 | 3 | 0.5 | 2 | 8.5 |
| 6 | 4 and 5 | 1 | 4 | 4 |

6. What is the expected project duration corresponding with a 90 % probability of completion, using the project data given in Table 5.5?

   (a) 19
   (b) 20
   (c) 21
   (d) 22

## *BS16: A Critical Note on PERT*[7]

PERT is a well-known and widely used project scheduling technique used to create an activity timetable taking variability activity durations into account. It relies on principles borrowed from statistics such as the central limit theorem and the normal distribution (see section "*BS15: PERT*" on page 59).

However, PERT relies, like any other technique, on a number of underlying assumptions that needs to be put into the right perspective. The main PERT assumptions are given along the following lines:

- The activity durations are independent random variables.
- The critical paths are independent (i.e. have no activities in common).
- The normal distribution can be applied ($n$ is sufficiently large) thanks to the central limit theorem.

### Assumptions

Activity durations: The expected variance E(V) of a critical path is assumed to be equal to the sum of the variances of the critical activities, and hence, no covariances between activities are taken into account. Consequently, the activities are assumed to be independent.

[7]This section is also available as an article entitled "A critical note on PERT: A manual simulation study" published online at PM Knowledge Center.

Critical path: Independency of multiple critical paths is treated in PERT as follows: P(project duration ≤ project deadline T) = P(length critical path 1 ≤ project deadline T) * P(length critical path 2 ≤ project deadline T), with P($x$) the probability of event $x$.

Normal distribution: The project duration is assumed to follow a normal distribution, independent of the distribution of the activity duration, as a result of the central limit theorem principles.

## Validity?

In order to verify whether the underlying PERT assumptions are correct, a small manual simulation of a PERT project network will be described, as shown in Fig. 5.15. Each nondummy activity can have three different durations, with an equal probability. Although PERT assumes a beta distribution with an upper limit (b), lower limit (a) and central mode (c), this simplification to three possible durations with an equal probability is made to enumerate all possible scenarios of the project of Fig. 5.15. This complete enumeration is given in Table 5.6.

Following the principles of the PERT calculations, the expected length of the project is equal to the expected length of the critical path(s). Since each activity has an expected duration of 4 time units, the expected project duration is equal to 4 time units. Table 5.6, however, shows that the expected length is equal to 4.44 time units, which is more than the expected duration forecast by PERT.

Therefore, it can be concluded that PERT throws an interesting light on project scheduling since it allows the incorporation of activity time variability in projects, but it not always able to predict the expected project duration for the following reasons:

- PERT fails to allow for time delays resulting from path interactions.
- Since numerous parallel paths may have the potential of becoming critical (because of randomness, some other paths, estimated as being noncritical, may in reality take longer to complete than the purported critical path), the expected project duration might be completely different.

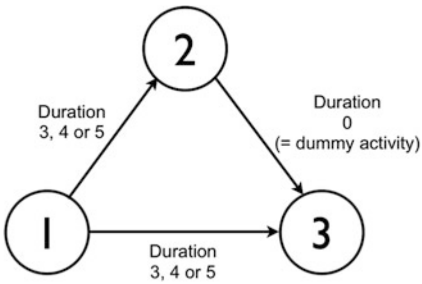



**Fig. 5.15** An activity-on-the-arc project network with 2 activities and 1 dummy activity

**Table 5.6** A complete enumeration of all possible activity durations

| Activity 1–2 | Activity 1–3 | Project duration |
|---|---|---|
| 3 | 3 | 3 |
| 3 | 4 | 4 |
| 3 | 5 | 5 |
| 4 | 3 | 4 |
| 4 | 4 | 4 |
| 4 | 5 | 5 |
| 5 | 3 | 5 |
| 5 | 4 | 5 |
| 5 | 5 | 5 |
| | | Average = 4.44 |

## Questions

1. In case multiple critical path exist, PERT assumes an independency between them.
   (a) True
   (b) False
2. PERT explicitly takes path interactions into account.
   (a) True
   (b) Only implicitly
   (c) False
3. In a project network with multiple parallel paths, the expected project duration is always equal to the expected duration of one of these paths.
   (a) True
   (b) False
4. Consider a project network with three independent critical paths, each with a certain probability of completion before the deadline equal to P1, P2 and P3, respectively. How do you accurately calculate the total probability of project completion before the deadline?
   (a) P1+P2+P3
   (b) P1*P2*P3
   (c) Max (P1, P2, P3)
   (d) Min (P1, P2, P3)

## Resource Scheduling

### *BS17: Priority Rule Based Scheduling*[8]

Project scheduling is the act of constructing a timetable for each project activity, respecting the precedence relations and the limited availability of the renewable resources, while optimizing a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). The presence of resources during project scheduling often leads to a complex scheduling process and hence, simple priority based scheduling rules are used to facilitate this scheduling process. In this article, the priority rule based scheduling approach is discussed along the three following dimensions:

- Priority rule based scheduling: An overview of the general principle of this technique
- Priority rules: Assign priorities to project activities and create a ranking
- Generation schemes: Assign start and finish times to project activities and construct a schedule

#### Priority Rule Based Scheduling

Priority rule based scheduling is a simple and quick heuristic scheduling method used to construct feasible schedules for projects with a regular scheduling objective. Three definitions are explained along the following lines:

- Heuristic scheduling: Since resource constrained project scheduling is often a complex algorithmic task, heuristic scheduling is a quick and easy alternative to generate a project schedule without the guarantee that the best possible theoretical solution is found (see section "*BS21: Validating the Schedule Quality*" on page 83).
- Feasible schedule: A feasible schedule should respect the logic of the project network (i.e. the precedence relations) and the limited availability of resources (i.e. the resource conflicts), as explained in section "*BS7: Critical Path/Chain*" on page 31.
- Regular scheduling objective: A regular scheduling objective is discussed in section "*BS23: Regular and Nonregular Objectives*" on page 90, for which the minimization of the total duration of the project is probably the most well-known and widely used objective.

A priority rule based scheduling approach consists of two components, a priority rule to determine the list with the rankings of activities and a schedule generation

[8]This section is also available as an article entitled "Optimizing regular scheduling objectives: Priority rule based scheduling" published online at PM Knowledge Center.


**Fig. 5.16** The priority rule based scheduling approach to construct a feasible project schedule

scheme to construct a feasible project schedule based on the constructed activity list. In Fig. 5.16, the approach is illustrated graphically and shows that the project data is used to construct a list of activities using a priority rule, which is then transformed by a schedule generation scheme into a feasible project baseline schedule.

### Priority Rules

A priority rule contains information to construct a list of activities that ranks all project activities in a certain order to determine the priorities in which the activities are assigned to the project schedule. Such a list is constructed based on the project data in order to assign priorities to activities, as follows:

- Activity information: information about time or cost estimates of the activities determines the activity priorities.
- Network information: information on the project network logic determines the activity priorities.
- Scheduling information: information obtained from simple critical path scheduling tools determines the activity priorities.
- Resource information: information about the project resources determines the activity priorities.

The constructed list is then used and activities are removed one by one from the list and are put in the schedule during the heuristic scheduling process. To that purpose, a schedule generation scheme is used to determine the exact start and finish time of the activity. Examples on priority rule calculations on a fictitious project network can be found in section "*BS18: Priority Rules*" on page 69.

### Schedule Generation Schemes

A schedule generation scheme makes use of the priority list constructed in the previous step and aims at the generation of a feasible schedule by extending the partial schedule (i.e. a schedule where only a subset of the activities has been assigned a start and finish time) in a stage-wise fashion. At the start of the heuristic scheduling process, the partial schedule is empty and all activities are available to be scheduled. Afterwards, activities are selected according to their priorities and are put in the schedule following the rules of the generation scheme. Basically, two well-known generation schemes are available, as follows:

- Serial schedule generation scheme: Selects the activities one by one from the list and schedules it as soon as possible in the schedule.
- Parallel schedule generation scheme: Selects at each predefined time period the activities available to be scheduled and schedules them in the list as long as enough resources are available.

Illustrations of the use of schedule generation schemes on a project network example are given in section "*BS19: Generation Schemes*" on page 73.

## Questions

1. Heuristic solution procedures always find an optimal solution.
   (a) True
   (b) False
2. Priority rules are used to simplify the project scheduling approach with renewable resources.
   (a) True
   (b) False
3. A priority rule determines the order in which activities are to be scheduled.
   (a) True
   (b) False
4. The activity start times are equal to the values in the priority list.
   (a) True
   (b) False
5. Select the part that is not needed for priority rule based scheduling.
   (a) Schedule generation scheme
   (b) PERT calculations
   (c) Priority rule

## *BS18: Priority Rules*[9]

Priority based project scheduling is a quick and easy heuristic scheduling technique that makes use of two components to construct a resource feasible project schedule, a priority rule and a schedule generation scheme (see section "*BS17: Priority Rule Based Scheduling*" on page 66). In this article, the use and calculations of priority rules will be illustrated on a fictitious project example network. More precisely, the calculations of activity priorities will be based on four sources of information, as follows:

- Activity information: Information about time or cost estimates of the activities determines the activity priorities.
- Network information: Information on the project network logic determines the activity priorities.
- Scheduling information: Information obtained from simple critical path scheduling tools determines the activity priorities.
- Resource information: Information about the project resources determines the activity priorities.

Figure 4.7 on page 37 displays a project network with 11 activities and finish-start precedence relations. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the renewable resource demand. The maximum availability of the resource is equal to 6 units (e.g. people). More information on renewable resources can be found in section "*BS6: Resource Types*" on page 29.

Below, a summary of the most commonly used priority rules is given for each of the first four classes. It should be noted that this is certainly an incomplete list of priority rules since one can think of many other priority rules or extensions or combinations of these rules.

### Activity Information

The construction of an activity list is based on a priority rule taking the characteristics of the project activities into account, such as the duration of each activity. Example priority rules are:

- Shortest Processing Time (SPT): Put the activities in an increasing order of their durations in the list.
- Longest Processing Time (LPT): Put the activities in a decreasing order of their durations in the list.

[9]This section is also available as an article entitled "Optimizing regular scheduling objectives: Priority rule calculations" published online at PM Knowledge Center.

## Network Information

The construction of an activity list is based on a priority rule taking the logic of the network structure into account, i.e. the set of activities and the precedence relations between them. Example priority rules are:

- Most Immediate Successors (MIS): Put the activities with the most direct successors first in the activity list. As an example, activity 4 has two immediate successors (activities 7 and 8) and will therefore come before activity 3 (which has only one immediate successor).
- Most Total Successors (MTS): Put the activities with the most direct and indirect successors first in the activity list. As an example, activity 4 has four successors (activities 7, 8, 10 and 11) and will therefore come before activity 3 (which has three successor activities 6, 9 and 11).
- Least Nonrelated Jobs (LNJ): A job (or activity) is not related to another job if there is no precedence related path between the two activities in the project network. As an example, activity 7 is not related to activities 3, 5, 6, 8, 9, 10 and consequently has six nonrelated jobs. Activities are ranked in an increasing order of their number of nonrelated activities.
- Greatest Rank Positional Weight (GRPW): The GRPW is calculated as the sum of the duration of the activity and the durations of its immediate successors. As an example, activity 4 has a GRPW value equal to 12 and should therefore be put before activity 5 (GRPW value = 10) in the activity list.

## Scheduling Information

Priority rules are used to construct feasible project schedules with resource constraints. However, simple scheduling techniques that ignore these resource constraints, such as the critical path method, can also be used to define new priority rules. Example priority rules are:

- Earliest Start Time (EST): Put the activities in an increasing order of their earliest start in the list.
- Earliest Finish Time (EFT): Put the activities in an increasing order of their earliest finish in the list.
- Latest Start Time (LST): Put the activities in an increasing order of their latest start in the list.
- Latest Finish Time (LFT): Put the activities in an increasing order of their latest finish in the list.
- Minimum Slack (MSLK): Put the activities in an increasing order of their slack value in the list.

More information on the calculation of these earliest/latest start/finish times and the activity slack can be found in section "*BS10: Activity Slack*" on page 43.

### Resource Information

The construction of an activity list is based on a priority rule taking the logic of the network structure as well as information from the resource constraints into account. The resource work content is here defined as the product of the duration of the activity and the resource requirements. Example priority rules are:

- Greatest Resource Work Content (GRWC): Put the activities in a decreasing order of their work content in the list. As an example, activity 4 has a total work content equal to 16 and should therefore be put before activity 3 (work content = 4) in the list.
- Greatest Cumulative Resource Work Content (GCRWC): Put the activities in a decreasing order of the sum of their work content and the work content of all their immediate successors in the list. As an example, activity 4 has a total cumulative work content equal to $16 + 10 + 6 = 32$ and should therefore be put before activity 3 (cumulative work content $= 4 + 12 = 16$) in the list.

More information on the demand of resources in a project can be found in section "*BS8: Linking Resources*" on page 36.

Table 5.7 shows for each of the priority rules the corresponding activity lists for the project network of Fig. 4.7 on page 37. Each activity list can then be used as an input for a schedule generation scheme (see section "*BS19: Generation Schemes*" on page 73) that transforms the project network into a resource feasible schedule.

It should be noted that for each activity list, the network structure has been taken into account. This means that an activity can never be placed in the list before another activity if the latter is a (direct or indirect) predecessor of the former. As an example, the SPT rule should place activity 8 with a duration of 3 time units before activity 5 that has a duration of 7 time units. However, due to the network structure, activity 8 can never be scheduled before activity 5, and therefore, this situation will

**Table 5.7** The priority rules and their corresponding activity lists for the example project of Fig. 4.7 of page 37

| Priority rule | Activity list |
|---|---|
| SPT | [1, 2, 3, 6, 4, 7, 5, 8, 10, 9, 11] |
| LPT | [1, 2, 5, 4, 7, 8, 3, 6, 9, 10, 11] |
| MIS | [1, 2, 4, 3, 5, 6, 7, 8, 9, 10, 11] |
| MTS | [1, 2, 4, 3, 5, 6, 8, 7, 9, 10, 11] |
| LNJ | [1, 2, 4, 3, 5, 8, 10, 6, 9, 7, 11] |
| GRPW | [1, 2, 4, 5, 7, 3, 6, 9, 8, 10, 11] |
| EST | [1, 2, 3, 4, 5, 6, 7, 9, 8, 10, 11] |
| EFT | [1, 2, 3, 4, 6, 5, 7, 8, 9, 10, 11] |
| LST | [1, 2, 3, 5, 6, 4, 9, 7, 8, 10, 11] |
| LFT | [1, 2, 3, 6, 4, 5, 8, 7, 9, 10, 11] |
| MSLK | [1, 2, 3, 5, 6, 9, 4, 8, 10, 7, 11] |
| GRWC | [1, 2, 4, 5, 7, 8, 3, 6, 9, 10, 11] |
| GCRWC | [1, 2, 4, 5, 3, 6, 9, 7, 8, 10, 11] |

never occur. This extra feature facilitates the construction of a resource feasible schedule using the schedule generation schemes but is not a necessary requisite to obtain a feasible schedule.

## Questions

1. The priority rule "Most Immediate Successors" orders the activities based on their total number of successors.
   (a) True
   (b) False
2. The priority rule "Greatest Resource Work Content" orders the activities based on the sum of the resource demand of the activity and the resource demand of its successors.
   (a) True
   (b) False
3. Activity 6 has 1 "Non-Related Jobs" in the project network of Fig. 5.17.
   (a) True
   (b) False
4. What activity in the project network of Fig. 5.17 has the "Most Total Successors"?
   (a) Activity 1
   (b) Activity 2
   (c) Activity 3

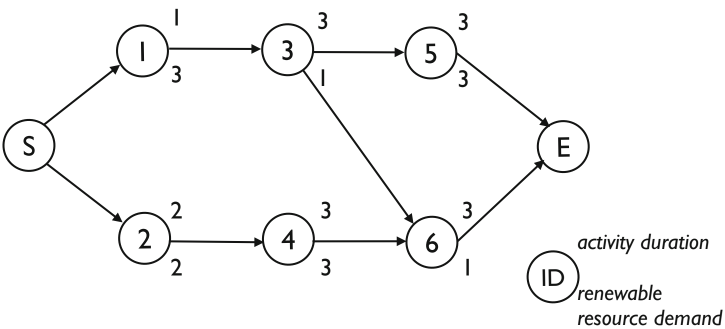



**Fig. 5.17** An example project network

5. What is the order for scheduling the activities in the project of Fig. 5.17 when using the "Earliest Finish Time" rule?
   (a) {2, 1, 3, 5, 4, 6}
   (b) {1, 2, 4, 3, 5, 6}
   (c) {1, 2, 3, 4, 5, 6}
6. What is the order for scheduling the activities in the project of Fig. 5.17 when using the "Greatest Cumulative Resource Work Content" rule?
   (a) {3, 2, 4, 5, 1, 6}
   (b) {4, 5, 2, 1, 3, 6}
   (c) {2, 4, 1, 3, 5, 6}
   (d) (a) and (b), since they will both lead to the same schedule
   (e) (a) and (c), since they will both lead to the same schedule

## *BS19: Generation Schemes*[10]

Priority based project scheduling is a quick and easy heuristic scheduling technique that makes use of two components to construct a resource feasible project schedule, a priority rule and a schedule generation scheme (see section "*BS17: Priority Rule Based Scheduling*" on page 66). In this article, the use of two alternative schedule generation schemes will be illustrated on a fictitious project example network, as follows:

- Serial schedule generation scheme (SSGS): Selects the activities one by one from the list and schedules them as soon as possible in the schedule.
- Parallel schedule generation scheme (PSGS): Selects at each predefined time period the activities available to be scheduled and schedules them in the list as long as enough resources are available.

Figure 4.7 displays a project network with 11 activities and finish-start precedence relations. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand. The maximum availability of the resource is equal to 6 units (e.g. people) (see section "*BS8: Linking Resources*" on page 36).

A schedule generation scheme determines the way in which a feasible schedule is constructed by assigning start times to the project activities. Both schemes remove activities from the priority list and add them to a partial schedule (i.e. a schedule that starts from an empty schedule and that is gradually built up) until all project activities have a start time assigned. A serial schedule generation scheme makes use of an activity-incrementation principle while a parallel schedule generation scheme follows a time-incrementation scheme.

[10]This section is also available as an article entitled "Optimizing regular scheduling objectives: Schedule generation schemes" published online at PM Knowledge Center.

### Serial Schedule Generation Scheme

The activity incrementation approach of a serial schedule generation scheme schedules each activity one at a time and as soon as possible within the precedence and resource constraints. To that purpose,the scheme scans the priority list and selects at each stage the next activity from the priority list in order to schedule it at its first possible start time without violating both the precedence and resource constraints.

This principle can be easily illustrated on the project network of Fig. 4.7, using the GCRWC priority rule that results in the activity list [1, 2, 4, 5, 3, 6, 9, 7, 8, 10, 11] (see Table 5.8 or section "*BS18: Priority Rules*" on page 69). Figure 5.18 displays the resulting resource feasible schedule when each activity is added one by one to the partial schedule using the SSGS until all activities are removed from the list.

**Table 5.8** The priority rules and activity lists used to generate resource feasible schedules by two generation schemes (serial and parallel generation scheme) for the example project of Fig. 4.7

| Priority rule | Activity list | SSGS | PSGS |
|---|---|---|---|
| SPT | [1, 2, 3, 6, 4, 7, 5, 8, 10, 9, 11] | 24 | 24 |
| LPT | [1, 2, 5, 4, 7, 8, 3, 6, 9, 10, 11] | 29 | 29 |
| MIS | [1, 2, 4, 3, 5, 6, 7, 8, 9, 10, 11] | 24 | 24 |
| MTS | [1, 2, 4, 3, 5, 6, 8, 7, 9, 10, 11] | 24 | 24 |
| LNJ | [1, 2, 4, 3, 5, 8, 10, 6, 9, 7, 11] | 24 | 24 |
| GRPW | [1, 2, 4, 5, 7, 3, 6, 9, 8, 10, 11] | 27 | 27 |
| EST | [1, 2, 3, 4, 5, 6, 7, 9, 8, 10, 11] | 24 | 24 |
| EFT | [1, 2, 3, 4, 6, 5, 7, 8, 9, 10, 11] | 24 | 24 |
| LST | [1, 2, 3, 5, 6, 4, 9, 7, 8, 10, 11] | 24 | 24 |
| LFT | [1, 2, 3, 6, 4, 5, 8, 7, 9, 10, 11] | 24 | 24 |
| MSLK | [1, 2, 3, 5, 6, 9, 4, 8, 10, 7, 11] | 26 | 26 |
| GRWC | [1, 2, 4, 5, 7, 8, 3, 6, 9, 10, 11] | 29 | 29 |
| GCRWC | [1, 2, 4, 5, 3, 6, 9, 7, 8, 10, 11] | 26 | 27 |

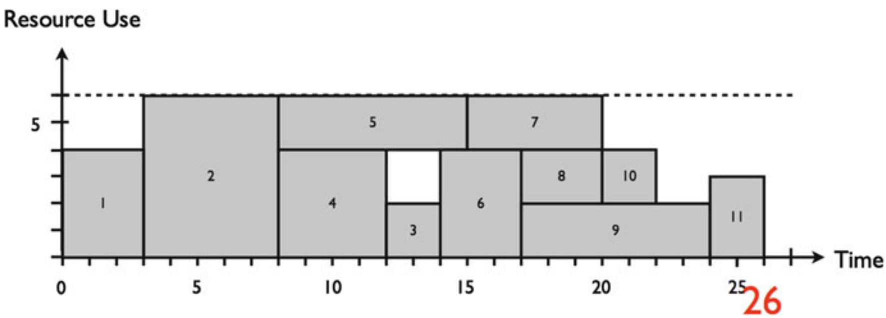



**Fig. 5.18** Project schedule using a serial schedule generation scheme

This dynamic process can also be viewed in the animated SSGS approach that can be found at www.pmknowledgecenter.com/sites/default/files/files/Animation_Serial.mov.

### Parallel Schedule Generation Scheme

A parallel schedule generation scheme iterates over the time horizon of the project (i.e. a time incrementation) instead of iterating over the priority list (i.e. an activity incrementation) and adds activities that are eligible to be scheduled. More precisely, the scheme starts at time point $t = 0$ and schedules activities before the time pointer is increased. It selects at each decision point $t$ the eligible activities $E$ and assigns a scheduling sequence of these eligible activities according to the priority list. At each decision point, the eligible activities are scheduled with a start time equal to the decision point (on the condition that there is no resource conflict). Activities that cannot be scheduled due to a resource conflict are skipped and become eligible to be scheduled at the next decision point $t' > t$, which equals the earliest completion time of all activities active at the current decision point $t$.

This principle can also be easily illustrated on the project network of Fig. 4.7 using the same activity list [1, 2, 4, 5, 3, 6, 9, 7, 8, 10, 11]. The parallel schedule generation scheme starts with an empty schedule and proceeds as follows:

$t = 0$: $E = 1$
→ Schedule 1
→ Increase $t$ to the earliest finish time of the activities in process

$t = 3$: $E = 2$
→ Schedule 2
→ Increase $t$ to the earliest finish time of the activities in process

$t = 8$: $E = 3, 4, 5$ with the schedule sequence = 4-5-3
→ Schedule 4 and 5. 3 cannot be scheduled at $t = 8$ due to a resource conflict
→ Increase $t$ to the earliest finish time of the activities in process

$t = 12$: $E = 3, 7$ with the schedule sequence = 3-7
→ Schedule 3 and 7
→ Increase $t$ to the earliest finish time of the activities in process

$t = 14$: $E = 6$
→ 6 cannot be scheduled at $t = 14$ due to a resource conflict
→ Increase $t$ to the earliest finish time of the activities in process

$t = 15$: $E = 6, 8$ with the schedule sequence = 6-8
→ Schedule 6. 8 cannot be scheduled at $t = 15$ due to a resource conflict
→ Increase $t$ to the earliest finish time of the activities in process

$t = 17$: $E = 8$
→ Schedule 8
→ Increase $t$ to the earliest finish time of the activities in process

$t = 18$: $E = 9$
→ Schedule 9
→ Increase $t$ to the earliest finish time of the activities in process

$t = 20$: $E = 10$
→ Schedule 10
→ Increase $t$ to the earliest finish time of the activities in process

$t = 22$: $E = 11$
→ 11 cannot be scheduled at $t = 22$ due to a resource conflict
→ Increase $t$ to the earliest finish time of the activities in process

$t = 25$: $E = 11$
→ Schedule 11
→ All activities have been scheduled. STOP

Figure 5.19 displays the resulting resource feasible schedule. This dynamic process can also be viewed in the animated PSGS approach that can be found at www.pmknowledgecenter.com/sites/default/files/files/Animation_Parallel.mov.

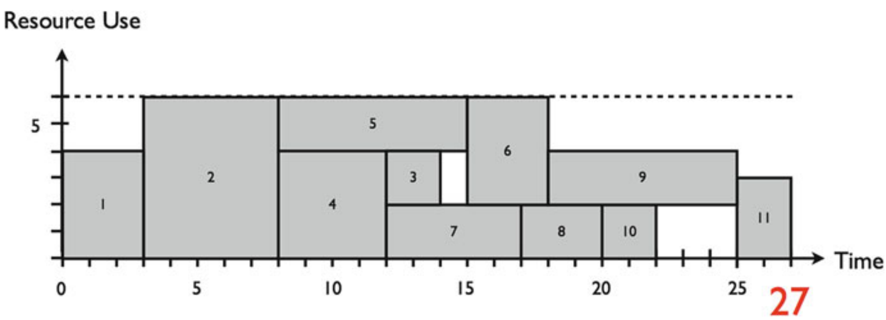



**Fig. 5.19** Project schedule using a parallel schedule generation scheme

## Summary

Table 5.8 gives the total project durations that have been found by using both the serial (SSGS) and parallel (PSGS) schedule generation schemes using the various priority rules as discussed in section "*BS18: Priority Rules*" on page 69. For the sake of clarity, the activity lists that result from using these priority rules on the project data of Fig. 4.7 are also displayed.

## Questions

1. The serial schedule generation scheme schedules all activities as soon as possible for a given priority list.
   (a) True
   (b) False
2. The "Shortest Processing Time" rule can only be used with the parallel schedule generation scheme.
   (a) True
   (b) False
3. The serial and parallel schedule generation schemes make use of a certain priority list generated by a priority rule to construct a feasible schedule. Assume that all possible priority lists are generated for a particular project, and each list is used for both schedule generation schemes to construct several different schedules. In this case, the best possible solution (with a minimum duration) should be found by at least one of the priority lists. This statement is true for ... (Since this is an *advanced question*, you could use the example project in Fig. 5.20 on page 77 to validate your answer.)

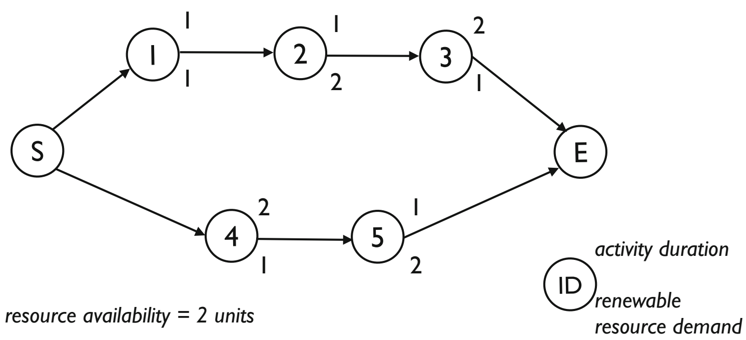



**Fig. 5.20** An example project network

(a) the parallel scheduling generation scheme.
(b) the serial scheduling generation scheme.
(c) both the serial and the parallel schedule generation schemes.

4. Activities 4 and 5 of the project network of Fig. 5.17 shown on page 72 can be scheduled in parallel with a resource availability of 6.

(a) True
(b) False

5. What is the project finish time when both a serial schedule generation scheme and the priority list 1, 3, 2, 5, 4, 6 are used for the project network of Fig. 5.17 shown on page 72? The resource availability is now assumed to be equal to 5 instead of 6.

(a) 9
(b) 11
(c) 13
(d) 15

6. How can the schedule from the previous question be improved?

(a) Swap the order of activities 4 and 5 in the priority list.
(b) Swap the order of activities 4 and 6 in the priority list.
(c) Schedule activity 4 in parallel with 3, the rest of the schedule remains the same.

## *BS20: Lower Bounds*[11]

Minimizing the total duration of a resource constrained project is often a generally accepted project scheduling objective (see section "*BS24: Project Lead Time*" on page 93). However, optimizing a scheduling objective within the presence of limited renewable resources often requires software algorithms. In this article, three lower bound calculations will be discussed, as follows:

- Critical path lower bound: Ignores the resource constraints and only looks at the project network.
- Resource based lower bound: Ignores the project network and only looks at the resource constraints.
- Critical sequence lower bound: Combination of the two previous bounds by looking at both the project network and resource constraints.

[11] This section is also available as an article entitled "What is my scheduling objective? Lower bounds on the total project duration" published online at PM Knowledge Center.

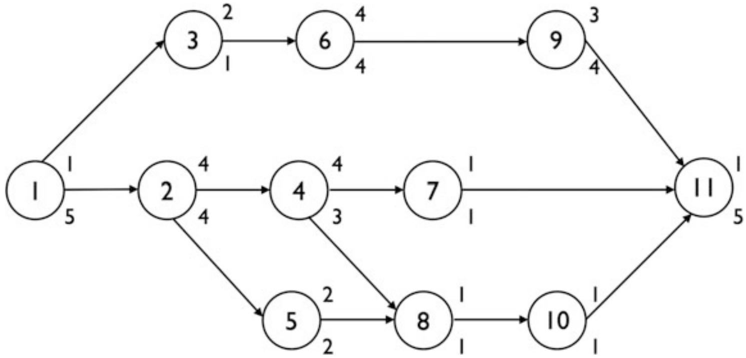



**Fig. 5.21** Project network and activity data

Figure 5.21 displays a project network with 11 activities and finish-start precedence relations. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand. The maximum availability of the resource is equal to 5 units (e.g. people). More information on resource demand and availability is given in section "*BS8: Linking Resources*" on page 36.

## Critical Path Lower Bound (CPLB)

In the critical path calculations proposed by the PERT/CPM scheduling approach, projects are not subject to limited renewable resources, and hence, the construction of a baseline schedule boils down to sequencing all activities according to their precedence relations. The minimal duration of the project is determined by the length of the critical path, which is a sequence of activities in the project network. The critical path of the project network of Fig. 5.21 is equal to 1-2-4-8-10-11 with a total length equal to 12 time units.

Obviously, when the limited availability of the renewable resources is taken into account, the minimal project duration will probably increase (due to the presence of resource conflicts), or remain unchanged (see section "*BS7: Critical Path/Chain*" on page 31). Consequently, the critical path duration of 12 time units is a lower bound on the real project duration of the resource constrained project of Fig. 5.21.

## Resource Based Lower Bound (RBLB)

While the critical path lower bound only takes the precedence relations into account and ignores the presence of limited renewable resources, the resource based lower bound takes an inverse approach. It looks at the total use of resources for all project activities and ignores the precedence relations between the activities. The lower bound is equal to the sum of the work content (= activity duration multiplied by its resource requirement) of all activities divided by the availability of resources, as follows:

- Sum of work content = 1 * 5 + 4 * 4 + 2 * 1 + 4 * 3 + 2 * 2 + 4 * 4 + 1 * 1 + 1 * 1 + 3 * 4 + 1 * 1 + 1 * 5 = 75
- Lower bound = 75/5 = 15

In the example, the resource based lower bound value is bigger than the critical path lower bound value, but this is obviously not always the case since they both have a completely different calculation approach. The highest lower bound is retained as the strongest lower bound, and hence, at this point it can be concluded that a resource feasible schedule must have a duration of minimum 15 time units, or probably higher.

### Critical Sequence Lower Bound (CSLB)

While the critical path lower bound only takes the precedence relations into account but ignores the resources, and the resource based lower bound does exactly the opposite, the critical sequence lower bound considers both the precedence and the resource constraints in a two-step approach:

- Step 1: Consider a critical path in the network and set the CSLB = CPLB
- Step 2: Do for each noncritical activity $i$
  - Determine the earliest start $ES_i$ and latest finish $LF_i$
  - Determine how many time periods $e_i$ this activity can be scheduled consecutively in parallel with the activities on the critical path (between $ES_i$ and $LF_i$)
  - If ($e_i < d_i$) then set $\Delta_i = d_i - e_i$ else $\Delta_i = 0$
- Step 3: CSLB = CPLB + maximum of all $\Delta_i$ values

with

$\Delta_i$ Increase in the critical path length to add activity $i$ in the schedule without generating a resource conflict
$ES_i$ Earliest start of activity $i$
$LF_i$ Latest finish of activity $i$
$e_i$ Number of periods between $ES_i$ and $LF_i$ with the remaining resource availability $\geq$ resource demand for activity $i$
$d_i$ Duration of activity $i$

Figure 5.22 displays the critical path activities with a total length of 12 time units and the earliest start and latest finish of all noncritical activities. For each noncritical activity, the difference between its earliest start and its latest finish is indicated by the arcs above the schedule.

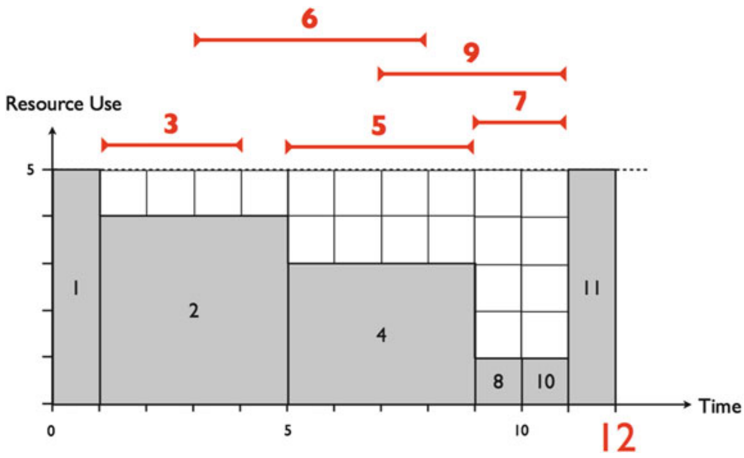



**Fig. 5.22** Schedule with the critical activities and their resource use

**Table 5.9** Parameter calculations for all noncritical activities to calculate the CSLB

| Activity $i$ | $d_i$ | $\text{ES}_i$ | $\text{LF}_i$ | $e_i$ | $\Delta_i$ |
|---|---|---|---|---|---|
| 3 | 2 | 1 | 4 | 3 | 0 |
| 5 | 2 | 5 | 9 | 4 | 0 |
| 6 | 4 | 3 | 8 | 0 | 4 |
| 7 | 1 | 9 | 11 | 2 | 0 |
| 9 | 3 | 7 | 11 | 2 | 1 |

Table 5.9 displays the information about the noncritical activities needed to calculate the critical sequence lower bound. Consider, as an example, activity 9 that has four periods between its earliest start and latest finish, of which only 2 time periods have enough resources (4 units). Consequently, the $e_9$ value is equal to 2 time units. Since $e_9 < d_9$, the $\Delta_9$ parameter is set to 1 to express that the critical path length of 12 must be increased by one time period in order to add activity 9 to the schedule without generating a resource conflict. Activity 6, for example, has no periods between its earliest start and latest finish with enough resources (the activity needs 4 units and maximum 2 are available in Fig. 5.22) and hence, $e_6 = 0$ and $\Delta_6 = 4 - 0 = 4$. Doing these calculations for each activity and taking the maximum of all $\Delta_i$ values leads to the critical sequence lower bound equal to $12 + 4 = 16$ time units. A stepwise calculation of all values is illustrated in the animation available at www.pmknowledgecenter.com/sites/default/files/files/Animation_LowerBounds.mov.

## Conclusion

As a general conclusion, the maximum of all lower bounds is equal to $\max\{12, 15, 16\} = 16$ and hence, any resource feasible schedule must have a total project duration that is equal to 16 time periods, or possibly larger. It is worth noting that the minimal project duration is equal to 17 time units, which is one time unit larger than the best lower bound found in this article. It is up to the reader to construct such a resource feasible schedule using techniques as described in section "*BS17: Priority Rule Based Scheduling*" on page 66. Although lower bound calculations do not lead to a resource feasible schedule, they might provide the user with information on the solution quality of a scheduling tool, as illustrated in section "*BS21: Validating the Schedule Quality*" on page 83.

## Questions

1. In order to calculate the critical path based lower bound, the resource constraints can be ignored but the precedence relationships and the activity durations must be taken into account.
   (a) True
   (b) False
2. In order to calculate resource based lower bound, both the precedence relations and activity durations are taken into account, but the resource constraints can be ignored.
   (a) True
   (b) False
3. In order to calculate the critical path lower bound, you should calculate the …
   (a) longest path in the project network.
   (b) shortest path in the project network.
4. The actual duration of a project will always be larger than the lower bound.
   (a) True
   (b) False
5. If the critical path lower bound (CPLB) is larger than the resource based lower bound (RBLB) for a certain project, the strongest lower bound in this case is the …
   (a) CPLB.
   (b) RBLB.
6. In order to calculate the critical sequence lower bound, the maximum of $\Delta_i$ has to be calculated for all …
   (a) the critical activities.
   (b) the noncritical activities.

**Table 5.10** Example project data

| Activity | Predecessors | Duration | Resource use |
|---|---|---|---|
| 1 | – | 0 | 0 |
| 2 | 1 | 6 | 7 |
| 3 | 1 | 5 | 1 |
| 4 | 1 | 3 | 5 |
| 5 | 3 | 1 | 6 |
| 6 | 3 | 3 | 1 |
| 7 | 3 | 2 | 2 |
| 8 | 4 | 1 | 2 |
| 9 | 2 | 4 | 8 |
| 10 | 5 | 3 | 6 |
| 11 | 7 | 1 | 2 |
| 12 | 6, 10, 11 | 3 | 6 |
| 13 | 8, 12 | 5 | 8 |
| 14 | 9, 13 | 0 | 0 |

7. Consider the project data in Table 5.10 on page 83 with activity and resource information.[12] Assume the availability of the resource is equal to 10 units. Which of the following equations is true (CPLB = critical path lower bound; RBLB = resource based lower bound; CSLB = critical sequence lower bound)?

   (a) CPLB > RBLB > CSLB
   (b) CPLB > CSLB > RBLB
   (c) RBLB > CPLB > CSLB
   (d) RBLB > CSLB > CPLB
   (e) CSLB > CPLB > RBLB
   (f) CSLB > RBLB > CPLB

## *BS21: Validating the Schedule Quality*[13]

Project scheduling is the act of constructing a timetable for each project activity, and differs in complexity due to the presence of renewable resources with limited availability. The construction of a resource feasible project schedule within the restrictions of a limited availability of renewable resources can be done to optimize a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). In this article, the differences between three possible

[12]This example was taken from the book "Project Management with Dynamic Scheduling: Baseline Scheduling, Risk Analysis and Project Control" published by Springer.

[13]This section is also available as an article entitled "Heuristic project scheduling: Validating the quality of a project schedule" published online at PM Knowledge Center.

schedules that can be constructed from project network data are described, as follows:

- Infeasible schedule: A schedule with one or more resource conflicts.
- Feasible schedule: A schedule without any resource conflict. This is also known as a heuristic schedule to denote that it does not contain resource conflicts but the scheduling objective can possibly be improved.
- Optimal schedule: A schedule without any resource conflict and with the best possible scheduling objective value.

The difference between these three possible schedules is explained in detail for scheduling problems with a minimization and a maximization scheduling objective.

Figure 5.23 gives an overview of the three possible schedules and shows that a heuristic solution found by a software tool is a resource feasible solution without any resource conflict but not necessarily the best possible schedule that can be found. The optimal schedule is a feasible schedule with the best possible scheduling objective (the lower (higher), the better in case of a minimization (maximization) objective). Better schedules cannot be found unless resource conflicts are added, which then results in infeasible project schedules. Bounds are calculated on resource infeasible schedules to have an idea about the range of possible scheduling

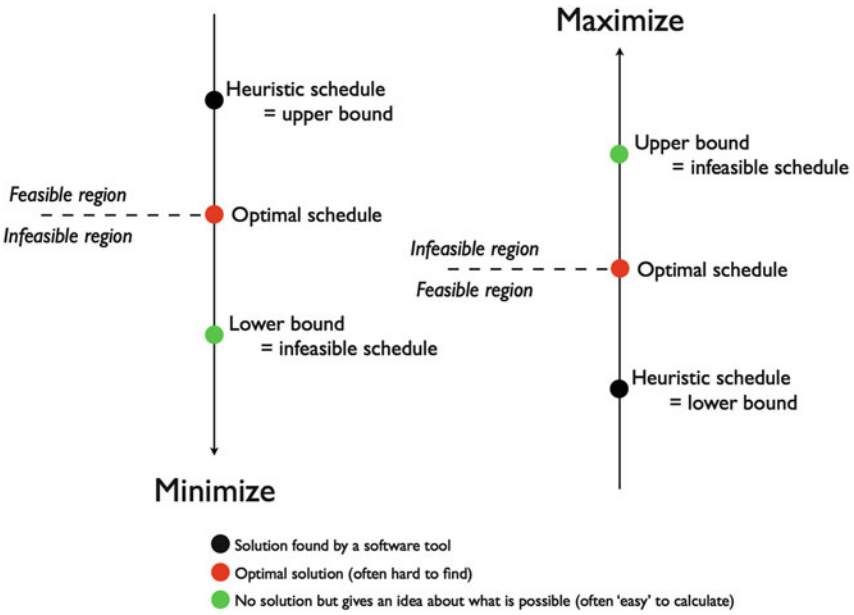



**Fig. 5.23** The gap between a feasible heuristic solution and an infeasible (lower or upper) bound

objectives for the project. Both the calculation of bounds and feasible schedules can be used to define the gap, as follows:

GAP
=
the difference between the scheduling objective of a heuristic (= feasible) schedule
and the value of a (lower or upper) bound.

Since both the generation of heuristic schedules and the calculation of bounds can be done by simple and fast algorithms, the gap gives an indication of the room for improvement one can make by starting with a feasible schedule and by making small changes and adaptations to the schedule to improve the value of the scheduling objective.

### Minimization Objective

The most well-known scheduling minimization objective is the minimization of the total duration of a project baseline schedule, as described in section "*BS24: Project Lead Time*" on page 93.

Infeasible schedules can be easily generated using simple and straightforward lower bound calculations as shown in section "*BS20: Lower Bounds*" on page 78. Feasible heuristic schedules can be constructed by quick and easy methods such as priority rule based scheduling techniques, illustrated in section "*BS17: Priority Rule Based Scheduling*" on page 66. Optimal schedules are harder to find due to the inherent complexity of resource constrained project scheduling and are often the subject of research at academic institutions.

The top left corner in Fig. 5.24 displays a project network with 11 activities and finish-start precedence relations. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand. The maximum availability of the resource is equal to 5 units (e.g. people).

The optimal schedule is given by the schedule on the top right corner of the picture. Its optimal duration of 17 days lies in between the feasible schedule found by the Greatest Cumulative Resource Work Content (GCRWC) priority rule and the best lower bound calculated by the resource based lower bound (see section "*BS20: Lower Bounds*" on page 78). The heuristic schedule has a total duration of 19 days displayed at the bottom right corner of the picture. The GCRWC priority rule results in the activity list [1, 2, 3, 6, 9, 4, 7, 5, 8, 10, 11] that can be transformed in the heuristic schedule using a serial generation scheme as discussed in section "*BS17: Priority Rule Based Scheduling*" on page 66. The gap between the heuristic schedule and the lower bound is equal to 3 days.

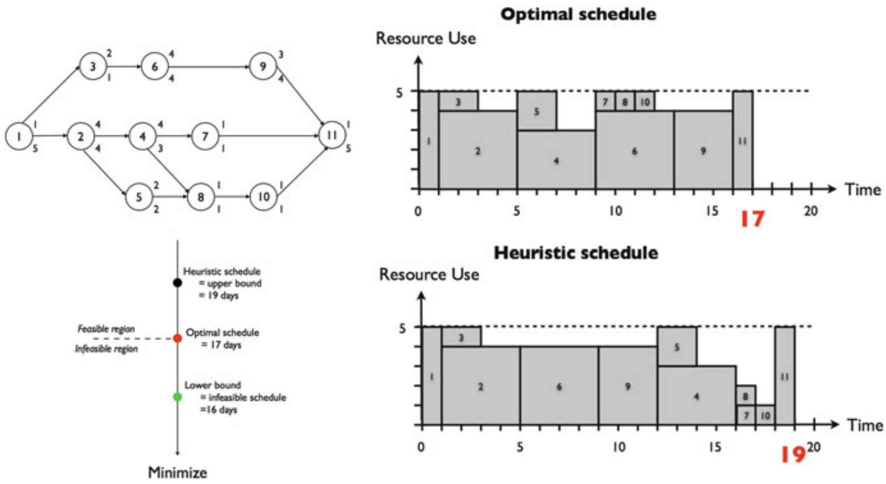



**Fig. 5.24** An example project network with its optimal schedule and the gap between a feasible schedule and the resource based lower bound

It should be noted that this general approach is also applicable to scheduling objectives other than the time minimization. While the time minimization is the most dominant scheduling objective, other minimization objectives for which no lower bounds are available are given in section "*BS26: Resource Idle Time*" on page 100 and section "*BS27: Resource Leveling*" on page 103. Moreover, while the heuristic solutions found by priority rules (see section "*BS17: Priority Rule Based Scheduling*" on page 66) are restricted to regular scheduling objectives, other heuristic techniques such as the Burgess and Killebrew algorithm can be used to construct feasible schedules with nonregular scheduling objectives. This algorithm is outside the scope of this book. The difference between regular and nonregular scheduling objectives is explained in section "*BS23: Regular and Nonregular Objectives*" on page 90.

### Maximization Objective

When the scheduling objective is a value to maximize instead of to minimize, the gap calculation is similar but opposite to the approach for a minimization objective. The heuristic schedule is now a lower bound since improvement from a heuristic feasible schedule to an optimal schedule will lead to an increase (i.e. improvement) of the scheduling objective. Likewise, the calculation of a bound using an infeasible schedule is now an upper bound since it ignores resource constraints and allows resource conflicts. A typical and well-known scheduling maximization objective is the net present value optimization objective as discussed in section "*BS25: Net Present Value*" on page 96.

## Questions

1. An upper bound for a resource constrained project scheduling problem with a maximization objective might result in a solution with resource conflicts.
   (a) True
   (b) False
2. A lower bound for a minimization solution is a solution found by a heuristic search procedure such as a genetic algorithm.
   (a) True
   (b) False
3. If the optimal solution of a project scheduling problem with the net present value objective is equal to € 800, then ...
   (a) € 1,000 is a valuable upper bound but will not be the result of a resource feasible schedule.
   (b) € 1,000 is a valuable upper bound and will be the result of a resource feasible schedule.
   (c) € 1,000 is a valuable lower bound but will not be the result of a resource feasible schedule.
   (d) € 1,000 is a valuable lower bound and will be the result of a resource feasible schedule.
4. A project manager wants to minimize the duration of a project. He has used an algorithm and found a feasible schedule with a duration of 20 days which is exactly equal to the length of the critical path. He knows that with better algorithms ...
   (a) the optimal duration might be lower.
   (b) nothing will change, since this is the optimal solution.
   (c) the optimal duration might be higher.
5. By applying a heuristic that produces several feasible schedules for a maximization problem, ...
   (a) the gap between the upper bound and the optimal schedule will decrease.
   (b) the gap between the lower bound and the optimal solution will decrease.
   (c) the gap between the lower bound and the upper bound will increase.
6. Assume a lower bound for the project scheduling problem with a time minimization objective. Which of the following statements is correct?
   (a) The optimal solution will have a lower value than this lower bound.
   (b) The schedule resulting from this lower bound might still show some resource conflicts.
   (c) This lower bound has been obtained by using heuristic methods to produce feasible schedules.

## Scheduling Objectives

### *BS22: What Is My Scheduling Objective?*[14]

Project scheduling is the act of constructing a timetable for each project activity, and differs in complexity due to the presence of renewable resources with limited availability. In this article, three important aspects of scheduling will be discussed, as given along the following lines:

- Sequencing: Scheduling with unlimited resources.
- Scheduling: Scheduling within limited resource constraints.
- Scheduling objectives.

A formal definition of project scheduling can be given as follows: It involves the construction of an activity timetable, i.e. the determination of a start and finish time for each project activity, respecting the precedence relations and the limited availability of the renewable resources, while optimizing a predefined scheduling objective.

#### Sequencing

In the absence of renewable resource constraints, project scheduling boils down to activity sequencing by putting each individual activity as soon as possible in the timetable, respecting the precedence relations, resulting in an earliest start schedule. Consequently, in this scheduling approach, it is implicitly assumed that the minimization of the total project lead time is the scheduling objective.

Two well-known techniques that rely on a straightforward activity sequencing approach are the PERT and CPM scheduling techniques. More information can be found in the sections "*BS15: PERT*" on page 59 and "*BS11: CPM*" on page 46.

#### Scheduling

The presence of renewable resources, constrained by their limited periodic availability, leads to a complexity increase during the construction of a project schedule. Due to the limited availability of resources, the straightforward activity sequencing approach often leads to so-called resource conflicts (see section "*BS8: Linking Resources*" on page 36). These conflicts result from over-allocations of renewable

[14]This section is also available as an article entitled "Resource constrained project scheduling: What is my scheduling objective?" published online at PM Knowledge Center.

resources when activities scheduled in parallel require more resources than available. In order to solve these resource conflicts, activities need to be shifted further in time to periods where resources are still available for the activities. The aim of this scheduling approach is to create a so-called resource feasible schedule (i.e. a schedule without any resource conflict) and is often a complex task. Moreover, the construction of such a resource feasible project schedule requires a scheduling objective that needs to be optimized.

## Scheduling Objectives

A scheduling objective is the objective one aims to reach while constructing a resource feasible project schedule. Although time is often considered as the dominant scheduling objective, other objectives are often crucial from a practical point-of-view. A nonexhaustive list of possible objectives is given along the following lines:

- Time: Minimize the total duration of the project.
- Net present value: Maximize the discounted cash flows of project activities.
- Work continuity: Avoid idle time of bottleneck resources in a project.
- Leveling: Avoid resource jumps but try to balance the use of resources.
- Others: Many other scheduling objectives can occur and are often project specific. Moreover, it is logical that in practical environments, a combination of objectives is strived for.

## Questions

1. During project scheduling, an activity timetable is constructed, which assigns a start and finish time to each project activity, taking into account the precedence relations and the limited availability of the renewable resources.
   (a) True
   (b) False
2. The earliest start schedule is always the optimal solution if no resources are taken into account.
   (a) True
   (b) False
3. When the limited availability of renewable resources has to be taken into account, the complexity of the scheduling procedure increases.
   (a) True
   (b) False

4. Which of the following scheduling objectives tries to avoid idle time of bottleneck resources in a project?
   (a) Net present value
   (b) Work continuity
   (c) Leveling
   (d) Time

5. Which of the following statements is correct? (multiple answers possible)
   (a) It is only possible to strive for one objective at a time, not to combine them.
   (b) Sequencing is scheduling in the absence of renewable resources constraints.
   (c) A resource feasible schedule does not take precedence relations into account.
   (d) Resource conflicts can be resolved by postponing activities.

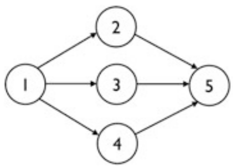
## *BS23: Regular and Nonregular Objectives*[15]

Project scheduling is the act of constructing a timetable for each project activity, respecting the precedence relations and the limited availability of the renewable resources, while optimizing a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). Although time is often considered as the dominant scheduling objective, other objectives are often crucial from a practical point-of-view. The various possible scheduling objectives can be classified in two categories, as follows:

- Regular scheduling objectives
- Nonregular scheduling objectives

In this article, the difference between the two types of scheduling objectives is explained and illustrated on an example project shown in Fig. 5.25. The figure shows a project with five activities. Each project has a duration estimate, a cash flow (positive or negative) and a resource requirement as shown in the table to the right of the project network.

**Fig. 5.25** An example project network with activity durations, cash flows and resources




| | Duration | Cash | Resource |
|---|---|---|---|
| 1 | 3 days | -€20 | Paul |
| 2 | 2 days | -€20 | John |
| 3 | 7 days | -€10 | Paul |
| 4 | 3 days | -€50 | John |
| 5 | 2 days | €150 | John |

[15]This section is also available as an article entitled "Resource constrained project scheduling: Regular and nonregular scheduling objectives" published online at PM Knowledge Center.

### Regular Scheduling Objectives

Since the construction of a project schedule involves the presentation of an activity timetable, a schedule can be characterized by the start time of the project activities. Assume two project schedules S and S′, each characterized by their activity start times as follows:

- Schedule S: $s_1, s_2, \ldots, s_n$
- Schedule S′: $s'_1, s'_2, \ldots, s'_n$

with $n$ the number of activities in the project.

A formal definition of a regular scheduling objective RSO can be given as follows:

**Definition of regular scheduling objective**

A regular scheduling objective RSO is a function of the activity start times $s_1, s_2, \ldots, s_n$ such that

when

$$s_1 \leq s'_1, s_2 \leq s'_2, \ldots, s_n \leq s'_n$$

then

$$\text{RSO}(s_1, s_2, \ldots, s_n) \leq \text{RSO}(s'_1, s'_2, \ldots, s'_n) \text{ for a minimization objective}$$

or

$$\text{RSO}(s_1, s_2, \ldots, s_n) \geq \text{RSO}(s'_1, s'_2, \ldots, s'_n) \text{ for a maximization objective}$$

A nonregular scheduling objective is an objective for which this definition does not hold.

This definition implies that when two resource feasible schedules have been constructed such that each activity under the first schedule starts no later than the corresponding start time in the second schedule, then the first schedule is at least as good as the second schedule. Consequently, it will never be beneficial to delay an activity of a resource feasible schedule towards the end of the project.

Figure 5.26 illustrates the definition on a project schedule with a time minimization scheduling objective. The schedule has a project deadline of 12 time units, and has no resource conflicts (see section "*BS7: Critical Path/Chain*" on page 31). Minimizing the time of a project is a regular scheduling objective, and hence, it is never beneficial to delay an activity of a resource feasible schedule.

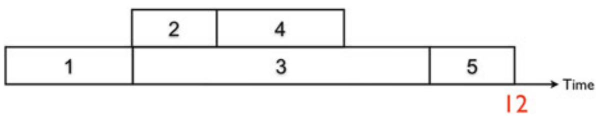



**Fig. 5.26** An example schedule S for the network of Fig. 5.25

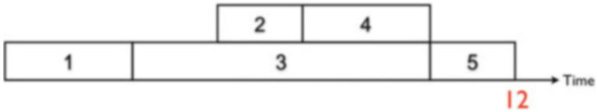



**Fig. 5.27** An example schedule S' for the network of Fig. 5.25

The definition of a regular scheduling objective can be illustrated on two project schedules shown in Figs. 5.26 and 5.27. The schedule S of Fig. 5.26 and the schedule S' of Fig. 5.27 can be characterized by the following activity start times:

- Schedule S: $s_1 = 0, s_2 = 3, s_3 = 3, s_4 = 5, s_5 = 10$
- Schedule S': $s'_1 = 0, s'_2 = 5, s'_3 = 3, s'_4 = 7, s'_5 = 10$

It is clear that all start times of schedule S are lower than or equal to the start times of schedule S'. Indeed, delaying activities 1, 3 or 5 would lead to an increase of the total project duration of 12. Delaying activities 2 and 4 would not lead to a project duration increase when they are delayed within their slack. Consequently, delaying activities in the schedule of Fig. 5.26 will never lead to an improvement of the scheduling objective, i.e. in a project duration reduction. This corresponds to the formal definition, i.e. TIME($s_1 = 0, s_2 = 3, s_3 = 3, s_4 = 5, s_5 = 10$) $= 12 \leq$ TIME($s'_1 = 0, s'_2 = 5, s'_3 = 3, s'_4 = 7, s'_5 = 10$) $= 12$.

Constructing resource feasible project schedules with regular scheduling objectives can be done by using priority rules and generation schemes, as discussed in section "*BS17: Priority Rule Based Scheduling*" on page 66.

### Non-regular Scheduling Objectives

A nonregular scheduling objective is an objective for which the formal definition above does not hold. Consequently, a given resource feasible schedule can be improved by delaying one or more activities towards the end. A typical nonregular scheduling objective is the maximization of the net present value of a project, where activities with positive cash flows are scheduled as soon as possible, while activities with negative cash flows are scheduled as late as possible (see section "*BS25: Net Present Value*" on page 96).

In the example project schedules of Figs. 5.26 and 5.27, it is clear that delaying activities 2 and 4 of schedule S leads to an improvement of the scheduling objective, since it increases the net present value. Consequently, the schedule S' has a higher net present value than the schedule S, and hence, the definition above does not hold, i.e. NPV($s_1 = 0, s_2 = 3, s_3 = 3, s_4 = 5, s_5 = 10$) is not larger than or equal to NPV($s'_1 = 0, s'_2 = 5, s'_3 = 3, s'_4 = 7, s'_5 = 10$).

Other nonregular scheduling objectives are the idle time minimization and the resource leveling objective, as discussed in sections "*BS26: Resource idle time*" on page 100 and "*BS27: Resource leveling*" on page 103.

Constructing resource feasible project schedules with nonregular scheduling objectives can be done by using priority rules and an iterative shifting algorithm, such as the Burgess and Killebrew algorithm, which will not be discussed in this book.

**Questions**

1. With regular objectives it is never beneficial to delay an activity in a resource feasible schedule.
   (a) True
   (b) False
2. With nonregular objectives it is never beneficial to delay an activity in a resource feasible schedule.
   (a) True
   (b) False
3. Which of the following scheduling objectives is not a nonregular scheduling objective?
   (a) Resource leveling
   (b) Minimize resource idle time
   (c) Minimize project lead time
   (d) Maximize project net present value
4. An earliest start schedule that is feasible with respect to the renewable resource constraints always yields the optimal solution.
   (a) True
   (b) False
5. Assume a project scheduling problem with a nonregular objective. Assume that two solution vectors exist $(s_1, s_2, s_3, s_4, s_5)$ and $(t_1, t_2, t_3, t_4, t_5)$ representing the start times of activities in two different schedules, with $s_1 < t_1, s_2 > t_2, s_3 > t_4, s_4 = t_4, s_5 < t_5$. Which solution vector contains the best solution?
   (a) $(s_1, s_2, s_3, s_4, s_5)$
   (b) $(t_1, t_2, t_3, t_4, t_5)$
   (c) This is impossible to decide with the information available.

## *BS24: Project Lead Time*[16]

Resource-constrained project scheduling involves the construction of an activity timetable, i.e. the determination of a start and finish time for each project activity,

[16]This section is also available as an article entitled "What is my scheduling objective? Minimizing the project lead time" published online at PM Knowledge Center.

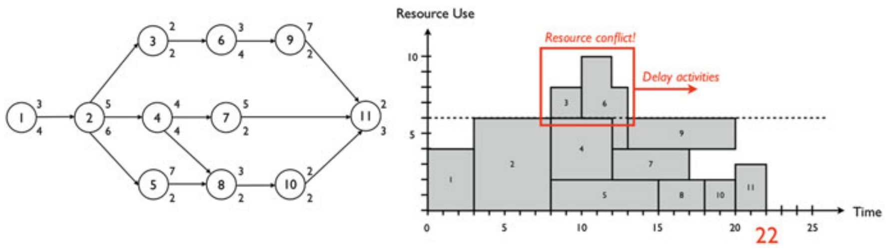



**Fig. 5.28** Project network with activity data and the resource profile of the earliest start schedule

respecting the precedence relations and the limited availability of the renewable resources, while optimizing a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). In this article, the scheduling objective is assumed to be the minimization of the total duration (or lead time) of the project, which is often a standard scheduling objective option in commercial software tools.

Figure 5.28 displays a project network with 11 activities and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand. It is assumed that the availability of the resource is restricted to six units for all periods of the project. The right part of the picture displays an earliest start schedule respecting the precedence relations of the project. Each activity is represented by a rectangle with the horizontal length the activity duration and the vertical height the renewable resource requirement/demand. The earliest start schedule shows a resource conflict between period 8 and 13 since the total use of the renewable resource by activities 3, 4, 5, 6 and 7 exceeds its limited availability of 6. In order to solve this resource conflict, activities need to be shifted further in time to time periods where resources are still available (see section "*BS8: Linking Resources*" on page 36).

## Scheduling Objective: Minimize the Time

Solving resource conflicts results in activity shifts in time in order to schedule them at periods where resources are still available. Activities 3 and 6 cause a resource conflict in Fig. 5.28, and must therefore be shifted such that the over-allocations disappear. A simple shift of these activities results in a resource feasible project schedule as shown in Fig. 5.29 with a total project duration of 26.

However, this schedule has been constructed without any scheduling objective to optimize, and hence, the shifts of activities 3 and 6 are rather random choices. In Fig. 5.30, activities 5 and 6 have been delayed, resulting in a different resource feasible schedule. In this schedule, time is indeed optimized, resulting in a resource feasible project schedule with the lowest possible project duration.

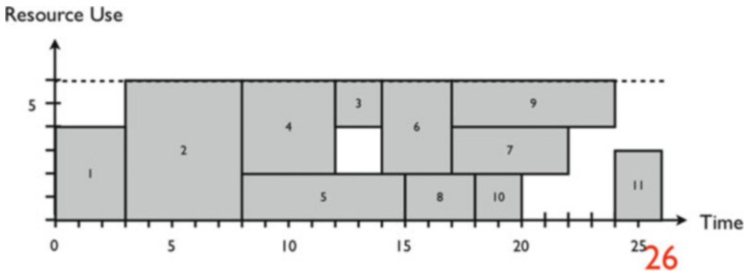



**Fig. 5.29** A resource feasible project schedule obtained by delaying activities 3 and 6

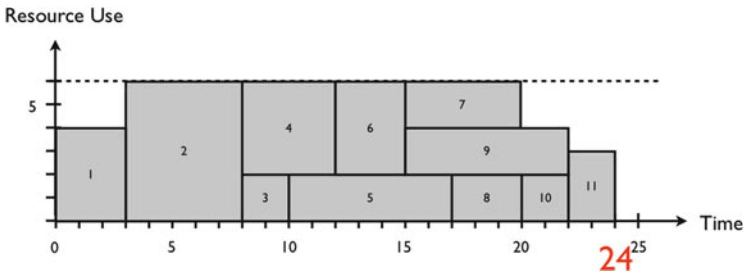



**Fig. 5.30** A resource feasible project schedule with the lowest possible total project duration

Note that the critical path in Fig. 5.28 is equal to the activity sequence 1-2-3-6-9-11 while the critical chain of Fig. 5.30 is equal to activity sequence 1-2-4-6-9-11 (see section "*BS7: Critical Path/Chain*" on page 31).

## Questions

1. An earliest start schedule that respects the precedence relations is always resource feasible.
   (a) True
   (b) False
2. In order to build a feasible project schedule, two types of constraints have to be taken into account, namely the precedence relations and the renewable resource availability.
   (a) True
   (b) False
3. The critical path and critical chain are the same when minimizing the project lead time.
   (a) True
   (b) False

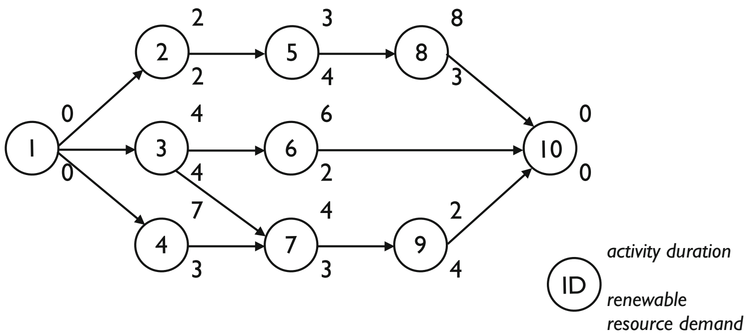



**Fig. 5.31** An example project network

4. Minimizing the project lead time is a regular objective.
   (a) True
   (b) False

5. The minimum project duration is always equal to the critical path when the resource availability is equal to the resource requirements of the activity with the highest resource demand.
   (a) True
   (b) False

6. Consider the project in Fig. 5.31 on page 96. The number above each node represents the activity duration and the number below each node is used to refer to the renewable resource demand. Assume that the availability of the renewable resource is equal to 8 units. Which of the following statements is correct?
   (a) The critical path is equal to 13 and the minimal project duration is 15.
   (b) The critical path is equal to 13 and the minimal project duration is 13.
   (c) The critical path is equal to 11 and the minimal project duration is 15.
   (d) The critical path is equal to 11 and the minimal project duration is 13.

## *BS25: Net Present Value*[17]

Resource constrained project scheduling involves the construction of an activity timetable, i.e. the determination of a start and finish time for each project activity, respecting the precedence relations and the limited availability of the renewable

[17]This section is also available as an article entitled "What is my scheduling objective? Maximizing the net present value" published online at PM Knowledge Center.

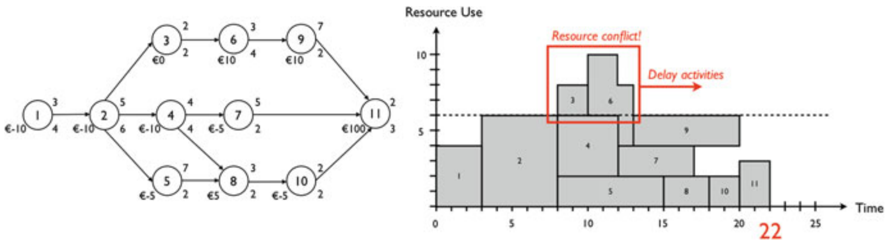



**Fig. 5.32** Project network with activity data and the resource profile of the earliest start schedule

resources, while optimizing a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). In this article, the scheduling objective is assumed to be the maximization of the net present value of the project.

Figure 5.32 displays a project network with 11 activities and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity while the numbers below the node are used to refer to the net cash flows (payments receipts) and resource demand. Remark that these cash flows are represented as a terminal value, realized at the end of each activity. It is assumed that the availability of the resource is restricted to six units for all periods of the project. The right part of the picture displays an earliest start schedule respecting the precedence relations of the project. Each activity is represented by a rectangle with the horizontal length the activity duration and the vertical height the renewable resource requirement/demand. The earliest start schedule shows a resource conflict between period 8 and 13 since the total use of the renewable resource by activities 3, 4, 5, 6 and 7 exceeds its limited availability of 6. In order to solve this resource conflict, activities need to be shifted further in time to time periods where resources are still available (see section "*BS8: Linking Resources*" on page 36).

## The Net Present Value

Any project involves large financial implications. Costs (cash outflows) for raw material, human resources and material are connected with the execution of each activity. Moreover, a large scaled project involves progress payments at previously determined time windows in order to guarantee a constant cash inflow and to secure the heavy cash outflows during the execution of the project. The complex interaction between these two cash flows leads to interesting insights during the scheduling of the project.

When one takes these financial aspects into account, the objective should be to maximize the net present value (*npv*) of the project. This well-known concept is based on a simple principle: Accept the project when the $npv > 0$ and reject it when the $npv < 0$. The value of each cash flow (either inflow or outflow) is compounded towards the beginning of the project by means of the discount rate $\alpha$ as follows:

$$npv = \sum_i c_i * e^{-\alpha * f_i}$$

where $\sum_i$ is used to take the sum over all project activities, $c_i$ denotes the cash flow linked with the execution of activity $i$, $f_i$ the finish time of activity $i$ in the baseline schedule and $n$ the total number of activities in the project.

### Scheduling Objective: Maximize the Net Present Value

Once the project has been accepted (i.e. $npv > 0$), the accept/reject principle can be translated into an as soon as possible/as late as possible mechanism. Indeed, during the project scheduling phase, all activities have to be scheduled in time such that no precedence relations and resource constraints are violated. Time was, and still is, an important objective to that purpose, but financial aspects must be incorporated too. This translation is straightforward and lies in the very heart of the net present value: Cash inflows should be scheduled as soon as possible while cash outflows should be scheduled as late as possible. Despite this simple principle, the construction of such a resource profile is a complex task.

Solving resource conflicts results in activity shifts in time in order to schedule them at periods where resources are still available. The choice of which activities have to be shifted depends on the scheduling objective. In section "*BS24: Project Lead Time*" on page 93, it has been shown that shifting activities 5 and 6 results in a resource feasible schedule with a minimal project duration (i.e. 24 time periods), as displayed in Fig. 5.30.

However, a shift of activity 4 instead of 6 can result in a better schedule when the scheduling objective is to maximize the net present value. Indeed, activity 4 has a negative cash flow and needs to be scheduled as late as possible, while activity 6 has a positive cash flow, which gives the scheduler an incentive to schedule it as soon as possible. Although the minimization of the total project duration is not the scheduling objective, the presence of a big lump sum payment at the end of the project (activity 11) gives the scheduler the objective to finish the project as soon as possible, and hence, both time and *npv* are the scheduling objectives, resulting in the baseline schedule of Fig. 5.33.

It is up to the reader to calculate the total net present value that is equal to € 13.57 for the baseline schedule of Fig. 5.30 and € 15.23 for the schedule shown in Fig. 5.33, under the assumption that $\alpha = 5\,\%$. Note that the critical path in Fig. 5.32 is equal to the activity sequence 1-2-3-6-9-11 while the critical chain of Fig. 5.30 and Fig. 5.33 are equal to activity sequences 1-2-4-6-9-11 and 1-2-3-6-4-7-11, respectively. This example illustrates that the critical chain is not unique but depends on the scheduling objective, and hence, on the choice of delaying activities to solve resource conflicts (see section "*BS7: Critical Path/Chain*" on page 31).

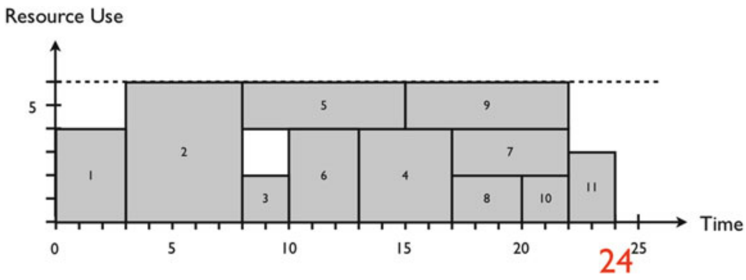



**Fig. 5.33** A resource feasible project schedule with a maximum net present value

## Questions

1. A project should have a net present value lower than 0, otherwise it should not be executed.
   (a) True
   (b) False
2. If the scheduling objective is the maximization of the net present value, activities with a negative cash flow should be scheduled as soon as possible.
   (a) True
   (b) False
3. If a big lump sum payment is present at the end of the project, the scheduler will try to finish the project as soon as possible.
   (a) True
   (b) False
4. Consider the project network in Fig. 5.34, with the activity duration in the top right corner above each node and the renewable resource demand in the bottom right corner below each node. The activity cash flows are displayed below each node and are assumed to be incurred at the finish of each activity. What is the net present value of the schedule shown on top of the figure? (discount rate = 0.05 per period)
   (a) € 47.01
   (b) € 75.34
   (c) € 81.49
   (d) € 124.61
5. Which of the following changes will increase the net present value in the schedule of Fig. 5.34?
   (a) Activity 1 and 2 are swapped. The start of activity 3 is postponed to the finish time of activity 1. All the other activities keep their start times.

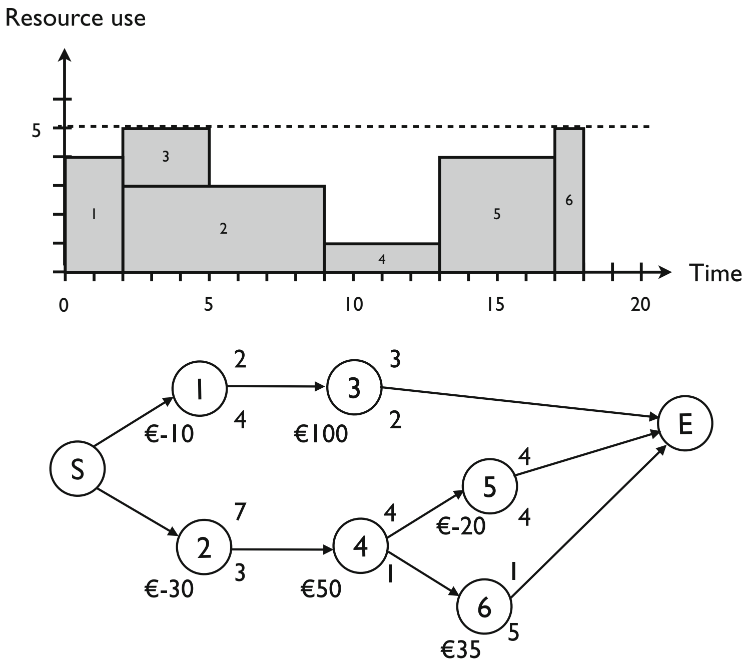



**Fig. 5.34** An example project network and resource feasible schedule

(b) Activity 3 starts at time unit 9.
(c) Activity 1 and 2 are swapped. The start of activity 3 is postponed to the finish time of activity 1. Activity 4 is scheduled at the finish time of activity 2. Activities 5 and 6 start 1 time unit earlier.
(d) Activities 5 and 6 are swapped (start time of activity 5 = 14 and start time of activity 6 = 13).

## *BS26: Resource Idle Time*[18]

Resource-constrained project scheduling involves the construction of an activity timetable, i.e. the determination of a start and finish time for each project activity, respecting the precedence relations and the limited availability of the renewable resources, while optimizing a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). In this article, the scheduling

[18]This section is also available as an article entitled "What is my scheduling objective? Minimizing the resource idle time" published online at PM Knowledge Center.

objective is assumed to be the minimization of the resource idle time of one or more bottleneck resources of the project.

Figure 4.7 displays a project network with 11 activities and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand. It is assumed that the availability of the resource is restricted to six units for all periods of the project. Moreover, it is also assumed that activities 3, 5 and 6 need an extra resource, such as a dry dock, in which the activities need to be executed (parallel execution is possible since the dry dock is large enough).

### Work Continuity

The term "work continuity constraints" is used to refer to the need for uninterrupted usage of resources used by similar activities, in order to enable the timely movement of resources (e.g. crews) from one activity to the other and to avoid idle time or loss of efficiency. These activities that make use of the same resource are grouped in so-called activity group and the objective is to minimize the time-span between the start and the finish of all activities in the group.

- Repetitive activities in the construction industry: Construction projects are often characterized by repeating activities that have to be performed from unit to unit. Examples are the construction of highway projects, pipeline constructions and high-rise buildings in which the crews perform the work in a sequence and move from one unit of the project to the next. In section "*BS6: Resource Types*" on page 29, spatial resources are mentioned as a renewable resource type that are required by a group of activities, rather than by a single activity. The spatial resource is occupied from the first moment an activity from the group starts until the finish of all activities from that group. Examples are dry docks in a ship yard or a freezing machine in the Westerscheldetunnel (see a case study in the book "Dynamic Scheduling" (www.or-as.be/books) about work continuity constraints in the Westerscheldetunnel).
- Outsourcing activities/hired material: Project scheduling problems where a set of activities has been outsourced or that rely on external resources (subcontracting, consultants, etc.) need to be scheduled with work continuity constraints. This means that the set of activities can be divided into activity groups that have to be scheduled within the smallest possible time span in order to minimize the total cost of outsourcing.
- Program scheduling with different stakeholders: In programs that consist of different projects for different stakeholders, each subproject can be seen as an individual "activity group" where work continuity can be of importance. Consequently, it is beneficial to schedule the activities within an activity group within the smallest possible time span within the precedence and resource constraints of the complete program. In doing so, the project duration towards

each stakeholder is minimized and the satisfaction of the different stakeholders might be increased.

- Project with time critical sub-projects: In projects where only a subpart is time critical, work continuity constraints can be important for scheduling the time critical subnetwork.

### Scheduling Objective: Minimize the Resource Idle Time

In Fig. 5.30, the baseline resource feasible schedule that minimizes the total project duration is shown, as discussed in section "*BS24: Project Lead Time*" on page 93. In Fig. 5.33, a similar resource feasible schedule is displayed, as a result of the maximization of the net present value (see section "*BS25: Net Present Value*" on page 96).

It is obvious that the baseline schedule of Fig. 5.33 is better than the schedule of Fig. 5.30 since activities 3, 5 and 6, which all belong to the same activity group, are scheduled closer to each other. The time span for the use of the dry dock is equal to 8 time periods, which is better than the 9 time periods in the schedule of Fig. 5.30. Consequently, although none of the schedules have real idle time in their use of the dry dock, work continuity constraints imply that the time span of the use of these resources is minimized, which also entails the avoidance of idle time, if possible.

### Questions

1. The work continuity constraint is introduced in project scheduling to …
   (a) minimize the breakdowns of a machine in a project.
   (b) minimize the total duration of a project.
   (c) minimize the idleness of resources and the inefficient use of resources.
   (d) minimize the tardiness of a project.
2. In work continuity constraints, activity groups are …
   (a) groups of activities that have direct predecessors and for which the net present value has to be maximized.
   (b) groups of activities that make use of the same resources and for which the net present value has to be maximized.
   (c) groups of activities that have direct predecessors and for which the time span between the start and finish times of all activities has to be minimized.
   (d) groups of activities that make use of the same resources and for which the time span between the start and finish times of all activities has to be minimized.

3. Activities in an activity group are always directly connected by a precedence constraint.
   (a) True
   (b) False
4. A spatial resource is a good example of a resource that is required by an activity group.
   (a) True
   (b) False

## *BS27: Resource Leveling*[19]

Resource-constrained project scheduling involves the construction of an activity timetable, i.e. the determination of a start and finish time for each project activity, respecting the precedence relations and the limited availability of the renewable resources, while optimizing a predefined scheduling objective (see section "*BS22: What Is My Scheduling Objective?*" on page 88). In this article, the scheduling objective is assumed to be the leveling of the use of resources along the project duration.

Figure 5.35 displays a project network with nine activities (plus a start and an end dummy node) and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the resource demand. It is assumed that the availability of the resource is restricted to ten units for all periods of the project.

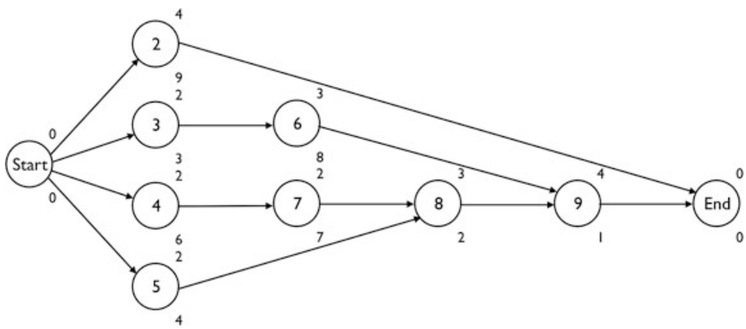



**Fig. 5.35** Project network with activity data

[19]This section is also available as an article entitled "What is my scheduling objective? Leveling the use of resources" published online at PM Knowledge Center.

**Scheduling Objective: Resource Leveling**

In Fig. 5.36, an earliest start schedule is shown with a lot of over-allocations, with peaks up to 25 units needed during periods 3 and 4. Solving resource conflicts results in activity shifts in time in order to schedule them at periods where resources are still available. When the scheduling objective is to level or balance the use of the renewable resources, the final schedule should have a total resource profile where upward or downward jumps are minimized and the use of resources is as balanced as possible.

Figure 5.37 displays a resource profile where activities are scheduled such that the use of resources is perfectly leveled, within the maximum availability of 10 units per period. Of course, this example displays an ideal situation where the use of resources is equal over all periods of the project, and hence, the schedule shows no upward or downward jumps over time. Moreover, the project duration is equal to the critical path length, which is not always possible. Indeed, the critical path in Fig. 5.35 is equal to the activity sequence 4-7-8-9, which is equal to the

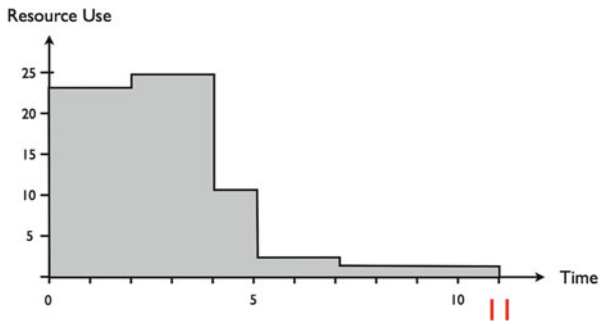



**Fig. 5.36** The earliest start schedule of the project network of Fig. 5.35

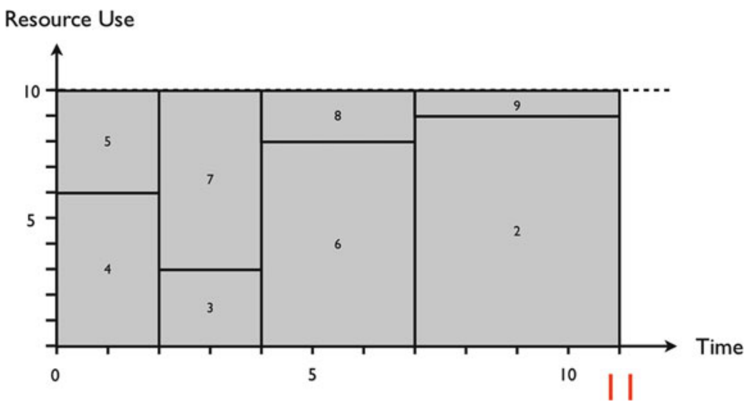



**Fig. 5.37** A resource leveled schedule of the project network of Fig. 5.35

critical chain. Obviously, in many cases, the critical chain will be longer than the critical path length when taking renewable resources into account (see section "*BS7: Critical Path/Chain*" on page 31).

## Questions

1. If resource leveling is taken as scheduling objective, the final schedule has a resource profile where the use of resources is as balanced as possible.
   (a) True
   (b) False
2. The resource leveling scheduling objective often is the standard scheduling objective option in commercial software tools.
   (a) True
   (b) False
3. In the project displayed in Fig. 5.38, the earliest start schedule is the optimal solution for both a lead time minimization and a resource leveling objective.
   (a) True
   (b) False
4. Which of the following statements with respect to resource leveling as a schedule objective is not correct?
   (a) Taking resource leveling as a scheduling objective, minimizing the upward and downward jumps in the schedule could be use to validate the final result.
   (b) An earliest start schedule with no over-allocations is always optimal if resource leveling is the schedule objective.
   (c) In many cases, the critical chain will be longer than the critical path length when taking renewable resources into account.
   (d) If the use of the resources is equal over all periods of the project, the schedule is optimal if resource leveling is the schedule objective.

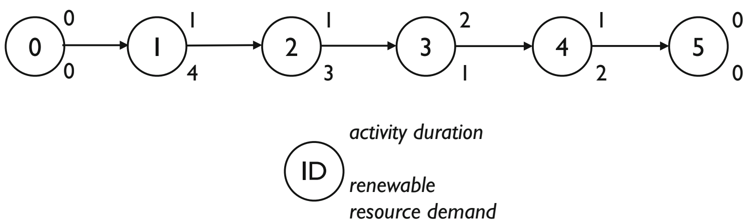



**Fig. 5.38** An example project network

# Part II
# Schedule Risk Analysis

# Chapter 6
# Preface

## RA1: An Introduction to Risk Analysis[1]

Since uncertainty is what typifies projects in progress, risk management is key to the success of a project. A technique known as Schedule Risk Analysis connects the risk information of project activities to the baseline schedule and provides sensitivity information of individual project activities as a way to assess the potential impact of uncertainty on the final project duration. The protection of a project's deadline using a technique called Critical Chain/Buffer Management assures that delays in project activities are captured by buffers that are cleverly inserted into the project baseline schedule.

The analysis of a project schedule's risk and its management through the use of detailed risk information and the incorporation of buffers is key to the success of a project. However, risk management is not a goal in itself, but rather it serves as a tool to improve and steer the project control phase (see article "*PC1: An introduction to project control*" on page 197). Indeed, the combination of the baseline schedule (see article "*BS1: An introduction to baseline scheduling*" on page 9) and the risk information should be interpreted and used as a point-of-reference to reveal threats and opportunities during the project progress.

The schedule risk analysis topics of this book have been classified in the following categories:

- Schedule risk analysis
- Buffer management

[1]This section is also available as an article entitled "Dynamic scheduling: An introduction to risk management" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_6

For an overview of the three dynamic scheduling dimensions, see Fig. 1.1 on page 2 or article "*Welcome to PM Knowledge Center*" on page 1. The different categories are briefly explained below.

## *Schedule Risk Analysis*

Schedule Risk Analysis is a technique that allows you to refine the traditional Critical Path Method (CPM) to degrees of criticality and risk. It connects the risk information of project activities to the baseline schedule and provides sensitivity information of individual project activities as a way to assess the potential impact of uncertainty on the final project duration and cost. In doing so, it gives the project manager an idea on how dangerous/sensitive/risky an activity is for the project objective.

## *Buffer Management*

The Critical Chain/Buffer Management approach assumes the construction of a resource feasible schedule as described in article "*BS1: An introduction to baseline scheduling*" on page 9, but incorporates a certain degree of flexibility in the activity start times in order to easily monitor schedule deviations and quickly respond by taking corrective actions to keep the whole project on schedule. It is based on the novel "Critical Chain" by E. Goldratt using his Theory of Constraints.

# Chapter 7
# Schedule Risk Analysis

## RA2: Simulating Project Progress[1]

Monte Carlo simulations can be used in dynamic project scheduling to measure the sensitivity of project activities as described in article "*RA5: Schedule sensitivity*" on page 123 or to evaluate the accuracy of forecasting methods used in Earned Value Management (see article "*PC13: Forecast accuracy*" on page 236). In this article, the underlying principle used during these simulations is briefly explained and illustrated on a small example. In the example, the baseline duration of an activity must be replaced by a number generated from a predefined distribution function. For more information on the use of distribution functions, see article "*RA4: Activity distributions*" on page 119. A simple simulation approach consisting of eight special simulation scenarios is proposed in article "*RA3: CPM schedule control*" on page 115.

A Monte Carlo simulation run generates a duration for each project activity given its predefined uncertainty profile, as follows:

- Generate a random number from the interval [0,1[
- Generate a number using a cumulative distribution function
- Add the new number to the baseline schedule

Figure 7.1 displays the general three-step approach of Monte Carlo simulation as an illustration to replace the single point duration estimate of activity 2 by a randomly generated number drawn from a statistical distribution.

[1]This section is also available as an article entitled "Monte Carlo simulations: How to imitate a project's progress?" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_7

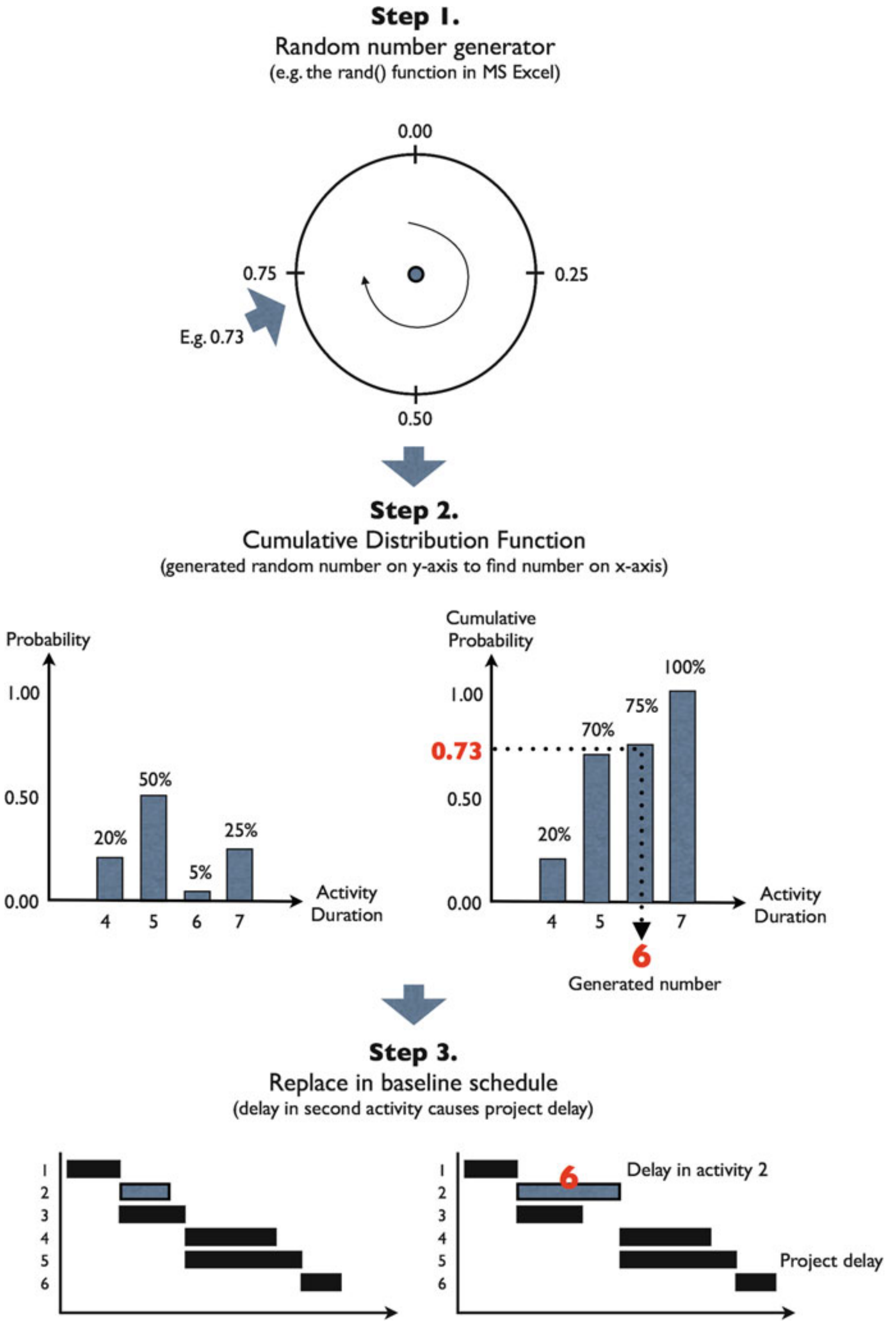



**Fig. 7.1** The Monte Carlo simulation approach in project scheduling

## *Step 1: Random Number Generation*

A random number generator is a computerized device designed to generate a sequence of numbers that lack any pattern, i.e. appear random. The use of computational algorithms to produce long sequences of apparently random results, which are in fact completely determined by a shorter initial value known as a seed value, are called pseudorandom number generators. The use and quality of various pseudorandom number generators is a widely studied problem in the computer science literature and is outside the scope of this article.

In Fig. 7.1, this first step is represented by a Monte Carlo wheel labeled with a number between 0 (inclusive) and 1 (exclusive), which enables the user to generate random numbers from that wheel with an equal probability. In MS Excel, this random wheel function is called RAND() for which each turn on the wheel (F9 in MS Excel) provides the user with a number from the [0,1[ interval.

## *Step 2: Cumulative Distribution Function*

The random number is used to generate the number from a predefined distribution. The left part in the second step of Fig. 7.1 displays a discrete distribution function with four possible values (4, 5, 6 or 7) and their corresponding probabilities. The cumulative distribution function on the right of Fig. 7.1 can be used to generate numbers from that distribution function by generating a sequence of random numbers between 0 and 1.

Suppose that the random number is equal to 0.73, then this will result in a generated number 6 from the probability distribution. Indeed, since $0.73 > 0.70$ and $0.73 < 0.75$, the number corresponds to the number 6 on the x-axis. Multiple generations will lead to different random numbers on the y-axis, but the generated numbers on the x-axis will have a probability of being generated equal to the probability function of the figure.

## *Step 3: Add the New Number to the Baseline Schedule*

The generated numbers will be used to replace the baseline duration with the newly generated value to simulate fictitious project progress, as described in article "*RA5: Schedule sensitivity*" on page 123. In doing so, the activities will experience a faster execution or a delay compared to their baseline duration, resulting in a shift of the successor activities and a decrease/increase in the total project duration. In the third step of Fig. 7.1, activity 2 experiences a delay resulting in an overall project delay.

Although this Monte Carlo simulation approach is illustrated on the activity duration estimates, estimates of other project parameters can also be subject to variability. Consequently, this approach can also be repeated for other parameters than the activity duration, such as the activity cost, the time-lag between project activities (see article "*BS3: Precedence relations*" on page 16) and many more.

## Questions

1. Random number generation is usually based on a pseudorandom number generator which generates numbers from the interval ...
   (a) [0, 1].
   (b) [0, 1[.
   (c) ]0, 1[.
2. Random number generators can generate fully random sequences.
   (a) Yes, if the algorithm is not inherently flawed.
   (b) Yes, if the algorithm is first provided with a random seed such as the clock time of a computer.
   (c) No, algorithms can never produce truly random numbers.
3. Random numbers in the range [0,1[ are used together with a ... in order to generate durations for activities.
   (a) probability density function
   (b) cumulative distribution function
   (c) neither of the two above
4. When performing a Monte Carlo analysis of the durations, only the critical path has to be simulated since this will always determine the duration of the project.
   (a) True
   (b) False
5. Monte Carlo simulation can only be used for discrete distributions.
   (a) True
   (b) False
6. Assume the probability distribution presented in Fig. 7.2 is used for simulating an activity duration. If the random number generator presents the number 0.379403, then which duration will be selected?
   (a) 4
   (b) 5
   (c) 6
   (d) 7
7. Assume the last number from the random number generator was equal to 0.11134 using the distribution function of Fig. 7.2, resulting in an activity duration equal

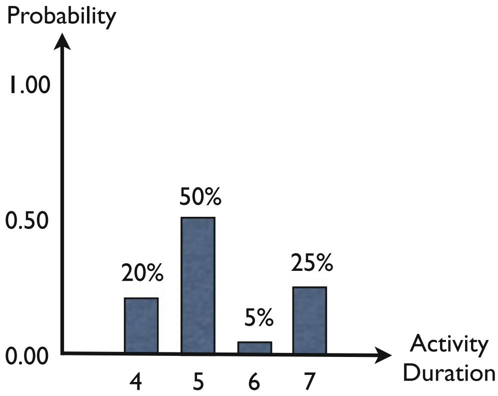



**Fig. 7.2** An example distribution

to 4. What is the probability that the duration of the next simulated number is also equal to 4?

(a) 0 %
(b) 2.2268 %
(c) 20 %
(d) 8.8866 %

## RA3: CPM Schedule Control[2]

Monte Carlo simulations can be used for various purposes to analyze the behaviour of projects in (fictitious) progress. It can be used to measure the sensitivity of project activities as described in article "*RA5: Schedule sensitivity*" on page 123 or to evaluate the accuracy of forecasting methods used in Earned Value Management (see article "*PC13: Forecast accuracy*" on page 236). In this article, a simple yet effective Monte Carlo simulation approach is proposed consisting of nine simulation scenarios that can be used to link critical path schedules to project control information.

A Monte Carlo simulation run generates a duration for each project activity given its predefined uncertainty profile. This article will not give much information about the underlying principle of distribution functions that are used in simulation studies to define activity uncertainty. The reader is referred to the article "*RA4: Activity distributions*" on page 119 for more information. Instead, it will focus on the general concept of the simulation scenarios and the interpretation of each of the nine scenarios.

[2]This section is also available as an article entitled "Monte Carlo simulations: Linking critical path schedules to project control" published online at PM Knowledge Center.

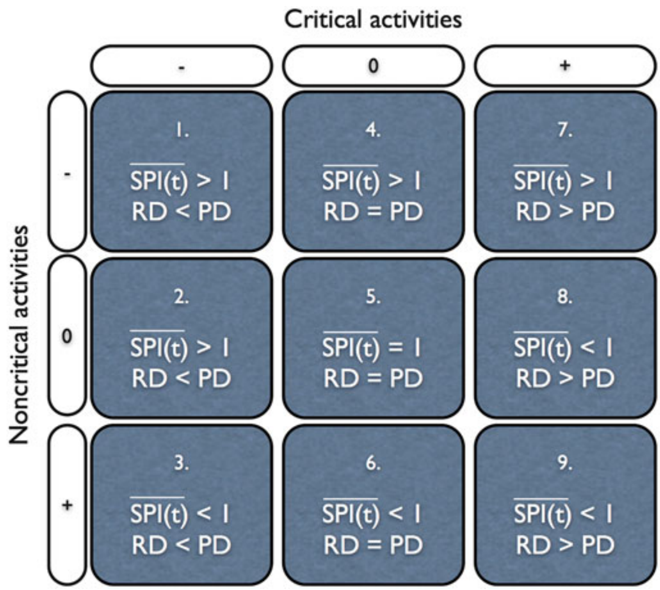



**Fig. 7.3** The nine scenarios of the simulation approach

## *Simulation Approach*

Figure 7.3 graphically summarizes the nine simulation scenarios of the simulation approach. The nine scenarios are constructed in such a way to make a connection between the critical path schedule, the project performance measurement information along the project progress and the final project status. This connection and all details of Fig. 7.3 are explained along the following lines.

Critical path schedule: Uncertainty in the activity durations can be defined differently for critical versus noncritical activities (see article "*BS10: Activity slack*" on page 43). Both classes of activities can have simulated durations that are shorter, equal to or longer than their planned duration. In Fig. 7.3, this is displayed as follows:

- $-$: activity duration shorter than planned
- 0: activity on time
- $+$: activity duration longer than planned

Project performance (during simulation): Each simulation run imitates fictitious project progress in which periodic project performance can be measured using Earned Value Management techniques. In the simulation scenarios, the time performance is measured by the Schedule Performance Index SPI(t) (see article "*PC9: Time performance*" on page 220) at periodic time intervals. This metric serves as a periodic warning signal that gives a project based view on the project's performance at the current time. The average of all SPI(t) values (displayed as $\overline{\text{SPI(t)}}$) between the start and finish of the project is displayed in the body of Fig. 7.3 and gives an indication of the average signal reported by the EVM metric and has the following meaning:

- $\overline{\text{SPI(t)}} = 1$: average 'on time' signal
- $\overline{\text{SPI(t)}} > 1$: average positive signal (ahead of schedule)
- $\overline{\text{SPI(t)}} < 1$: average negative signal (schedule delay)

Final project status (after simulation): After each simulation run, the real project duration (RD) can be different from the planned project duration (PD), leading to project under- or overruns.

- RD = PD: project on time
- RD > PD: late project
- RD < PD: early project

## *Interpretation*

The nine simulation scenarios of Fig. 7.3 can be classified into three categories, each having a different meaning and purpose. The interpretation of the nine scenarios can be summarized as follows:

- True scenarios: Scenarios 1 and 2 report an average project 'ahead of schedule' progress where the project finishes earlier than planned. Scenarios 8 and 9 report an average 'project delay' progress and the project finishes later than planned. Scenario 5 reports an 'on-time' progress where the project finishes exactly on time. Consequently, these five scenarios report on average a true situation.
- Misleading scenarios: Scenario 4 reports an average project 'ahead of schedule' progress but the project finishes exactly on time. Likewise, scenario 6 reports an average 'project delay' progress but the project finishes exactly on time. Consequently, these two scenarios report on average a schedule deviation that is not true, and hence, they are called misleading simulation scenarios.
- False scenarios: Scenario 3 reports an average 'project delay' progress but the opposite is true: the project finishes earlier than planned. Scenario 7 reports an average project 'ahead of schedule' progress but the opposite is true: the project finishes later than planned. Consequently, these two scenarios report a false performance signal, and hence, they are called false simulation scenarios.

## Questions

1. The Schedule Performance Index SPI(t) is a metric that measures the …
   (a) cost performance.
   (b) time performance.
2. When the Schedule Performance Index SPI(t) metric is bigger than 0, the project is …
   (a) behind schedule.
   (b) on schedule.
   (c) ahead of schedule.
   (d) None of the previous answers is always correct.
3. When the Schedule Performance Index SPI(t) metric is bigger than 1 the project is …
   (a) behind schedule.
   (b) on schedule.
   (c) ahead of schedule.
   (d) None of the above.
4. Consider the simulation matrix in Fig. 7.3 on page 116. In scenario 4, the project will finish …
   (a) too soon.
   (b) on time.
   (c) too late.
5. In which of the scenarios of the simulation matrix in Fig. 7.3 are the EVM metrics fully reliable?
   (a) 1
   (b) 3
   (c) 6
6. Consider the simulation matrix in Fig. 7.3. In which scenarios can the SPI(t) metric result in the most inadequate actions by the project manager?
   (a) 7
   (b) 1
   (c) 4
   (d) 6

## RA4: Activity Distributions[3]

Risk management requires analytical skills and basic knowledge of statistics, which is often perceived as mathematically complex and sometimes theoretical and far from practice. However, a basic understanding of probability and distribution functions allows the project manager to better estimate the effects of unexpected events on the project outcome. The use of single point estimates for the project data, such as activity durations and costs or the value of the time-lags between project activities (see article "*BS3: Precedence relations*" on page 16), often leads to unrealistic project estimates due to the inherent uncertainty that typifies these projects. Therefore, the use of statistical distributions is crucial for a thorough and realistic analysis of the project as a preparation of its future progress that will be characterized by changes compared to the original point estimates.

- Single point estimates (= no risk): The activity entails no risk and the duration is a single point estimate (i.e. the estimate used in the baseline schedule).
- Interval estimates (= risk): Using statistical distributions or making use of simple triangular distribution functions.

In the next paragraphs, the use of interval estimates is discussed.

### *Use Statistics…*

Defining uncertainty on project parameters using statistics requires a profound knowledge of the formulas and characteristics of statistical distribution functions. Once the parameters of these functions are known, one can easily transform any distribution function into a cumulative distribution function (CDF), which allows the generation of a random number from this function.

The use of cumulative distribution functions to generate numbers is described in article "*RA2: Simulating project progress*" on page 111. Consider, as an example, the use of an exponential distribution. The cumulative distribution function of a random variable X that follows an exponential distribution can be given by

$$P(X \leq x) = 1 - e^{-\lambda x}$$

with $1/\lambda$ the mean of the exponential distribution.

When $u$ is used as a parameter to denote the cumulative probability $P(X \leq x)$ that obviously lies between 0 and 1, one can have

$$u = 1 - e^{-\lambda x} \rightarrow x = -1/\lambda * ln(1 - u)$$

[3]This section is also available as an article entitled "Project risk: Statistical distributions or single point estimates?" published online at PM Knowledge Center.

Consequently, since $u$ lies between 0 (inclusive) and 1 (exclusive), it can be replaced by a random number generated from the interval [0,1[. This can be obtained by a random number generator, e.g. the random wheel function RAND() that is available in MS Excel. This leads to a randomly generated number $x$ from an exponential distribution with an average equal to $1/\lambda$.

## *...or Make It Simple*

The realism of using statistical distributions in practical settings is often questionable due to uniqueness of the project or lack of data about the specific probability distributions. However, the method can be used in research environments where the influence of various project parameters on the project outcome is measured under different scenarios by varying the parameters of well-known statistical probability distributions.

In practice, risk is often defined in a very subjective and vague way and can therefore be modeled through a degree of skewness as a measure of the asymmetry of the probability distribution of a real-valued random variable. The skewness approach can be easily used with simple three-point estimates $(a, m, b)$ with $a < m < b$. Three-point estimates are widely used in project management for modeling the duration of project activities, as illustrated in article "*BS15: PERT*" on page 59. Three-point estimates can be used in the triangular distributions as shown in Fig. 7.4 to express risk as follows:

- Skewed to the left: The activity is subject to risk within a certain range, where $m - a > b - m$.
- Symmetric: The activity is subject to risk within a certain range, with worst case scenario $a$ and best case scenario $b$ symmetric above and below the value $m$, i.e. $m - a = b - m$.
- Skewed to the right: The activity is subject to risk within a certain range, where $m - a < b - m$.

A basic knowledge about the statistical terminology and the willingness to rely on easy-to-use software tools like MS Excel or graphical supported risk distribution

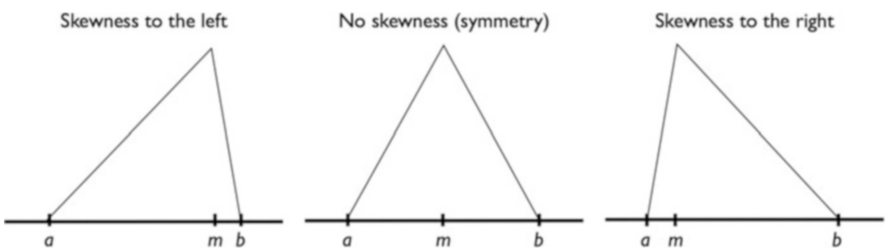



**Fig. 7.4** Illustration of triangular distributions with and without skewness

tools allow the project manager to easily set up a schedule risk analysis. The use of basic three-point estimates for risk as an easy approximate alternative for the complex statistical distributions makes schedule risk analysis understandable to a broad audience. Obviously, more advanced distribution shapes are possible with more than three-point estimates as in the triangular distribution functions.

## *Relevance*

Generally, the use of distributions instead of single point estimates in dynamic project scheduling replaces the deterministic nature of project parameters by interval estimates to have a more accurate and realistic estimate of the project outcome. As an example, the critical path method determines the longest path in a project network based on single point estimates for the activity durations (see article "*BS10: Activity slack*" on page 43), which results in a total project duration as a single value. When the activity duration estimates are replaced by interval estimates using probability distributions, the total project duration lies between two extreme values and consequently, the probability that the project will finish on or before a certain time from that interval can be calculated. This is illustrated in Fig. 7.5.

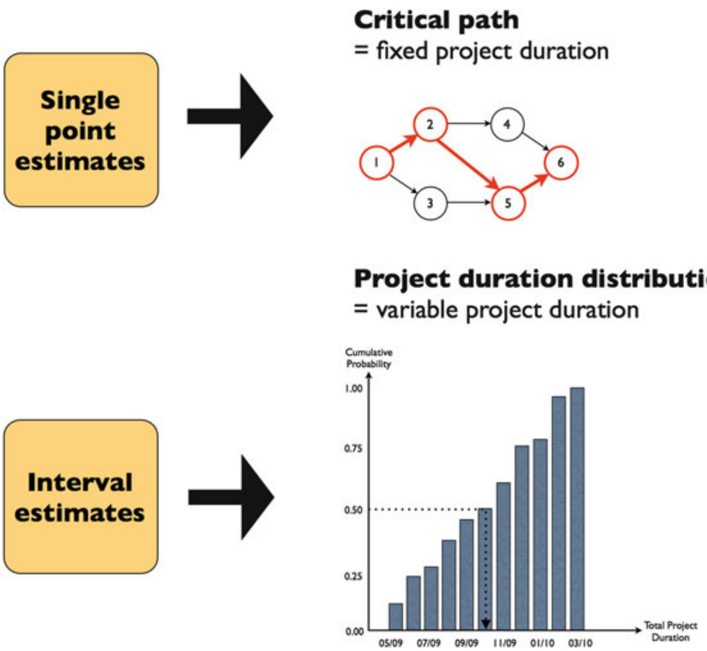



**Fig. 7.5** A fixed and variable project duration as a result of single point or interval duration estimates

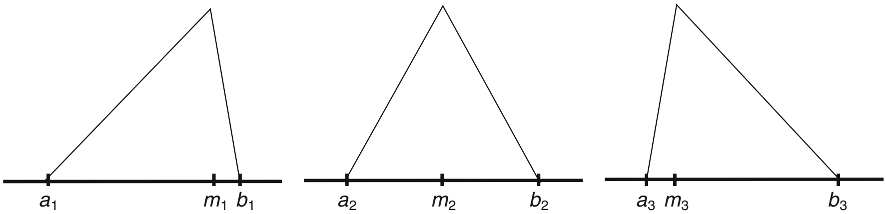



**Fig. 7.6** Three example triangular distributions

Obviously, the usefulness of Monte Carlo simulation in dynamic project scheduling goes further than determining a variable project duration. Monte Carlo simulations can be useful to measure the sensitivity of project activities and to validate the impact of changes on the project outcome (see article "*RA5: Schedule sensitivity*" on page 123) or to guide the project manager in the selection of the best performing forecasting technique for a project in progress (see articles "*PC13: Forecast accuracy*" on page 236 and "*RA3: CPM schedule control*" on page 115).

## Questions

1. Assume three triangular distributions using simple three-point estimates as shown in Fig. 7.6. Which of the three distributions is skewed to the left?
   (a) Distribution 1
   (b) Distribution 2
   (c) Distribution 3
2. Based on the triangular distributions in Fig. 7.6, which of the three distributions has the highest risk, expressed as a variance? Assume that $m_1 > m_2 > m_3$, $(b_1 - a_1) = (b_2 - a_2) = (b_3 - a_3)$ and $b_1 - m_1 = m_3 - a_3$.
   (a) Distribution 1
   (b) Distribution 2
   (c) Distribution 3
   (d) Distribution 1 and 2
   (e) Distribution 1 and 3
   (f) Distribution 2 and 3
3. By using interval estimates rather than point estimates, a new deterministic critical path will be identified for the project.
   (a) True
   (b) False

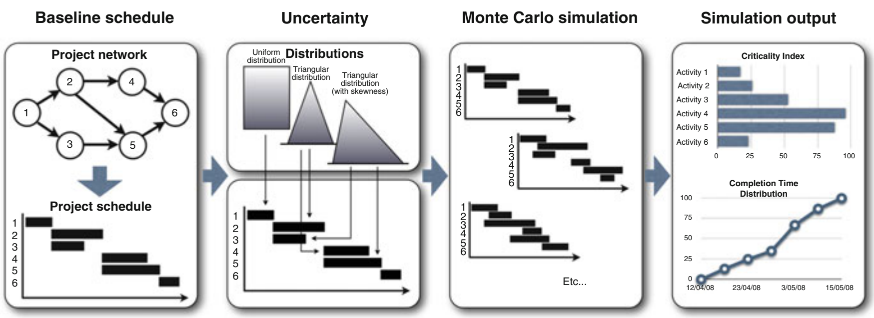



**Fig. 7.7** Four steps of schedule risk analysis

4. When triangular distributions are used in the simulation approach, the outcome of the analysis will be identical to the outcome of the PERT technique, since the latter also uses the triangular distribution to describe activity duration.

   (a) True
   (b) False

## RA5: Schedule Sensitivity[4]

Schedule Risk Analysis (SRA) is a simple yet effective technique to connect the risk information of project activities to the baseline schedule, in order to provide sensitivity information of individual project activities to assess the potential impact of uncertainty on the final project duration and cost.

Since estimates about activity time and cost are predictions for the future and human beings often tend to be overly optimistic or, on the contrary, often add some reserve safety to protect themselves against unexpected events, knowledge about the potential impact of these estimation errors on the project objective is a key add-on to the construction of a project's baseline schedule.

Figure 7.7 displays the four different steps of a schedule risk analysis. Each step will be briefly explained along the following sections, and can be summarized as follows:

- Baseline schedule: Construct an activity timetable.
- Define uncertainty: Define activity time and cost probability distributions.
- Run Monte Carlo simulations: Run multiple project progress simulations.
- Interpret the simulation results: Interpret the sensitivity measures.

[4]This section is also available as an article entitled "Schedule Risk Analysis: How to measure your baseline schedule's sensitivity?" published online at PM Knowledge Center.

### *Step 1: Baseline Schedule*

The construction of a project baseline schedule involves the definition of start and finish times for each project activity, using earliest and latest start calculations with or without the presence of limited resources. There is a wide range of techniques available (PERT, CPM, etc.), which will not be discussed in this article.

The project baseline schedule serves as a point of reference to which the simulated project progress of step 3 is compared to. Although it is generally accepted that it is very unlikely that everything will go according to plan, the baseline schedule plays a central role in schedule risk analysis and the lack of it would lead to incomparable data or even biased results.

### *Step 2: Define Risk/Uncertainty*

Since time and cost estimates are often, if not always, subject to a margin or error, people feel more comfortable with a range of duration and cost estimates for project activities. Range estimates and risk assessment require analytical skills and basic knowledge of statistics, which is often perceived as mathematically complex and sometimes theoretical, and hence, far from practice. However, a basic understanding of probability and distribution functions already allows the project manager to improve estimating the effects of unexpected events on the project outcome.

### *Step 3: Monte Carlo Simulations*

Monte Carlo simulation is a simple technique to quickly generate multiple runs simulating real project progress. Each simulation run generates a duration and cost for each project activity given its uncertainty profile defined in step 2. During each simulation run, the simulation engine records all project schedules and critical paths during progress in order to be able to measure the degree of activity sensitivity and the expected impact of activity variation on the project objective, as reported in step 4.

### *Step 4: Sensitivity Results*

The output of a schedule risk analysis is a set of measures that define the degree of activity criticality and sensitivity. These measures refine the black-and-white view of the critical path (which defines that an activity is either critical or not) to a degree of sensitivity, as follows:

- Criticality Index (CI): Measures the probability that an activity is on the critical path.

- Significance Index (SI): Measures the relative importance of an activity.
- Schedule Sensitivity Index (SSI): Measures the relative importance of an activity taking the CI into account.
- Cruciality Index (CRI): Measures the correlation between the activity duration/cost and the total project. duration/cost.

Each measure gives the project manager an indication of how sensitive the activity is towards the final project duration or total cost (See article "*RA6: Time sensitivity*" on page 126). The values of the sensitivity measures are available upon completion of the simulation run and are used as triggers to focus on the risky activities that probably require higher attention in order to achieve successful project fulfillment (See article "*PC17: Bottom-up control*" on page 251).

## Questions

1. Which of the following statements is true?
   (a) SRA is a component of Monte Carlo simulation.
   (b) Monte Carlo simulation is a component of SRA.
2. Which of the definitions below is the right definition of the Cruciality Index CRI?
   (a) It measures the probability that an activity is on the critical path.
   (b) It measures the relative importance of an activity.
   (c) It measures the relative importance of an activity taking the CI into account.
   (d) It measures the correlation between the activity duration/cost and the total project duration/cost.
3. Which of the definitions below is the right definition of the Criticality Index CI?
   (a) It measures the relative importance of an activity.
   (b) It measures the correlation between the activity duration/cost and the total project duration/cost.
   (c) It measures the probability that an activity is on the critical path.
4. Which of the definitions below is the right definition of the Sensitivity Index SI?
   (a) It measures the probability that an activity is on the critical path.
   (b) It measures the relative importance of an activity taking the CI into account.
   (c) It measures the relative importance of an activity.
   (d) It measures the correlation between the activity duration/cost and the total project duration/cost.

5. Which of the definitions below is the right definition of the Schedule Sensitivity Index SSI?
   (a) It measures the relative importance of an activity taking the CI into account.
   (b) It measures the probability that an activity is on the critical path.
   (c) It measures the relative importance of an activity.
   (d) It measures the correlation between the activity duration/cost and the total project duration/cost.
6. Is an activity with a high CI and a very low SI a good candidate for intensive control?
   (a) Yes
   (b) No

## RA6: Time Sensitivity[5]

Schedule Risk Analysis (SRA) is a simple yet effective technique to connect the risk information of project activities to the baseline schedule, in order to provide sensitivity information of individual project activities to assess the potential impact of uncertainty on the final project duration. A traditional schedule risk analysis requires four steps, as described in article "*RA5: Schedule sensitivity*" on page 123, to report activity sensitivity measures that evaluate each activity's time estimate on a scale of risk. These sensitivity measures can be used by the project manager to distinguish between risky and less risky activities in order to better focus on those activities that might have an impact on the overall project objective, as described in article "*PC17: Bottom-up control*" on page 251.
In this article, the four sensitivity measures introduced in article "*RA5: Schedule sensitivity*" on page 123 will be discussed in detail.

### *Criticality Index (CI)*

The Criticality Index measures the probability that an activity lies on the critical path. It is a simple measure expressed as a percentage denoting the likelihood of being critical. Although the CI has been used throughout various studies and implemented in many software tools, the CI often fails in adequately measuring the project risk. The main drawback of the CI is that its focus is restricted to measuring probability, which does not necessarily mean that high CI activities have a high

[5] This section is also available as an article entitled "Schedule Risk Analysis: Measuring the time sensitivity of an activity" published online at PM Knowledge Center.

impact on the total project duration (e.g. think of an activity with a very low duration always lying on the critical path, but with a low impact on the total project duration due to its negligible duration). More information and an example project simulation run can be found in article "*RA7: Criticality index*" on page 130.

## *Significance Index (SI)*

In order to reflect the relative importance between project activities, the Sensitivity Index of a project activity can be calculated as follows:

$$\mathrm{SI} = E\{\frac{\text{Activity Duration}}{\text{Activity Duration} + \text{Activity Slack}} * \frac{\text{Project Duration}}{E\{\text{Project Duration}\}}\}$$

with $E\{\mathrm{x}\}$ the expected value of x.

The SI has been defined as a partial answer to the criticism on the CI. Rather than expressing an activity's criticality by the probability concept, the SI aims at exposing the significance of individual activities on the total project duration. In some examples, the SI seems to provide more acceptable information about the relative importance of activities. Despite this, there are still examples where counter-intuitive results are reported. More information and an example project simulation run can be found in article "*RA12: Significance index*" on page 150.

## *Schedule Sensitivity Index (SSI)*

The Project Management Body Of Knowledge (PMBOK) mentions quantitative risk analysis as one of many risk assessment methods, and proposes to combine the activity duration and project duration standard deviations with the CI. The Schedule Sensitivity Index is calculated as follows:

$$\mathrm{SSI} = \frac{StDev\{\text{Activity Duration}\} * \mathrm{CI}}{StDev\{\text{Project Duration}\}}$$

with $StDev\{\mathrm{x}\}$ the standard deviation of x.

More information and an example project simulation run can be found in article "*RA11: Schedule sensitivity index*" on page 146.

### *Cruciality Index (CRI)*

Another measure to calculate the duration sensitivity of individual activities is given by the correlation between the activity duration and the total project duration and can be calculated as follows:

$$\text{CRI} = \mid Corr\{\text{Activity Duration, Project Duration}\} \mid$$

with $Corr\{x,y\}$ the correlation between x and y.

This measure reflects the relative importance of an activity in a more intuitive way and calculates the portion of total project duration uncertainty that can be explained by the uncertainty of an activity.

- Pearson's product-moment CRI($r$) is a traditional measure of the degree of linear relationship between two variables. The correlation is 1 in the case of a clear positive linear relationship, $-1$ in the case of a clear negative linear relationship, and some value in between in all other cases, indicating the degree of linear dependence between the activity duration and the total project duration. The closer the coefficient is to either $-1$ or 1, the stronger the correlation between these two variables. When the absolute value is taken, the CRI($r$) lies between 0 and 1.

However, the relation between an activity duration and the total project duration often follows a nonlinear relation. Therefore, nonlinear correlation measures such as the Spearman rank correlation coefficient or Kendall's tau measure can also be calculated. These two correlation measures can be calculated as follows:

- Spearman's rank correlation CRI($\rho$) (rho) assumes that the values for the variables (i.e. activity durations and project durations) are converted to ranks, followed by the calculation of the difference between the ranks of each observation of the two variables. The measure is a so-called nonparametric measure to deal with situations where the strict statistical assumptions of the parametric CRI($r$) measure are not met. The CRI($\rho$) measure has a similar meaning to the CRI($r$) measure, i.e. $-1 \leq \text{CRI}(\rho) \leq 1$ or, when the absolute value is taken, $0 \leq \text{CRI}(\rho) \leq 1$.
- Kendall's tau rank correlation CRI($\tau$) (tau) index measures the degree of correspondence between two rankings and assesses the significance of this correspondence. This nonparametric measure has a similar meaning to the CRI($r$) measure, i.e. $-1 \leq \text{CRI}(\tau) \leq 1$ or, when the absolute value is taken, $0 \leq \text{CRI}(\tau) \leq 1$.

More information and an example project simulation run can be found in three articles on the cruciality index, article "*RA8: Pearson's cruciality index*" on page 134, article "*RA10: Spearman's cruciality index*" on page 142 and article "*RA9: Kendall's tau cruciality index*" on page 138. Note that these three versions of the cruciality index can also be used to measure the sensitivity of the cost of each activity.

### Questions

1. Which of these metrics is not one of the four basic metrics used for SRA?
   (a) Criticality Index (CI)
   (b) Significance Index (SI)
   (c) Schedule Impact Index (SII)
   (d) Schedule Sensitivity Index (SSI)
   (e) Cruciality Index (CRI)
2. The cruciality index (CRI) measures …
   (a) the probability of an activity lying on the critical path.
   (b) the correlation between the activity duration and the total project duration.
   (c) both of the above.
3. Which of the activities below should be the focus of a project manager's attention?
   (a) An activity with a very high Cruciality Index CRI.
   (b) An activity with a very low Cruciality Index CRI.
   (c) Neither answer (a) nor answer (b).
   (d) Both answer (a) and answer (b) (note: Since the CRI is measured as an absolute value, the lowest value is equal to 0).
4. Which of the statistical techniques used to calculate the Cruciality Index is not a nonparametric test?
   (a) Spearman's rank correlation
   (b) Kendall's tau rank correlation
   (c) Pearson's product moment
5. Non-parametric tests can be of use when the relationship between an activity and the project duration is …
   (a) influenced by confounding variables.
   (b) of a nonlinear nature.
   (c) weak for every activity of the project.
6. Using sensitivity measures to monitor activities during the execution of a project is an example of …
   (a) top-down project management.
   (b) bottom-up project management.

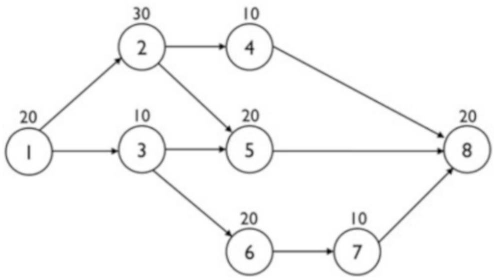



**Fig. 7.8** A fictitious example project network

## Time Sensitivity Measures

All calculations for this section have been made in MS Excel without rounding. However, since the results in Tables 7.3, 7.4, 7.5, 7.6 and 7.7 are displayed with rounding, some slightly different numbers might appear when using manual calculations. To that purpose, these tables are available in MS Excel at www.or-as.be/books/pmkc.

## *RA7: Criticality Index*[6]

This article discusses the use of time sensitivity measures introduced in article "*RA6: Time sensitivity*" on page 126. These measures are the result of a Schedule Risk Analysis (SRA) using Monte Carlo simulations explained in article "*RA5: Schedule sensitivity*" on page 123. More precisely, the Criticality Index (CI) calculations are illustrated on a fictitious example project network with eight activities, as displayed in Fig. 7.8. The number above each node is the baseline duration estimate (in days).

### Critical Path Versus Criticality Index

The baseline schedule has a planned duration PD = 90 days and the critical path is equal to the activity sequence 1-2-5-8. However, the baseline schedule is only an estimate of the real project duration and hence, the real project duration and the critical path might differ from the baseline schedule estimates. Consequently, the critical path is a black-and-white view on the critical parts of a project, and should be refined to capture more detailed and precise sensitivity information during project progress. The criticality index is a sensitivity measure that balances between this

[6]This section is also available as an article entitled "Measuring time sensitivity in a project: The criticality index" published online at PM Knowledge Center.

**Table 7.1** Ten simulation runs with random activity durations

| | *SAD* | | | | | | | | *SPD* | *CP* |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | |
| Run 1 | 4 | 12 | 1 | 4 | 5 | 4 | 7 | 8 | 29 | 1-2-5-8 |
| Run 2 | 22 | 23 | 14 | 5 | 28 | 26 | 5 | 29 | 102 | 1-2-5-8 |
| Run 3 | 25 | 38 | 12 | 15 | 24 | 20 | 13 | 22 | 109 | 1-2-5-8 |
| Run 4 | 25 | 25 | 15 | 13 | 25 | 25 | 13 | 10 | 88 | 1-3-6-7-8 |
| Run 5 | 21 | 42 | 12 | 7 | 30 | 21 | 14 | 23 | 116 | 1-2-5-8 |
| Run 6 | 28 | 44 | 7 | 9 | 15 | 20 | 10 | 22 | 109 | 1-2-5-8 |
| Run 7 | 19 | 21 | 14 | 13 | 23 | 24 | 14 | 28 | 99 | 1-3-6-7-8 |
| Run 8 | 12 | 36 | 14 | 9 | 19 | 17 | 7 | 24 | 91 | 1-2-5-8 |
| Run 9 | 28 | 35 | 7 | 12 | 28 | 27 | 13 | 28 | 119 | 1-2-5-8 |
| Run 10 | 27 | 44 | 6 | 5 | 30 | 10 | 11 | 28 | 129 | 1-2-5-8 |

black (an activity is not critical) and white (an activity is critical) point-of-view. More precisely, the criticality index measures the probability that an activity lies on the critical path. It is a simple measure expressed as a percentage denoting the likelihood of being critical.

Table 7.1 displays ten simulation runs where each activity has a certain duration that might differ from the original baseline duration estimate given in Fig. 7.8. These durations are further referred to as the simulated activity durations (*SAD*). The simulated project duration (*SPD*) and the critical path (*CP*) per simulation run is also given. Each simulation run reflects a possible real project progress scenario where unexpected changes to the activity estimates might occur.

Based on the simulation runs, the criticality index can be easily calculated. As an example, activity 2 lies on the critical path in 8 of the 10 runs, and hence, the criticality index is equal to 80 %. Although this activity lies on the critical path in the baseline schedule, the criticality index shows that the probability that this activity is critical is only 80 %. Figure 7.9 shows the CI values for all project activities, ranging between 0 and 100 %. More precisely, the values are equal to 1.00, 0.80, 0.20, 0.00, 0.80, 0.20, 0.20 and 1.00 for activities 1–8. Obviously, a schedule risk analysis should be done with care for the following two reasons: (1) the input distributions of the activity duration should reflect reality, and (2) the number of simulation runs should exceed 10 to provide more reliable results.

The graph of Fig. 7.9 can be used to set action thresholds that make a distinction between sensitive and insensitive activities to better steer the project control process during project progress. More information can be found in article "*PC17: Bottom-up control*" on page 251. Apart from the criticality index, other time sensitivity measures can also be calculated for the same simulation runs, as discussed in article "*RA6: Time sensitivity*" on page 126.

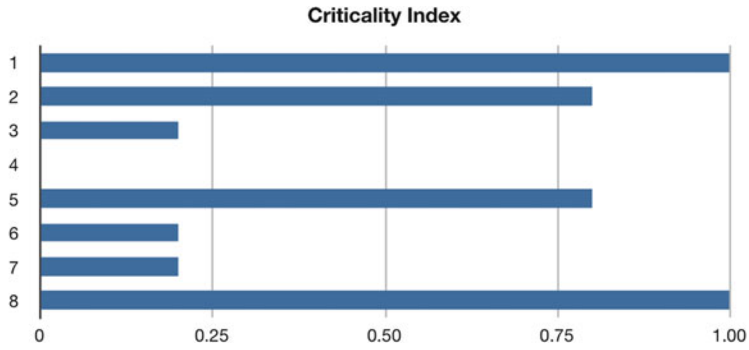



**Fig. 7.9** The criticality index values for the eight activities after ten simulation runs

**Table 7.2** An example simulation run

| | Run 1 | Run 2 | Run 3 | Run 4 |
|---|---|---|---|---|
| Act A duration (*SAD*) | 8 | 10 | 12 | 10 |
| Act B duration (*SAD*) | 6 | 12 | 12 | 8 |
| Project duration (*SPD*) | 8 | 12 | 12 | 10 |

## Questions

1. An activity in a serial network (i.e. execute activity 1, then activity 2, then activity 3, ...) will have a criticality index value of ...
   (a) CI = 0.
   (b) CI = 1.
   (c) CI = 1/# activities.
   (d) The value can only be calculated using simulations, since this depends on the specific activity duration distributions.
2. An activity in a parallel network (i.e. all activities can be simultaneously executed), will have a criticality index value of ...
   (a) CI = 0.
   (b) CI = 1.
   (c) CI = 1/# activities.
   (d) This has to be calculated using simulations, since this depends on the specific activity duration distributions.
3. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2. The criticality index CI of activity A is equal to ...
   (a) 50 %.
   (b) 65 %.
   (c) 75 %.
   (d) 95 %.

4. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2. The criticality index CI of activity B is equal to …
   (a) 50 %.
   (b) 65 %.
   (c) 75 %.
   (d) 95 %.
5. Suppose a third activity C is added to the project described earlier (two parallel activities A and B) without any precedence relations. Which of the statements below correctly describes the potential impact on the CI of activities A and B?
   (a) The CI values of activities A and B decrease.
   (b) The CI values of activities A and B decrease or remain the same.
   (c) The CI values of activities A and B increase.
   (d) The CI values of activities A and B increase or remain the same.
   (e) This is impossible to determine without simulation runs.
6. Suppose a third activity C is added to the project described earlier (two parallel activities A and B) as a successor of activity A. Which of the statements below correctly describes the potential impact on the CI of activities A and B?
   (a) The CI of activity B will probably decrease.
   (b) The CI of activity B will probably increase.
   (c) The CI of activity A will probably decrease.
   (e) The CI of activity A will probably increase.
   (f) Most likely both a and c.
   (g) Most likely both a and d.
   (h) Most likely both b and c.
   (i) Most likely both b and d.
7. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The criticality index CI of activity B is equal to …
   (a) 33 %.
   (b) 50 %.
   (c) 90 %.
   (d) 95 %.
8. An activity C is added to the project from the previous question. This activity has to be executed after activity B has been completed and has the following probability distribution: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The criticality index CI of activity A is equal to …
   (a) 25 %.
   (b) 33 %.
   (c) 50 %.
   (d) 66 %.

## *RA8: Pearson's Cruciality Index*[7]

This article discusses the use of time sensitivity measures introduced in article "*RA6: Time sensitivity*" on page 126. These measures are the result of a Schedule Risk Analysis (SRA) using Monte Carlo simulations explained in article "*RA5: Schedule sensitivity*" on page 123. More precisely, the Pearson's product-moment cruciality index (CRI($r$)) calculations are illustrated on a fictitious example project network with eight activities, as displayed in Fig. 7.8 on page 130. This example project has been used in previous articles and contains numbers above each node to denote the baseline duration estimate (in days), resulting in a baseline schedule with a planned duration PD of 90 days and the critical path equal to the activity sequence 1-2-5-8.

### Cruciality Index CRI(*r*)

The cruciality index (CRI) is a measure that calculates the correlation between the activity duration and the total project duration, as follows:

$$\text{CRI} = \mid Corr(SAD, SPD) \mid$$

with

| | |
|---|---|
| \| x \| | The absolute value of x |
| *Corr*{x,y} | The correlation between x and y |
| *SAD* | The simulated activity duration |
| *SPD* | The simulated project duration |

This measure reflects the relative importance of an activity and calculates the portion of total project duration uncertainty that can be explained by the uncertainty of an activity, and can be measured in different ways. The abbreviations *SAD* and *SPD* are used to stress that all values for the activity durations and the project duration are obtained using Monte Carlo simulation runs.

One way to measure the CRI is the Pearson's product-moment CRI($r$), which is a traditional measure of the degree of linear relationship between two variables. The correlation is 1 in the case of a clear positive linear relationship, $-1$ in the case of a clear negative linear relationship, and some value in between in all other cases, indicating the degree of linear dependence between the activity duration and the total project duration. The closer the coefficient is to either $-1$ or 1, the stronger the correlation between these two variables. The absolute value of this correlation measure obviously lies between 0 (no relation) and 1 (perfect relation).

[7] This section is also available as an article entitled "Measuring time sensitivity in a project: The cruciality index (Pearson's product-moment)" published online at PM Knowledge Center.

**Table 7.3** Intermediate calculations for the CRI($r$) of activity 2

| | $(SAD - \overline{SAD})^2$ | $(SPD - \overline{SPD})^2$ | $(SAD - \overline{SAD}) * (SPD - \overline{SPD})$ |
|---|---|---|---|
| Run 1 | 400 | 4,914.01 | 1,402.0 |
| Run 2 | 81 | 8.41 | −26.1 |
| Run 3 | 36 | 98.01 | 59.4 |
| Run 4 | 49 | 123.21 | 77.7 |
| Run 5 | 100 | 285.61 | 169.0 |
| Run 6 | 144 | 98.01 | 118.8 |
| Run 7 | 121 | 0.01 | 1.1 |
| Run 8 | 16 | 65.61 | −32.4 |
| Run 9 | 9 | 396.01 | 59.7 |
| Run 10 | 144 | 894.01 | 358.8 |
| SUM | 1,100 | 6,882.90 | 2,188.0 |

The Pearson's product-moment cruciality index CRI($r$) can be calculated as follows:

$$\mathrm{CRI}(r) = \left| \frac{\sum\{(SAD - \overline{SAD}) * (SPD - \overline{SPD})\}}{\sqrt{\sum(SAD - \overline{SAD})^2 * \sum(SPD - \overline{SPD})^2}} \right|$$

with $\sum$ (x) the sum of all x-values over all simulation runs and $\overline{\mathrm{x}}$ the average value of x over all simulation runs.

Table 7.1 on page 131 displays ten simulation runs where each activity has a certain duration that might differ from the original baseline duration estimate given in Fig. 7.8. The simulated project duration ($SPD$) for activities 1–8 per simulation run is also given. Each simulation run reflects a possible real project progress scenario where unexpected changes in the activity estimates might occur.

Based on the simulation runs, the cruciality index CRI($r$) can be calculated with the intermediate calculations illustrated in Table 7.3 for activity 2 of Fig. 7.8. In the example, the values for $(SAD - \overline{SAD})^2$, $(SPD - \overline{SPD})^2$ and $(SAD - \overline{SAD}) * (SPD - \overline{SPD})$ are calculated for activity 2 with $\overline{SAD} = E(SAD) = 32$ and $\overline{SPD} = E(SPD) = 99.1$ as the averages or expected values of the $SAD$ and $SPD$ columns of Table 7.1.

Based on the values in Table 7.3, the various components of the CRI($r$) formula can be easily calculated as follows:

- $\sum(SAD - \overline{SAD})^2 = 1{,}100$ and is equal to the sum of the second column of Table 7.3
- $\sum(SPD - \overline{SPD})^2 = 6{,}882.90$ and is equal to the sum of the third column of Table 7.3
- $\sum\{(SAD - \overline{SAD}) * (SPD - \overline{SPD})\} = 2{,}188.0$ and is equal to the sum of the last column of Table 7.3

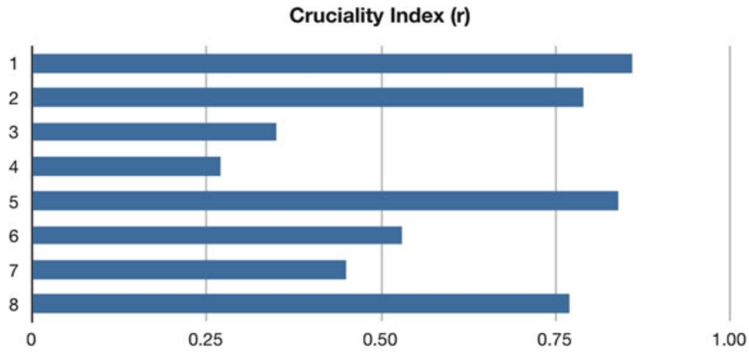



**Fig. 7.10** The cruciality index (Pearson's product-moment) values for the eight activities after ten simulation runs

and consequently, the cruciality index is equal to

$$\text{CRI}(r) = |2{,}188.0/\sqrt{1{,}100 * 6{,}882.90}| = 0.80.$$

Although activity 2 lies on the critical path of the baseline schedule, the cruciality index (based on Pearson's product-moment) is lower than 1. Figure 7.10 shows the CRI($r$) values for all project activities, ranging between 27 and 86 %. More precisely, the values are equal to 0.86, 0.80, 0.34, 0.27, 0.84, 0.53, 0.45 and 0.77 for activities 1–8. Obviously, a schedule risk analysis should be done with care for the following two reasons: (1) the input distributions of the activity duration should reflect reality, and (2) the number of simulation runs should exceed 10 to provide more reliable results.

The graph of Fig. 7.10 can be used to set action thresholds that make a distinction between sensitive and insensitive activities to better steer the project control process during project progress. More information can be found in article "*PC17: Bottom-up control*" on page 251. Apart from the cruciality index, other time sensitivity measures can also be calculated for the same simulation runs, as discussed in article "*RA6: Time sensitivity*" on page 126.

## Questions

1. An activity in a serial network (i.e. execute activity 1, then activity 2, then activity 3, …) will have a cruciality index (Pearson's product-moment) value of …

   (a) CRI($r$) $= 0$.
   (b) CRI($r$) $\in ]0, 0.5]$.
   (c) CRI($r$) $\in [0.5, 1]$.
   (d) CRI($r$) $= 1$.
   (e) CRI($r$) $= 1/\#$ activities.

(f) The value can only be calculated using simulations, since this depends on the specific activity duration distributions.

2. An activity in a parallel network (i.e. all activities can be simultaneously executed), will have a cruciality index (Pearson's product-moment) value of ...

   (a) $\text{CRI}(r) = 0$.
   (b) $\text{CRI}(r) \in ]0, 0.5]$.
   (c) $\text{CRI}(r) \in [0.5, 1]$.
   (d) $\text{CRI}(r) = 1$.
   (e) $\text{CRI}(r) = 1/\#$ activities.
   (f) This has to be calculated using simulations, since this depends on the specific activity duration distributions.

3. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The cruciality index (Pearson's product-moment) CRI($r$) of activity A is equal to ... (choose closest number)

   (a) 50 %.
   (b) 85 %.
   (c) 99 %.
   (d) 100 %.

4. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The cruciality index (Pearson's product-moment) CRI($r$) of activity B is equal to ... (choose closest number)

   (a) 50 %.
   (b) 85 %.
   (c) 99 %.
   (d) 100 %.

5. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The cruciality index CRI($r$) of activity A is equal to ...

   (a) 0 %.
   (b) 25 %.
   (c) 50 %.
   (d) 75 %.
   (e) 100 %.

6. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The cruciality index CRI($r$) of activity B is equal to ...

   (a) 0 %.
   (b) 25 %.

(c) 50 %.
(d) 75 %.
(e) 100 %.
(f) The CRI($r$) value cannot be calculated

7. Assume that activity B from the previous question is not deterministic, but has the same distribution as activity A. The cruciality CRI($r$) of activity A is equal to …

   (a) 25 %.
   (b) 37 %.
   (c) 50 %.
   (d) 58 %.
   (e) 100 %.

## *RA9: Kendall's Tau Cruciality Index*[8]

This article discusses the use of time sensitivity measures introduced in article "*RA6: Time sensitivity*" on page 126. These measures are the result of a Schedule Risk Analysis (SRA) using Monte Carlo simulations explained in article "*RA5: Schedule sensitivity*" on page 123. More precisely, the Kendall's tau rank correlation cruciality index (CRI($\tau$) (tau)) calculations are illustrated on a fictitious example project network with eight activities, as displayed in Fig. 7.8 on page 130. This example project has been used in previous articles and contains numbers above each node to denote the baseline duration estimate (in days), resulting in a baseline schedule with a planned duration PD of 90 days and the critical path equal to the activity sequence 1-2-5-8.

### Cruciality Index CRI($\tau$)

The cruciality index (CRI) is a measure that calculates the correlation between the activity duration and the total project duration as follows:

$$\text{CRI} = \mid Corr(SAD, SPD) \mid$$

with

| | |
|---|---|
| $\mid$ x $\mid$ | The absolute value of x |
| $Corr\{$x,y$\}$ | The correlation between x and y |
| *SAD* | The simulated activity duration |
| *SPD* | The simulated project duration |

[8]This section is also available as an article entitled "Measuring time sensitivity in a project: The cruciality index (Kendall's tau rank correlation)" published online at PM Knowledge Center.

This measure reflects the relative importance of an activity and calculates the portion of total project duration uncertainty that can be explained by the uncertainty of an activity, and can be measured in different ways. The abbreviations *SAD* and *SPD* are used to stress that all values for the activity durations and the project duration are obtained using Monte Carlo simulation runs.

One way to measure the CRI is the Pearson's product-moment CRI($r$), which is a traditional measure of the degree of linear relationship between two variables (see article "*RA8: Pearson's cruciality index*" on page 134). However, when it is conjectured that the relation between the variables is nonlinear, the Kendall's tau rank correlation CRI($\tau$) might be a good alternative. The CRI($\tau$) assumes that the values for the variables (i.e. activity durations and project durations) are converted to ranks, measures the degree of correspondence between two rankings and assesses the significance of this correspondence. The measure is a so-called nonparametric measure to deal with situations where the strict statistical assumptions of the parametric CRI($r$) measure are not met.

The Kendall's tau rank correlation cruciality index CRI($\tau$) can be calculated as follows:

$$\mathrm{CRI}(\tau) = \left|\frac{4 * PairwiseComparisonValue}{nrs * (nrs - 1)} - 1\right|$$

with *nrs* the number of simulation runs.

The value for the variable *PairwiseComparisonValue* is calculated by performing $((nrs-1)+(nrs-2)+\ldots+1)$ comparisons of activity and project duration values between all the simulation runs. In each comparison between simulation runs x and y, the following value VAL is calculated:

$$\text{if } \{(SAD^x - SAD^y) * (SPD^x - SPD^y)\} > 0 \text{ then VAL} = 1 \text{ else VAL} = 0$$

with

$SAD^x$ The simulated activity duration of simulation run x
$SPD^x$ The simulated project duration of simulation run x

The value for the variable *PairwiseComparisonValue* in the CRI($\tau$) formula is then the sum of all VAL values over all comparisons.

Table 7.1 on page 131 displays ten simulation runs where each activity has a certain duration that might differ from the original baseline duration estimate given in Fig. 7.8. The simulated project duration (*SPD*) for activities 1–8 per simulation run is also given. Each simulation run reflects a possible real project progress scenario where unexpected changes in the activity estimates might occur.

Based on the simulation runs, the cruciality index CRI($\tau$) can be easily calculated as illustrated for activity 2 of Fig. 7.8. The value for the *PairwiseComparisonValue* is equal to the sum of all VAL values as calculated below:

Simulation runs 1 and 2: $(12 - 23) * (29 - 102) > 0$ so VAL $= 1$
Simulation runs 1 and 3: $(12 - 38) * (29 - 109) > 0$ so VAL $= 1$

...
Simulation runs 9 and 10: $(35 - 44) * (119 - 129) >$ so VAL $= 1$

In total, there are 45 combinations. The number of times the VAL variable is equal to 1 is given for all simulations runs. As an example, simulation run 1 will be compared with all 9 others, and 9 of the comparisons give a VAL $= 1$. Simulation run 2 will only be compared with 8 other simulation runs (with run 3 to run 10) and only 6 comparisons give a VAL $= 1$. The complete list of 45 comparisons results in the following number of VAL $= 1$ observations:

Simulation run 1: 9 (= all)
Simulation run 2: 6 (of 8)
Simulation run 3: 5 (of 7)
Simulation run 4: 5 (of 6)
Simulation run 5: 3 (of 5)
Simulation run 6: 2 (of 4)
Simulation run 7: 2 (of 3)
Simulation run 8: 1 (of 2)
Simulation run 9: 1 (= all)

The sum of all VAL values is equal to $PairwiseComparisonValue = 34$, and hence, the CRI($\tau$) can be easily calculated as follows:

$$\mathrm{CRI}(\tau) = |(4 * 34)/(10 * 9) - 1| = 0.51.$$

Although activity 2 lies on the critical path in the baseline schedule, the cruciality index (based on Kendall's tau rank correlation) is lower than 1. Figure 7.11 shows the CRI($\tau$) values for all project activities, ranging between 7 and 56 %. More precisely, the values are equal to 0.47, 0.51, 0.56, 0.11, 0.47, 0.11, 0.07 and 0.33 for activities 1–8. Obviously, a schedule risk analysis should be done with care for the following two reasons: (1) the input distributions of the activity duration should reflect reality, and (2) the number of simulation runs should exceed 10 to provide more reliable results.

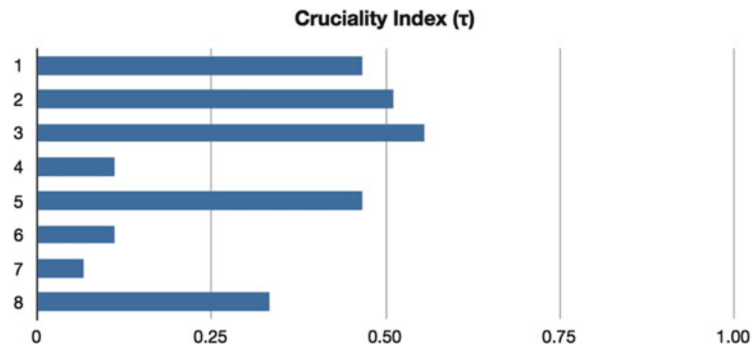



**Fig. 7.11** The cruciality index (Kendall's tau rank correlation) values for the eight activities after ten simulation runs

The graph of Fig. 7.11 can be used to set action thresholds that make a distinction between sensitive and insensitive activities to better steer the project control process during project progress. More information can be found in article "*PC17: Bottom-up control*" on page 251. Apart from the cruciality index, other time sensitivity measures can also be calculated for the same simulation runs, as discussed in article "*RA6: Time sensitivity*" on page 126.

## Questions

1. An activity in a serial network (i.e. execute activity 1, then activity 2, then activity 3, ...) will have a cruciality index (Kendall's tau rank correlation) value of ...

   (a) $\text{CRI}(\tau) = 0$.
   (b) $\text{CRI}(\tau) \in ]0, 0.5]$.
   (c) $\text{CRI}(\tau) \in [0.5, 1]$.
   (d) $\text{CRI}(\tau) = 1$.
   (e) $\text{CRI}(\tau) = 1/\#$ activities.
   (f) The value can only be calculated using simulations, since this depends on the specific activity duration distributions.

2. An activity in a parallel network (i.e. all activities can be simultaneously executed), will have a cruciality index (Kendall's tau rank correlation) value of ...

   (a) $\text{CRI}(\tau) = 0$.
   (b) $\text{CRI}(\tau) \in ]0, 0.5]$.
   (c) $\text{CRI}(\tau) \in [0.5, 1]$.
   (d) $\text{CRI}(\tau) = 1$.
   (e) $\text{CRI}(\tau) = 1/\#$ activities.
   (f) This has to be calculated using simulations, since this depends on the specific activity duration distributions.

3. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The cruciality index (Kendall's tau rank correlation) $\text{CRI}(\tau)$ of activity A is equal to ... (choose closest number)

   (a) 11 %.
   (b) 33 %.
   (c) 67 %.
   (d) 83 %.

4. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The cruciality index (Kendall's tau rank correlation) $\text{CRI}(\tau)$ of activity B is equal to ... (choose closest number)

   (a) 11 %.
   (b) 33 %.

(c) 67 %.
(d) 83 %.

5. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The cruciality index CRI($\tau$) of activity A is equal to …

(a) 0 %.
(b) 25 %.
(c) 33 %.
(d) 50 %.
(e) 66 %.
(f) 75 %.
(g) 100 %.

6. Does the CRI($\tau$) of activity A change when the activities A and B are now two serial activities (instead of two parallel activities)?

(a) Yes
(b) No
(c) The new value cannot be determined.

## *RA10: Spearman's Cruciality Index*[9]

This article discusses the use of time sensitivity measures introduced in article "*RA6: Time sensitivity*" on page 126. These measures are the result of a Schedule Risk Analysis (SRA) using Monte Carlo simulations explained in article "*RA5: Schedule sensitivity*" on page 123. More precisely, the Spearman's rank correlation cruciality index (CRI($\rho$) (rho)) calculations are illustrated on a fictitious example project network with eight activities, as displayed in Fig. 7.8 on page 130. This example project has been used in previous articles and contains numbers above each node to denote the baseline duration estimate (in days), resulting in a baseline schedule with a planned duration PD of 90 days and the critical path equal to the activity sequence 1-2-5-8.

### Cruciality Index CRI($\rho$)

The cruciality index (CRI) is a measure that calculates the correlation between the activity duration and the total project duration as follows:

$$\text{CRI} = \mid Corr(SAD, SPD) \mid$$

[9]This section is also available as an article entitled "Measuring time sensitivity in a project: The cruciality index (Spearman's rank correlation)" published online at PM Knowledge Center.

with

| | |
|---|---|
| $\mid x \mid$ | The absolute value of x |
| *Corr*{x,y} | The correlation between x and y |
| *SAD* | The simulated activity duration |
| *SPD* | The simulated project duration |

This measure reflects the relative importance of an activity and calculates the portion of total project duration uncertainty that can be explained by the uncertainty of an activity, and can be measured in different ways. The abbreviations *SAD* and *SPD* are used to stress that all values for the activity durations and the project duration are obtained using Monte Carlo simulation runs.

One way to measure the CRI is the Pearson's product-moment CRI($r$), which is a traditional measure of the degree of linear relationship between two variables (see article "*RA8: Pearson's cruciality index*" on page 134). However, when it is conjectured that the relation between the variables is nonlinear, the Spearman's rank correlation CRI($\rho$) might be a good alternative. The CRI($\rho$) assumes that the values for the variables (i.e. activity durations and project durations) are converted to ranks, followed by the calculation of the difference between the ranks of each observation of the two variables. The measure is a so-called nonparametric measure to deal with situations where the strict statistical assumptions of the parametric CRI($r$) measure are not met. The Spearman's rank correlation cruciality index CRI($\rho$) can be calculated as follows:

$$\text{CRI}(\rho) = |1 - 6 * \frac{\sum\{\text{DiffRank}^2\}}{nrs * (nrs^2 - 1)}|$$

with

| | |
|---|---|
| $\sum\{x\}$ | The sum of all x-values over all simulation runs |
| *nrs* | The number of simulation runs |
| DiffRank | The difference between the ranking values of *SAD* and *SPD* |

Table 7.1 on page 131 displays ten simulation runs where each activity has a certain duration that might differ from the original baseline duration estimate given in Fig. 7.8. The simulated project duration (*SPD*) for activities 1–8 per simulation run is also given. Each simulation run reflects a possible real project progress scenario where unexpected changes in the activity estimates might occur.

Based on the simulation runs, the cruciality index CRI($\rho$) can be calculated with the intermediate calculations illustrated in Table 7.4 for activity 2 of Fig. 7.8. In the example, the ranking values are given for *SAD* and *SPD*. Note that the lowest duration receives a ranking value equal to 1, and also note that the average of the ranking values is calculated in case of tie breaks. As an example, activity 2 has two times a duration of 44 (simulation runs 6 and 10) and hence, the ranking value is equal to the average of 9 and 10, which is equal to 9.5.

**Table 7.4** Intermediate calculations for the CRI($\rho$) of activity 2

| | Ranking value for *SAD* of activity 2 | Ranking value for *SPD* | DiffRank$^2$ |
|---|---|---|---|
| Run 1 | 1 | 1 | 0 |
| Run 2 | 3 | 5 | 4 |
| Run 3 | 7 | 6.5 | 0.25 |
| Run 4 | 4 | 2 | 4 |
| Run 5 | 8 | 8 | 0 |
| Run 6 | 9.5 | 6.5 | 9 |
| Run 7 | 2 | 4 | 4 |
| Run 8 | 6 | 3 | 9 |
| Run 9 | 5 | 9 | 16 |
| Run 10 | 9.5 | 10 | 0.25 |

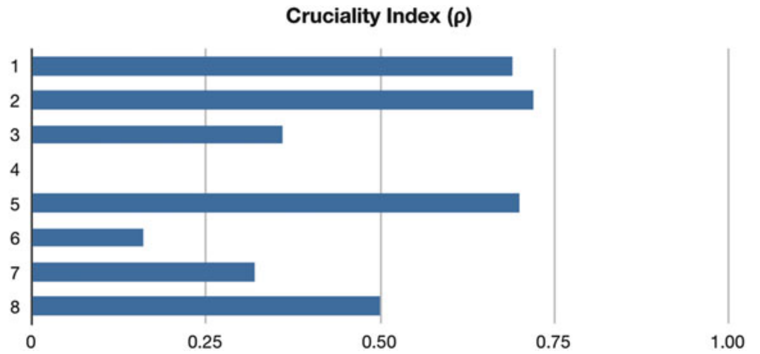



**Fig. 7.12** The cruciality index (Spearman's rank correlation) values for the eight activities after ten simulation runs

Based on the values in Table 7.4, the CRI($\rho$) can be easily calculated as follows:

$$\text{CRI}(\rho) = |1 - 6 * 46.5/\{10 * (10^2 - 1)\}| = 0.72$$

with $\sum \text{DiffRank}^2 = 46.5$ equal to the sum of the last column of Table 7.4.

Although activity 2 lies on the critical path of the baseline schedule, the cruciality index (based on Spearman's rank correlation) is lower than 1. Figure 7.12 shows the CRI($\rho$) values for all project activities, ranging between 0 and 72 %. More precisely, the values are equal to 0.69, 0.72, 0.36, 0.00, 0.70, 0.16, 0.32 and 0.50 for activities 1–8. Obviously, a schedule risk analysis should be done with care for the following two reasons: (1) the input distributions of the activity duration should reflect reality, and (2) the number of simulation runs should exceed 10 to provide more reliable results.

The graph of Fig. 7.12 can be used to set action thresholds that make a distinction between sensitive and insensitive activities to better steer the project control process during project progress. More information can be found in article "*PC17: Bottom-up control*" on page 251. Apart from the cruciality index, other time sensitivity measures can also be calculated for the same simulation runs, as discussed in article "*RA6: Time sensitivity*" on page 126.

## Questions

1. An activity in a serial network (i.e. execute activity 1, then activity 2, then activity 3, ...) will have a cruciality index (Spearman's rank correlation) value of ...

   (a) $\text{CRI}(\rho) = 0$.
   (b) $\text{CRI}(\rho) \in ]0, 0.5]$.
   (c) $\text{CRI}(\rho) \in [0.5, 1]$.
   (d) $\text{CRI}(\rho) = 1$.
   (e) $\text{CRI}(\rho) = 1/\#$ activities.
   (f) The value can only be calculated using simulations, since this depends on the specific activity duration distributions.

2. An activity in a parallel network (i.e. all activities can be simultaneously executed), will have a cruciality index (Spearman's rank correlation) value of ...

   (a) $\text{CRI}(\rho) = 0$.
   (b) $\text{CRI}(\rho) \in ]0, 0.5]$.
   (c) $\text{CRI}(\rho) \in [0.5, 1]$.
   (d) $\text{CRI}(\rho) = 1$.
   (e) $\text{CRI}(\rho) = 1/\#$ activities.
   (f) This has to be calculated using simulations, since this depends on the specific activity duration distributions.

3. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The cruciality index (Spearman's rank correlation) $\text{CRI}(\rho)$ of activity A is equal to ... (choose closest number)

   (a) 50 %.
   (b) 85 %.
   (c) 95 %.
   (d) 100 %.

4. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The cruciality index (Spearman's rank correlation) $\text{CRI}(\rho)$ of activity B is equal to ... (choose closest number)

   (a) 50 %.
   (b) 85 %.
   (c) 95 %.
   (d) 100 %.

5. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The cruciality index CRI($\rho$) of activity B is equal to ...

   (a) 20 %.
   (b) 40 %.
   (c) 50 %.
   (d) 60 %.
   (e) 70 %.
   (f) 80 %.

6. Assume that activity B from the previous question is not deterministic, but has the same distribution as activity A. The cruciality index CRI($\rho$) of activity B is equal to ...

   (a) 20 %.
   (b) 40 %.
   (c) 50 %.
   (d) 60 %.
   (e) 70 %.
   (f) 80 %.

7. Assume a project with two activities A and B in a serial network. Given that both activity A and activity B have a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The cruciality index CRI($\rho$) of activity A is equal to ...

   (a) 25 %.
   (b) 33 %.
   (c) 50 %.
   (d) 66 %.
   (e) 75 %.
   (f) 80 %.

## *RA11: Schedule Sensitivity Index*[10]

This article discusses the use of time sensitivity measures introduced in article "*RA6: Time sensitivity*" on page 126. These measures are the result of a Schedule Risk Analysis (SRA) using Monte Carlo simulations explained in article "*RA5: Schedule sensitivity*" on page 123. More precisely, the Schedule Sensitivity Index

[10]This section is also available as an article entitled "Measuring time sensitivity in a project: The schedule sensitivity index" published online at PM Knowledge Center.

(SSI) calculations are illustrated on a fictitious example project network with eight activities, as displayed in Fig. 7.8 on page 130. This example project has been used in previous articles and contains numbers above each node to denote the baseline duration estimate (in days), resulting in a baseline schedule with a planned duration PD of 90 days and the critical path equal to the activity sequence 1-2-5-8.

### Schedule Sensitivity Index

The Project Management Body Of Knowledge (PMBOK) proposes to combine the activity duration and project duration standard deviations with the Criticality Index. In this article, the following abbreviations will be used:

| | |
|---|---|
| $SAD$ | The simulated activity duration |
| $SPD$ | The simulated project duration |
| $\overline{x}$ | The average value of $x$ |
| $\sigma_x$ | The standard deviation of $x$ |
| CI | Criticality index |

where the abbreviations $SAD$ and $SPD$ are used to stress that all values for the activity durations and the project duration are obtained using Monte Carlo simulation runs.

The Schedule Sensitivity Index can be calculated as follows:

$$\text{SSI} = CI * \frac{\sigma_{SAD}}{\sigma_{SPD}}$$

Table 7.1 on page 131 displays ten simulation runs where each activity has a certain duration that might differ from the original baseline duration estimate given in Fig. 7.8. The simulated project duration ($SPD$) for activities 1–8 per simulation run is also given. Each simulation run reflects a possible real project progress scenario where unexpected changes in the activity estimates might occur.

Based on the simulation runs, the schedule sensitivity index can be calculated with the intermediate calculations illustrated in Table 7.5 for activity 2 of Fig. 7.8. In this example, the values for $(SAD - \overline{SAD})$ and $(SAD - \overline{SAD})^2$ are given for activity 2 with $\overline{SAD} = \text{E}(SAD) = 32$. Likewise, the $SPD - \overline{SPD}$ and $(SPD - \overline{SPD})^2$ are displayed with $\overline{SPD}$ or E($SPD$) the average or expected value of the $SPD$ column of Table 7.1, i.e. $\text{E}(SPD) = 99.1$.

Based on the values in Table 7.5, the standard deviations for the activity duration and project duration can be calculated as follows.

- $\sigma_{SAD} = 11.06$ and is equal to the square root of the sum of the $(SAD - \overline{SAD})^2$ column of Table 7.5 divided by 9.
- $\sigma_{SPD} = 27.65$ and is equal to the square root of the sum of the $(SPD - \overline{SPD})^2$ column of Table 7.5 divided by 9.

**Table 7.5** Intermediate calculations for the SSI of activity 2

| | $(SAD - \overline{SAD})$ | $(SAD - \overline{SAD})^2$ | $(SPD - \overline{SPD})$ | $(SPD - \overline{SPD})^2$ |
|---|---|---|---|---|
| Run 1 | −20 | 400 | −70.1 | 4,914.01 |
| Run 2 | −9 | 81 | 2.9 | 8.41 |
| Run 3 | 6 | 36 | 9.9 | 98.01 |
| Run 4 | −7 | 49 | −11.1 | 123.21 |
| Run 5 | 10 | 100 | 16.9 | 285.61 |
| Run 6 | 12 | 144 | 9.9 | 98.01 |
| Run 7 | −11 | 121 | −0.1 | 0.01 |
| Run 8 | 4 | 16 | −8.1 | 65.61 |
| Run 9 | 3 | 9 | 19.9 | 396.01 |
| Run 10 | 12 | 144 | 29.9 | 894.01 |

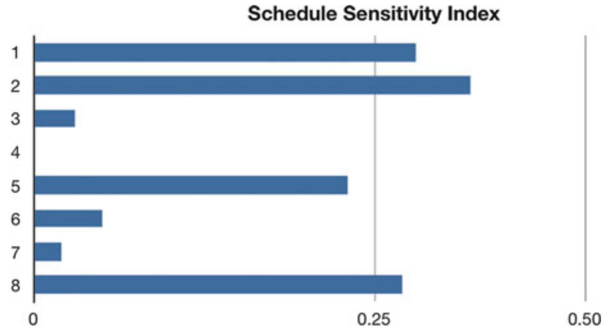



**Fig. 7.13** The sensitivity index values for the eight activities after ten simulation runs

Note that the division by 9 means that the sample standard deviation is calculated and not the population standard deviation. For more information, the reader is referred to any introductory statistical book. The criticality index of activity 2 is equal to CI = 0.8, as illustrated in article "*RA7: Criticality index*" on page 130, and consequently, the schedule sensitivity index is equal to:

$$\mathrm{SSI} = 0.8 * 11.06/27.65 = 0.32.$$

Although activity 2 lies on the critical path in the baseline schedule, the schedule sensitivity index is lower than 1. Figure 7.13 shows the SSI values for all project activities, ranging between 0 and 32 %. More precisely, the values are equal to 0.28, 0.32, 0.03, 0.00, 0.23, 0.05, 0.02 and 0.27 for activities 1–8. Obviously, a schedule risk analysis should be done with care for the following two reasons: (1) the input distributions of the activity duration should reflect reality, and (2) the number of simulation runs should exceed 10 to provide more reliable results.

The graph of Fig. 7.13 can be used to set action thresholds that make a distinction between sensitive and insensitive activities to better steer the project control process during project progress. More information can be found in article "*PC17: Bottom-up control*" on page 251. Apart from the schedule sensitivity index, other time sensitivity measures can also be calculated for the same simulation runs, as discussed in article "*RA6: Time sensitivity*" on page 126.

**Questions**

1. An activity in a serial network (i.e. execute activity 1, then activity 2, then activity 3, ...) will have a schedule sensitivity index value of ...

   (a) SSI $= \sigma_{SPD}/\sigma_{SAD}$.
   (b) SSI $= 0$.
   (c) SSI $= \sigma_{SAD}/\sigma_{SPD}$.
   (d) SSI $= 1$.
   (e) None of the above.

2. An activity in a parallel network (i.e. all activities can be simultaneously executed) will have a schedule sensitivity index value of ...

   (a) SSI $= \sigma_{SPD}/\sigma_{SAD}$.
   (b) SSI $= 0$.
   (c) SSI $= \sigma_{SAD}/\sigma_{SPD}$.
   (d) SSI $= 1$.
   (e) The value can only be calculated using simulations, since this depends on the specific activity duration distributions.

3. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 of page 132. The schedule sensitivity index SSI of activity A is equal to ... (choose closest number)

   (a) 52 %.
   (b) 64 %.
   (c) 78 %.
   (d) 84 %.

4. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 of page 132. The schedule sensitivity index SSI of activity B is equal to ... (choose closest number)

   (a) 52 %.
   (b) 64 %.
   (c) 78 %.
   (d) 84 %.

5. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration $= 10$) $= 0.5$ and P(duration $= 100$) $= 0.5$. The Schedule Sensitivity Index SSI of activity A is equal to ...

   (a) 0 %.
   (b) 24 %.

(c) 48 %.
(d) 56 %.
(e) 64 %.
(f) 100 %.

6. Assume a project with two activities A and B in a serial network. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The Schedule Sensitivity Index SSI of activity A is equal to ...

   (a) 0 %.
   (b) 24 %.
   (c) 48 %.
   (d) 56 %.
   (e) 64 %.
   (f) 100 %.

7. Assume a project with two activities A and B in a serial network. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The Schedule Sensitivity Index SSI of activity B is equal to ...

   (a) 0 %.
   (b) 24 %.
   (c) 48 %.
   (d) 56 %.
   (e) 64 %.
   (f) 100 %.

## *RA12: Significance Index*[11]

This article discusses the use of time sensitivity measures introduced in article "*RA6: Time sensitivity*" on page 126. These measures are the result of a Schedule Risk Analysis (SRA) using Monte Carlo simulations explained in article "*RA5: Schedule sensitivity*" on page 123. More precisely, the Significance Index (SI) calculations are illustrated on a fictitious example project network with eight activities, as displayed in Fig. 7.8 on page 130. This example project has been used in previous articles and contains numbers above each node to denote the baseline duration estimate (in days), resulting in a baseline schedule with a planned duration PD of 90 days and the critical path equal to the activity sequence 1-2-5-8.

[11]This section is also available as an article entitled "Measuring time sensitivity in a project: The significance index" published online at PM Knowledge Center.

**Table 7.6** Ten simulation runs with random activity durations

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | *SPD* | *SSL* |
|---|---|---|---|---|---|---|---|---|---|---|
| Run 1 | 4 | 12 | 1 | 4 | 5 | 4 | 7 | 8 | 29 | 0-0-5-1-0-5-5-0 |
| Run 2 | 22 | 23 | 14 | 5 | 28 | 26 | 5 | 29 | 102 | 0-0-6-23-0-6-6-0 |
| Run 3 | 25 | 38 | 12 | 15 | 24 | 20 | 13 | 22 | 109 | 0-0-17-9-0-17-17-0 |
| Run 4 | 25 | 25 | 15 | 13 | 25 | 25 | 13 | 10 | 88 | 0-3-0-15-3-0-0-0 |
| Run 5 | 21 | 42 | 12 | 7 | 30 | 21 | 14 | 23 | 116 | 0-0-25-23-0-25-25-0 |
| Run 6 | 28 | 44 | 7 | 9 | 15 | 20 | 10 | 22 | 109 | 0-0-22-6-0-22-22-0 |
| Run 7 | 19 | 21 | 14 | 13 | 23 | 24 | 14 | 28 | 99 | 0-8-0-18-8-0-0-0 |
| Run 8 | 12 | 36 | 14 | 9 | 19 | 17 | 7 | 24 | 91 | 0-0-17-10-0-17-17-0 |
| Run 9 | 28 | 35 | 7 | 12 | 28 | 27 | 13 | 28 | 119 | 0-0-16-16-0-16-16-0 |
| Run 10 | 27 | 44 | 6 | 5 | 30 | 10 | 11 | 28 | 129 | 0-0-38-25-0-47-47-0 |

## Significance Index

The Significance Index of a project activity is a measure to reflect this relative importance between the various project activities, and can be calculated as follows:

$$\mathrm{SI} = E\{\frac{SAD}{SAD + SSL} * \frac{SPD}{E(SPD)}\}$$

with

*SAD* The simulated activity duration
*SPD* The simulated project duration
*SSL* The simulated slack of an activity
E{x} Expected value of x

The SI has been defined as a partial answer to the criticism on the criticality index. Rather than expressing an activity's criticality by the probability concept, the SI aims at exposing the significance of individual activities and their impact on the total project duration. In some examples, the SI seems to provide more acceptable information about the relative importance of activities. The abbreviations *SAD* and *SPD* are used to stress that all values for the activity durations and the project duration are obtained using Monte Carlo simulation runs.

Table 7.6 displays ten simulation runs where each activity has a certain duration that might differ from the original baseline duration estimate given in Fig. 7.8. These simulated values are identical to the simulated values of Table 7.1 on page 131. The simulated project duration (*SPD*) and the activity slack (*SSL*) for activities 1–8 per simulation run are also given. Each simulation run reflects a possible real project progress scenario where unexpected changes in the activity estimates might occur.

Based on the simulation runs, the significance index can be calculated with the intermediate calculations illustrated in Table 7.7 for activity 2 of Fig. 7.8. In the example, the values for $SAD * SPD$, $SAD + SSL$ and $(SAD * SPD)/((SAD + SSL) * E(SPD))$ are calculated for activity 2 with E($SPD$) the average of the *SPD* column of

**Table 7.7** Intermediate calculations for the SI of activity 2

| | $SAD * SPD$ | $SAD + SSL$ | $\frac{SAD * SPD}{SAD + SSL} * \frac{1}{E(SPD)}$ |
|---|---|---|---|
| Run 1 | 348 | 12 | 0.29 |
| Run 2 | 2,346 | 23 | 1.03 |
| Run 3 | 4,142 | 38 | 1.10 |
| Run 4 | 2,200 | 28 | 0.79 |
| Run 5 | 4,872 | 42 | 1.17 |
| Run 6 | 4,796 | 44 | 1.10 |
| Run 7 | 2,079 | 29 | 0.72 |
| Run 8 | 3,276 | 36 | 0.92 |
| Run 9 | 4,165 | 35 | 1.20 |
| Run 10 | 5,676 | 44 | 1.30 |

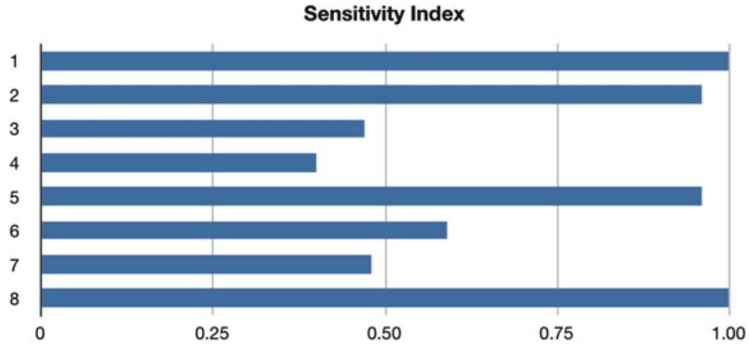



**Fig. 7.14** The significance index values for the eight activities after ten simulation runs

Table 7.6, i.e. E($SPD$) = 99.1. The average of the last column of Table 7.7 is equal to 0.96 and is the sensitivity value of activity 2. Although this activity lies on the critical path in the baseline schedule, the significance index shows that the SI of this activity is somewhat lower than 1. Figure 7.14 shows the SI values for all project activities, ranging between 40 and 100 %. More precisely, the values are equal to 1.00, 0.96, 0.47, 0.40, 0.96, 0.59, 0.48 and 1.00 for activities 1–8. Obviously, a schedule risk analysis should be done with care for the following two reasons: (1) the input distributions on the activity duration should reflect reality, and (2) the number of simulation runs should exceed 10 to provide more reliable results.

The graph of Fig. 7.14 can be used to set action thresholds that make a distinction between sensitive and insensitive activities to better steer the project control process during project progress. More information can be found in article "*PC17: Bottom-up control*" on page 251. Apart from the significance index, other time sensitivity measures can also be calculated for the same simulation runs, as discussed in article "*RA6: Time sensitivity*" on page 126.

## Questions

1. An activity in a serial network (i.e. execute activity 1, then activity 2, then activity 3, ...) will have an significance index value of ...
   (a) SI = 0.
   (b) SI = 1.
   (c) SI = 1/# activities.
   (d) The value can only be calculated using simulations, since this depends on the specific activity duration distributions.
2. An activity in a parallel network (i.e. all activities can be simultaneously executed), will have an significance index value of ...
   (a) SI = 0.
   (b) SI = 1.
   (c) SI = 1/# activities.
   (d) This has to be calculated using simulations, since this depends on the specific activity duration distributions.
3. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The significance index SI of activity A is equal to ... (choose closest number)
   (a) 33 %.
   (b) 50 %.
   (c) 90 %.
   (d) 95 %.
4. Assume a project with two activities A and B in parallel, and a simulation with 4 runs as given in Table 7.2 on page 132. The significance index SI of activity B is equal to ... (choose closest number)
   (a) 33 %.
   (b) 50 %.
   (c) 90 %.
   (d) 95 %.
5. Assume a project with two activities A and B in parallel. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The Significance Index SI of activity B is equal to ...
   (a) 25 %.
   (b) 20 %.
   (c) 33 %.
   (d) 40 %.
   (e) 66 %.

6. Assume a project with two activities A and B in a serial network. Given that activity B has a deterministic duration of 20 days, and activity A has a stochastic duration with the following probabilities: P(duration = 10) = 0.5 and P(duration = 100) = 0.5. The Significance Index SI of activity B is equal to ...

   (a) 25 %.
   (b) 33 %.
   (c) 40 %.
   (d) 66 %.
   (e) 100 %.

# Chapter 8
# Buffer Management

## RA13: Schedule Protection[1]

The Critical Chain/Buffer Management (CC/BM) approach originally introduced by E. Goldratt in his novel "Critical Chain", written in 1997 as an application of the "Theory of Constraints", is a project management technique that introduces some novel ideas on top of the well-known and generally accepted resource constrained project scheduling principles. It introduces the idea of buffering projects, rather than individual activities, in a clever way in order to guarantee that the project deadline is met, and can therefore be considered as a new approach to assure better project control through project and feeding buffer management.

This article gives an overview of the Critical Chain/Buffer Management (CC/BM) approach of project scheduling and will be briefly explained along the following six steps:

- Step 1: Come up with aggressive estimates
- Step 2: Construct an as late as possible (ALAP) schedule
- Step 3: Identify the critical chain
- Step 4: Determine appropriate buffer positions
- Step 5: Determine appropriate buffer sizes
- Step 6: Insert buffers into the schedule

Details on the various CC/BM steps will be discussed in other articles in this book for which a reference is given in between the different steps.

[1]This section is also available as an article entitled "Critical Chain/Buffer Management: Protecting the schedule against project delays" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_8

### *Step 1: Come Up with Aggressive Estimates*

Based on the general idea that the protection of the project deadline is the primary goal rather than the protection of each individual project activity, the activity durations should be set to an aggressive estimate to avoid that the work of an activity is smoothed out over a longer duration than really necessary. Goldratt mentions three main reasons why project activity durations might be smoothed out over longer durations, namely:

- Murphy's law: If something can go wrong, it will.
- Student syndrome: Wait until urgency.
- Parkinson's law: Work expands to fill the allotted time.

Aggressive time estimates refer to time estimates where the probability of exceedance is relatively large, e.g. the 50 % percentile. More details on these time estimates are given in article "*RA14: Aggressive estimates*" on page 158.

### *Step 2: Construct an ALAP Schedule*

In contrast to the commonly used earliest start scheduling approach mostly used in traditional project management and scheduling (see article "*BS10: Activity slack*" on page 43), it is suggested to schedule each project activity as late as possible nearby its predefined project deadline. This latest start schedule is quite risky since it makes every project activity part of the critical path, which might put, in combination with the aggressive activity time estimates, the project deadline into danger. However, this approach also offers certain advantages that are illustrated and discussed in article "*RA15: Latest start schedules*" on page 161.

### *Step 3: Identify the Critical Chain*

Similar to the critical path concept that distinguishes between critical and noncritical activities, a resource feasible project schedule has a so-called critical chain defined as the longest chain in the project that determines the total project duration. The main difference with the critical path concept is that it takes the limited availability of renewable resources into account (see article "*BS7: Critical path/chain*" on page 31).

### *Step 4: Determine Appropriate Buffer Positions*

The new idea of the CC/BM technique lies in the clever positioning of buffers to protect the project deadline through the use of three types of buffers. Each buffer has a specific goal and needs to be positioned at the right place to protect the right part of the project data.

- Project buffer: A unique and single buffer to protect the project deadline.
- Feeding buffer: Multiple buffers to protect parts of the critical chain.
- Resource buffer: Multiple artificial buffers that act as warning signals to assure the availability of resources.

More information and an illustration of the positioning of buffers can be found in article "*RA16: Buffering*" on page 166.

## *Step 5: Determine Appropriate Buffer Sizes*

Sizing buffers as a way to put safety time in the project is a crucial step of the CC/BM approach. Both the project and all the feeding buffers need to be sized appropriately to be able to act as buffers against delay. Although Goldratt initially proposed to use the simple 50 % rule (i.e. the size of the buffers is equal to 50 % of the duration of the project activities it should protect), a more clever way should be used that sizes buffers according to the risk of the activities it should protect. More information on buffer size techniques can be found in article "*RA17: Sizing buffers*" on page 169.

## *Step 6: Insert Buffers into the Schedule*

Inserting buffers into the project schedule creates a buffered project baseline schedule that can act as a tool to measure performance and provides dashboards (i.e. the buffers) that need to be monitored to trigger corrective actions. More information can be found in article "*RA22: Buffer insertion*" on page 186.

### Questions

1. The Critical Chain/Buffer Management (CC/BM) methodology focuses on . . .
   (a) controlling the cost of a project.
   (b) controlling the duration of a project.
   (c) controlling both the cost and duration of a project.
2. The CC/BM methodology takes into account the availability of scarce resources.
   (a) True
   (b) False

3. Scheduling a project using the CC/BM methodology happens in an . . . way.
   (a) ASAP (= as soon as possible)
   (b) ALAP (= as late as possible)
4. The critical chain is a synonym for the critical path.
   (a) True
   (b) False
5. The CC/BM methodology can be associated with . . .
   (a) top-down project management.
   (b) bottom-up project management.

   *Note*: Bottom-up and top-down project management and control are explained in articles "*PC17: Bottom-up control*" on page 251 and "*PC18: Top-down control*" on page 255.

## RA14: Aggressive Estimates[2]

Estimates for activity duration in the Critical Chain/Buffer Management (CC/BM) approach are assumed to be aggressive estimates and refer to the distinction between the average estimate and the low-risk estimate of an activity duration. It is a reaction to the commonly made assumption of adding safety time in the activity time estimates as is usually done in practice, which might lead to quite unrealistic project deadlines.

This article compares the inflated activity time estimates and the aggressive time estimates, as follows:

- Inflating activity time estimates: Unrealistic project deadlines and self-fulfilling prophecies.
- Aggressive time estimates: Removal of activity protection to obtain average or median time values.

### *Inflating Time Estimates*

Most people have the natural tendency to inflate the time estimates for project activities. Indeed, when people are asked to give a reasonable time estimate for an activity they have to perform, they tend to give an estimate they are comfortable

[2]This section is also available as an article entitled "Aggressive activity time estimates: Protecting against activity delays" published online at PM Knowledge Center.

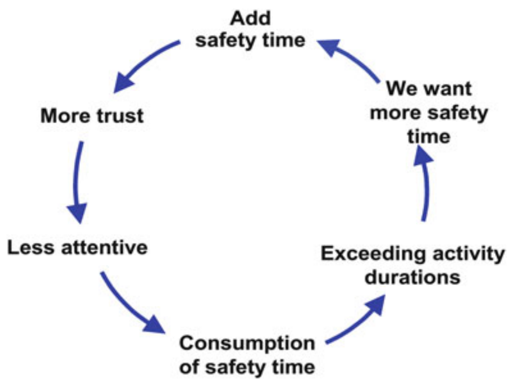



**Fig. 8.1** The self-fulfilling prophecy circle caused by comfortable time estimates

with. In this respect, the activity time estimates are rather risk-free and contain safety time as a kind of protection.

However, inflating the activity time estimates leads to self-fulfilling prophecies, and consequently, the initial idea of having a comfortable level safety time in the activity time estimates quickly fades away, as illustrated in Fig. 8.1.

The self-fulfilling prophecy circle is caused by a natural behavior that most people have, and can be summarized as follows:

- Student syndrome: When safety time is added in the activity time estimates, people have the tendency to wait until urgency (i.e. more trust and less attentive), similar to the behavior of students that postpone their work until exam deadlines.
- Parkinson's law: When safety time is added in our estimates, we feel more comfortable and have no incentive to speed up our work. Consequently, the work expands to fill the allotted time (consumption of safety time), known as the law of Parkinson.
- Murphy's law: If something can go wrong, it will (exceeding activity durations), which leads to activity durations that are even longer than their safe and risk-free time estimates. Hence, people tend to believe that they should have added even more safety time to their estimates.

## *Aggressive Time Estimates*

Based on the general idea that the protection of the project deadline is the primary goal rather than the protection of each individual project activity, the activity durations should be set to an aggressive estimate to avoid that the work of an activity is smoothed out over a longer duration than really necessary, as shown in Fig. 8.1. Since activity durations are stochastic by nature and it is therefore impossible to estimate the correct deterministic duration, it is generally conjectured that the average activity time estimates are on the order of one-half to one-third of

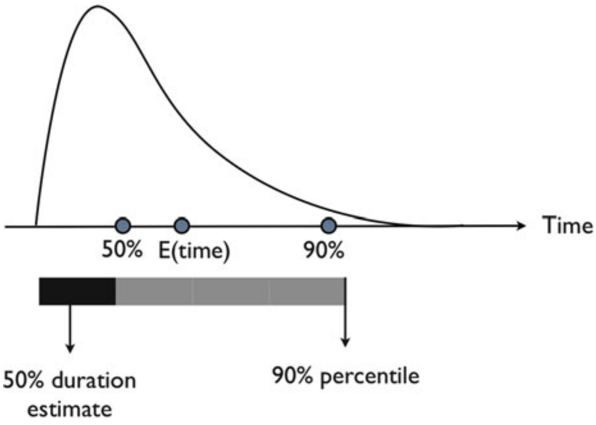



**Fig. 8.2** Aggressive and risk-free time estimate for a project activity

the low-risk estimates. Consequently, the safety time (i.e. the grey zone displayed in Fig. 8.2) should be removed from the activity time estimates to obtain aggressive time estimates, e.g. the 50 % percentile activity duration estimate or the average duration.

These aggressive time estimates (black bar of Fig. 8.2) should be used as inputs for the construction of a project baseline schedule in the critical chain/buffer management approach. The safety lost due to the use of aggressive estimates will be placed elsewhere in the project as project and feeding buffers. More information on the use of safety time incorporated in buffers can be found in article "*RA16: Buffering*" on page 166. General information on the critical chain/buffer management approach is written in article "*RA13: Schedule protection*" on page 155.

## Questions

1. Which of these definitions describes Parkinson's law?
   (a) If something can go wrong, it will (exceeding activity durations), which leads to activity durations that are even longer than their safe and risk-free time estimates.
   (b) When safety time is added in the activity time estimates, people have the tendency to wait until urgency.
   (c) When safety time is added in our estimates, we feel more comfortable and have no incentive to speed up our work. Consequently, the work expands to fill the allotted time.

2. Which of these definitions describes the student syndrome?
   (a) If something can go wrong, it will (exceeding activity durations), which leads to activity durations that are even longer than their safe and risk-free time estimates.
   (b) When safety time is added in the activity time estimates, people have the tendency to wait until urgency.
   (c) When safety time is added in our estimates, we feel more comfortable and have no incentive to speed up our work. Consequently, the work expands to fill the allotted time.

3. The aggressive estimate strategy implicitly assumes that the true distribution behind activity durations is . . .
   (a) skewed to the left.
   (b) symmetrical.
   (c) skewed to the right.

4. Since human beings tend to buffer their estimates for activity durations, it is very unlikely for any of these activities to be delivered late.
   (a) True
   (b) False

## RA15: Latest Start Schedules[3]

During the construction of a project baseline schedule in the Critical Chain/Buffer Management (CC/BM) approach, activities are scheduled as late as possible towards the predefined project deadline. In this article, the contrast between the traditional resource constrained project scheduling approach and the CC/BM scheduling approach is discussed along the following dimensions:

- Traditional resource constrained scheduling: Earliest start schedule based on risk-free activity duration estimates.
- Critical chain scheduling: Latest start schedule based on aggressive activity duration estimates.

[3]This section is also available as an article entitled "Critical Chain/Buffer Management: (Dis)advantages of scheduling projects as late as possible" published online at PM Knowledge Center.

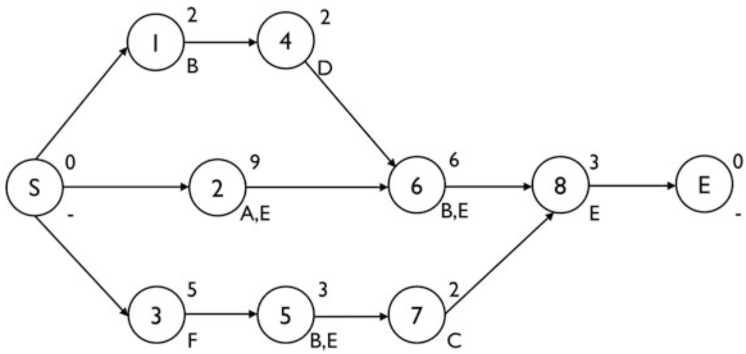



**Fig. 8.3** Project network data with normal activity durations

## *Traditional Resource Constrained Scheduling*

In the PERT/CPM scheduling approach, renewable resources are ignored and the critical path can be easily detected by scheduling all activities as soon as possible, leading to an earliest start schedule (ESS) (see article "*BS10: Activity slack*" on page 43). When renewable resources are taken into account, their limited availability might cause resource conflicts that have to be resolved (see article "*BS7: Critical path/chain*" on page 31). This can be done by working forwards from the project start date, gradually resolving resource conflicts using priority rule based scheduling techniques or other scheduling techniques (see article "*BS17: Priority rule based scheduling*" on page 66).

Figure 8.3 displays a project network with eight activities. The numbers above each node are used to refer to the activity duration estimate while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units.

A resource feasible schedule of the project network of Fig. 8.3 is easy to construct, since a simple earliest start schedule shows no resource conflict, and hence, the critical path 2-6-8 is equal to the critical chain. However, it is shown in other articles that the construction of a resource feasible schedule is often more complex and the critical chain is often different from the critical path, and depends on the scheduling objective (see article "*BS22: What is my scheduling objective?*" on page 88). The resource feasible earliest start schedule is displayed in Fig. 8.4.

## *Critical Chain Scheduling*

The CC/BM scheduling approach differs from the traditional resource constrained scheduling approach in the use of aggressive time estimates and the construction

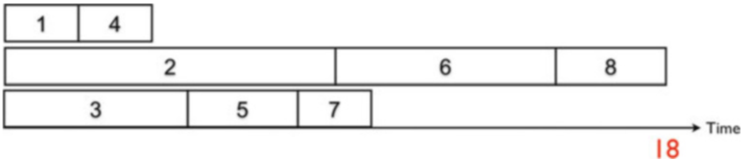



**Fig. 8.4** A resource feasible earliest start schedule (CC = 2-6-8) of the project in Fig. 8.3

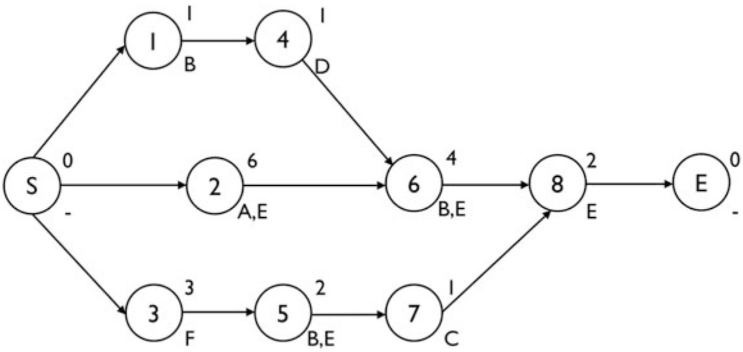
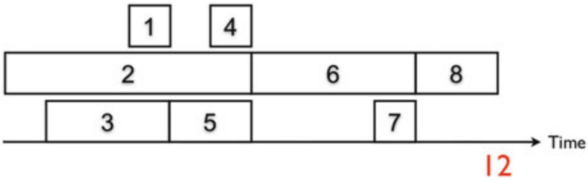



**Fig. 8.5** Project network data with aggressive activity durations

**Fig. 8.6** A resource feasible latest start schedule (CC = 2-6-8) of the project of Fig. 8.5




of latest start schedules (LSS). In the article "*RA14: Aggressive estimates*" on page 158, it has been shown that the use of aggressive time estimates is proposed to avoid the protection of the individual activities of the project. Instead, the risk-free estimates should be replaced by more risky estimates. In this article, it is assumed that the aggressive time estimates are equal to two thirds of their risk-free estimates (rounded down in case of fractional durations). This results in the project network displayed in Fig. 8.5.

In the CC/BM approach, schedules are constructed by putting activities as late as possible by working backwards from the project deadline and gradually resolving resource conflicts. This results in a resource feasible latest start schedule displayed in Fig. 8.6. Activity 5 has been pushed before activity 6 since it cannot be scheduled in parallel with activity 6 (due to their common use of resource B). Likewise, activity 1 has been pushed before activity 5 due to the resource conflict for resource B between activities 1 and 5.

The construction of a resource feasible latest start schedule has a number of advantages in comparison with the earliest start scheduling approach, namely:

- Latest start schedules tend to minimize the total work-in-progress. In this respect, it can be said that the construction of a LSS creates a more leveled resource workload than an ESS. More information on resource leveling can be found in article "*BS27: Resource leveling*" on page 103.
- Latest start schedules automatically delay important cash outflows. Consequently, the construction of a LSS contributes to the maximization of the net present value of a project, as discussed in article "*BS25: Net present value*" on page 96.
- Latest start schedules often lead to less rework. Indeed, delaying activities towards the project end postpones a portion of the uncertainty, and hence, many activities can be performed when the uncertain information is already available.

However, despite these advantages, the construction of a latest start schedule has one major disadvantage, which might endanger the project deadline. Since all activities are scheduled as late as possible, none of the activities have a positive slack value, and hence, all activities are critical. This disadvantage, in combination with the use of aggressive time estimates, might jeopardize the project deadline and will probably lead to project deadline violations. However, the CC/BM approach will add buffers to the latest start schedule to protect the project deadline against the aforementioned threat. The schedule of Fig. 8.5 will be used to construct a buffered schedule for which more general information is given in article "*RA13: Schedule protection*" on page 155.

## Questions

In the following questions, the abbreviation ASAP is used for "as soon as possible" while ALAP is used for "as late as possible".

1. Which of the following statements is true?
   (a) ALAP scheduling generally increases the work in progress, ensuring a faster delivery of the project since more people are working simultaneously.
   (b) ALAP scheduling increases the speed at which cash inflows are received.
   (c) When using the ALAP scheduling method, more rework is often required.
   (d) ALAP scheduling delays important cash outflows.

**Table 8.1** Example project data

| Activity | Duration (weeks) | Resources | Successors |
|---|---|---|---|
| 1 | 2 | A | 2,3 |
| 2 | 1 | B | 4 |
| 3 | 4 | B | 5 |
| 4 | 2 | B | 5 |
| 5 | 4 | A | – |

2. Projects with activities scheduled ALAP have . . .
   (a) more slack than projects scheduled using the ASAP method.
   (b) an equal amount of slack as projects scheduled using the ASAP method.
   (c) less slack than projects scheduled using the ASAP method.
3. Consider the project data in Table 8.1 on page 165. What is the start time of activity 4 using an ASAP scheduling approach, when resource constraints are ignored (assume the project starts in week 1)?
   (a) Week 2
   (b) Week 3
   (c) Week 4
   (d) Week 5
   (e) Week 6
4. Consider the project data in Table 8.1 on page 165. What is the start time of activity 4 using an ALAP scheduling approach, when resource constraints are ignored?
   (a) Week 2
   (b) Week 3
   (c) Week 4
   (d) Week 5
   (e) Week 6
5. Consider the project data in Table 8.1 on page 165. What is the project duration using an ALAP scheduling approach, when the availability of each resource is assumed to be equal to one unit?
   (a) 6 weeks
   (b) 8 weeks
   (c) 10 weeks
   (d) 13 weeks
6. When switching from an ASAP to an ALAP scheduling approach, the total project duration will . . .
   (a) increase.
   (b) remain the same.
   (c) decrease.

## RA16: Buffering[4]

The Critical Chain/Buffer Management (CC/BM) approach aims at the construction of latest start schedules where the project activities use aggressive time estimates and puts a clear focus on the determination of a realistic project deadline. In order to keep the probability high that the project deadline will be met, the CC/BM approach protects the project duration and the critical chain of the project using various buffers. This buffered scheduling approach is the topic of this article and is discussed along the following dimensions:

- Project buffer: A unique and single buffer to protect the project deadline.
- Feeding buffer: Multiple buffers to protect parts of the critical chain.
- Resource buffer: Multiple artificial buffers that act as warning signals to assure the availability of resources.

Figure 8.7 displays a project network with eight activities and a latest start schedule with a project duration of 12 time units. The numbers above each node in the network are used to refer to the aggressive activity duration estimate (see article "*RA14: Aggressive estimates*" on page 158) while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The construction of the resource feasible latest start schedule is discussed in article "*RA15: Latest start schedules*" on page 161.

The CC/BM approach assumes that meeting the project deadline is the scheduling objective (see article "*BS22: What is my scheduling objective?*" on page 88), and hence, the use of buffers is suggested to guarantee that the project is completed on or before the predefined project deadline with a high probability.

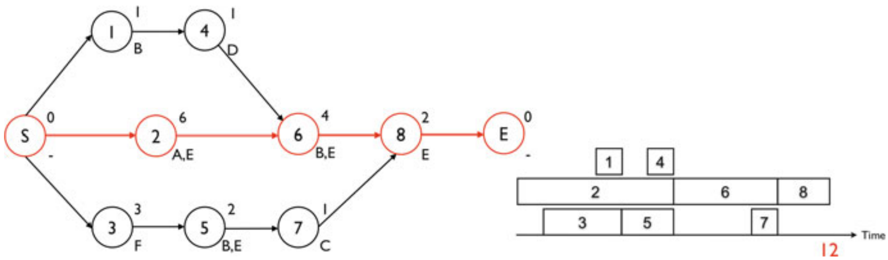



**Fig. 8.7** An example project network and a resource feasible latest start schedule

[4]This section is also available as an article entitled "Critical Chain/Buffer Management: Adding buffers to a project schedule" published online at PM Knowledge Center.

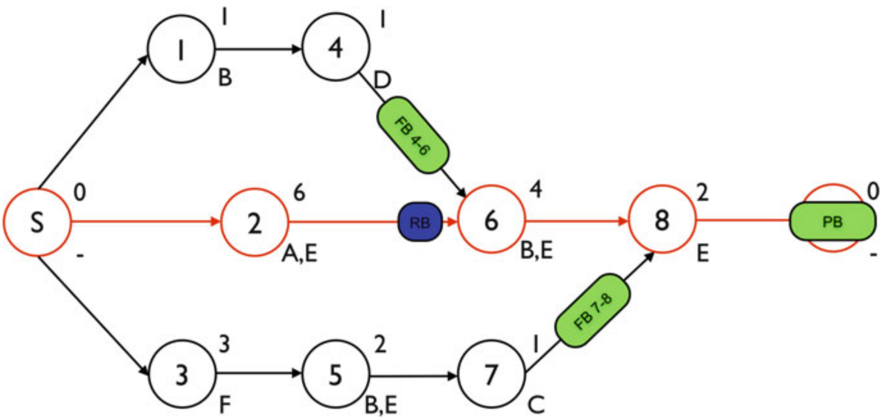



**Fig. 8.8** The project network of Fig. 8.7 with feeding, resource and project buffers

## *Project Buffer*

> A project buffer protects the project deadline against violations in the critical chain.

A single project buffer is added at the end of the project network between the last activity and the project deadline. Any delays on the critical chain will partly consume this buffer without having an effect on the project completion date. Consequently, a project buffer acts as a protection of the project completion date, which might be variable due to changes in activity durations on the critical chain. Its size should depend on the expected changes and variability of the activities on the critical chain. Figure 8.8 displays the unique project buffer at the end of the project (final node E).

## *Feeding Buffers*

> A feeding buffer protects the critical chain against violations in the feeding chain.

Any path of activities merging into the critical chain is called a feeding chain. In the example project of Fig. 8.7, feeding chains 1-4 and 3-5-7 merge into the critical chain 2-6-8 through the links (4-6) and (7,8), respectively. Since the project buffer is inserted in the project schedule as a protection of the project deadline against changes in the critical chain, the critical chain should also be protected against changes in any feeding chain. Consequently, similar to the project buffer between the last activity of the critical chain and the project completion date, feeding buffers between any last activity of a feeding chain and the activity on the critical chain will be added to the project baseline schedule. This results in feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ in the project network of Fig. 8.8.

## *Resource Buffers*

> A resource buffer acts as a warning signal when a shift in resources will occur on the critical chain.

While both the project buffer and feeding buffers act as mechanisms to transpose (part of) the removed safety time of the individual activities to safety time buffers, resource buffers act as warning signals that ensure the timely availability of project resources. More precisely, resource buffers can be set alongside of the critical chain to ensure that the renewable resources are available to work on the critical chain activities as soon as they are needed. Consequently, a resource buffer warning signal is added each time an activity needs a renewable resource that is not used by the previous activity. In Fig. 8.8, a resource buffer is added between activities 2 and 6 to ensure that resource B is available upon the start of activity 6.

Figure 8.8 shows the buffered network of Fig. 8.7 with one project buffer, two feeding buffers and one resource buffer. The size of the project and feeding buffers depends on the project characteristics and is discussed in article "*RA17: Sizing buffers*" on page 169.

### Questions

1. Which of these buffers is not a part of the Critical Chain/Buffer Management (CC/BM) methodology?
   (a) Project buffer
   (b) Feeding buffer
   (c) Cost buffer
   (d) Resource buffer
2. Whenever a part of the project buffer is consumed, the project manager should immediately take into account that the complete project is likely to be delivered past the defined deadline.
   (a) True
   (b) False

3. The size of the project buffer should ideally be determined by . . .
   (a) the variability of all the activities.
   (b) the variability of the activities on the critical chain.
   (c) the duration and variability of all the activities.
4. Feeding buffers are added to . . .
   (a) signal potential cost overruns to the project manager.
   (b) guarantee the availability of resources.
   (c) ensure that the deadline is not violated in case there are unexpected delays in the critical chain.
   (d) protect the critical chain from delays in feeding chains.
5. Since resource buffers are always added on the critical chain, the addition of more resource buffers always causes a delay of the (expected) project duration.
   (a) True
   (b) False

## Sizing CC/BM Buffers

### *RA17: Sizing Buffers*[5]

The Critical Chain/Buffer Management (CC/BM) approach uses project and feeding buffers to add safety time to the project baseline schedule and to guarantee the timely completion of the project with a high probability. Buffers are sized according to the properties of the path or chain feeding those buffers, such as the length of the path, its total variance, its average resource use or the number of activities it contains. In this article, a nonexhaustive overview of buffer sizing methods is given. Moreover, an example project is given in Fig. 8.8 on page 167, which is based on other CC/BM articles and will be used to illustrate the different buffer sizing methods in detail in other articles.

Figure 8.8 displays a project network with eight activities. The numbers above each node are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6.

[5]This section is also available as an article entitled “Critical Chain/Buffer Management: Sizing project and feeding buffers” published online at PM Knowledge Center.

**Table 8.2** Summary of project data

| Activity | Safe duration | Aggressive duration | Delta duration | Standard deviation |
|---|---|---|---|---|
| 1 | 2 | 1 | 1 | 0.5 |
| 2 | 9 | 6 | 3 | 1.5 |
| 3 | 5 | 3 | 2 | 1.0 |
| 4 | 2 | 1 | 1 | 0.5 |
| 5 | 3 | 2 | 1 | 0.5 |
| 6 | 6 | 4 | 2 | 1.0 |
| 7 | 2 | 1 | 1 | 0.5 |
| 8 | 3 | 2 | 1 | 0.5 |

More information on the determination of these buffers can be found in the article "*RA16: Buffering*" on page 166.

Table 8.2 is an overview table that displays the safe and aggressive durations of all project activities based on the example given in the article "*RA15: Latest start schedules*" on page 161. The difference between the safe and aggressive time estimates is also indicated in the column with label 'delta duration'. The use of aggressive time estimates is discussed in the article "*RA14: Aggressive estimates*" on page 158.

In Table 8.2, the standard deviations of the activities are assumed to be 50 % of the difference between the safe and aggressive durations. Although this has been suggested in literature, obviously other values can be set based on more detailed risk information and known or estimated distributions of activity durations. For more information on using distributions of activity parameters, see article "*RA4: Activity distributions*" on page 119. Alternatively, the standard deviation can also be based on three point estimates as is the case in the PERT method (see article "*BS15: PERT*" on page 59).

Given the assumed values for the activity standard deviations, the standard deviation $\sigma_{path}$ of a path is then equal to the square root of the sum of the variances of the activities on that path. In this case, it is assumed that the activity finish times are independent (an assumption that has also been taken by the PERT technique). As an example, the standard deviation of the critical chain is equal to:

- Critical chain 2-6-8: $\sqrt{1.5^2 + 1^2 + 0.5^2} = 1.87$
- Feeding chain 1-4: $\sqrt{0.5^2 + 0.5^2} = 0.71$
- Feeding chain 3-5-7: $\sqrt{1^2 + 0.5^2 + 0.5^2} = 1.22$

In the list below, an overview is given of five methods to determine the size of the buffers. Each method is briefly described and a reference to a more detailed example is given. In these referenced articles, the network of Fig. 8.8 and the values of Table 8.2 will be used as inputs.

- Cut and paste method: Buffer sizes are based on the duration of the chain feeding the buffer (Article "*RA18: Cut and paste method*" on page 172).
- Root Squared Error Method: Buffer sizes are based on risk in the activity durations (Article "*RA19: Root squared error method*" on page 174).

- Adaptive procedure with density (APD): Buffer sizes are based on the structure of the partial network to which the chain belongs (Article "*RA20: Adaptive density method*" on page 177).
- Adaptive procedure with resource tightness (APRT): Buffer sizes are based on the average resource use of the resources used by the activities on the longest path of the chain (Article "*RA21: Adaptive resource tightness method*" on page 182).

## Questions

For the ease of the calculations, it is assumed that the project used for the questions below does not use any resources. Despite the inherent presence of resource constraints in the Critical Chain/Buffer Management (CC/BM) methodology, projects without resources can also be buffered using exactly the same CC/BM principle. The only difference is that the critical chain is now identical to the critical path.

1. The project buffer of the CC/BM methodology takes into account the potential variability of . . .
   (a) all the activities of the project.
   (b) only the activities in the critical chain.
   (c) all the activities in the feeding chains.
2. Consider the project data given in Table 8.3 on page 172. Which of these activities is not on the critical chain using the CC/BM methodology?
   (a) 1
   (b) 3
   (c) 4
   (d) 5
3. What is the variance of the critical chain for the project data given in Table 8.3?
   (a) 2.5
   (b) 2.6
   (c) 3.5
   (d) 5.0
   (e) 6.5
   (f) 7.0
   (g) None of the above.
4. What is the variance of the feeding chain for the project data given in Table 8.3?
   (a) 1.4
   (b) 2.0
   (c) 2.5
   (d) 6.5

**Table 8.3** Example project data

| Activity | Safe duration | Aggressive duration | Delta duration | Standard deviation | Successors |
|---|---|---|---|---|---|
| 1 | 6 | 4 | 2 | 1 | 2,3 |
| 2 | 5 | 3 | 2 | 1 | 4 |
| 3 | 10 | 6 | 4 | 2 | 5 |
| 4 | 5 | 3 | 2 | 1 | 6 |
| 5 | 2 | 1 | 1 | 0.5 | 6 |
| 6 | 2 | 1 | 1 | 0.5 | 7 |
| 7 | 6 | 4 | 2 | 1 | – |

(e) 7.0
(f) None of the above.

5. What is the standard deviation of the feeding chain for the project data given in Table 8.3?

(a) 1.00
(b) 1.41
(c) 1.62
(d) 2.00
(e) None of the above.

## *RA18: Cut and Paste Method*[6]

This article discusses a buffer sizing method as discussed in the summary article "*RA17: Sizing buffers* on page 169. More precisely, it describes the cut and paste method, which is probably the simplest method to determine the size of the project and feeding buffers and is solely based on the activity duration of the chain merging into the buffer.

Figure 8.8 on page 167 displays a project network with eight activities. The numbers above each node are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6. More information on the determination of these buffers can be found in the article "*RA16: Buffering*" on page 166.

[6]This section is also available as an article entitled "Sizing CC/BM buffers: The cut and paste method" published online at PM Knowledge Center.

## Cut and Paste Method

The first and probably most simplified method to size buffers has been proposed by Eliyahu M. Goldratt in his book "Critical Chain" (1997) and has been labelled by other authors as the "cut and paste method" or the "50 % of the chain method".

| |
|---|
| buffer size<br>=<br>half of the duration of the longest path feeding the buffer |

The buffer is sized as half of the duration of the longest path feeding the buffer and consequently is equal to the following values:

- Project buffer: 50 % of $(6 + 4 + 2) = 6$
- $FB_{4\text{-}6}$: 50 % of $(1 + 1) = 1$
- $FB_{7\text{-}8}$: 50 % of $(3 + 2 + 1) = 3$

Although this sizing method has the advantage of being easy to calculate, it suffers from oversimplification and should therefore be used with care. The presence of activities with long durations will lead to overly large buffers and project durations. Moreover, this method does not take variability of the activities into account, which is often the reason why it is difficult to justify the use of this method. An overview of other buffer sizing methods as an attempt to overcome the simplicity and resulting weaknesses of the cut and paste method is available in the article "*RA17: Sizing buffers*" on page 169.

## Questions

For the ease of the calculations, it is assumed that the project used for the questions below does not use any resources. Despite the inherent presence of resource constraints in the Critical Chain/Buffer Management (CC/BM) methodology, projects without resources can also be buffered using exactly the same CC/BM principle. The only difference is that the critical chain is now identical to the critical path.

1. Consider the project data given in Table 8.3 on page 172. What is the length of the project buffer when calculated with the cut and paste method using the CC/BM methodology?

   (a) 8
   (b) 11
   (c) 13
   (d) 16
   (e) 18
   (f) 26

2. What is the size of the feeding buffer when calculated with the cut and paste method for the project data given in Table 8.3?
   (a) 2
   (b) 3
   (c) 6
   (d) 8
   (e) 11
3. When a project has typically many activities with short durations but with a large variability (long tail), the cut and paste method is likely to ...
   (a) overestimate the needed buffer.
   (b) be approximately adequate.
   (c) underestimate the needed buffer.
4. The coefficient of variation is the ratio of the standard deviation to the average duration of a distribution. Assume that the majority of the activities in a project has a large coefficient of variation. The cut and paste method will then typically ...
   (a) overestimate the needed buffer.
   (b) be approximately adequate.
   (c) underestimate the needed buffer.
5. Assuming that an activity with a duration equal to half of the expected duration of the critical chain is fully deterministic (i.e. its variance equals zero.) The cut and paste method will then typically ...
   (a) overestimate the needed buffer.
   (b) be approximately adequate.
   (c) underestimate the needed buffer.

## *RA19: Root Squared Error Method*[7]

This article discusses a buffer sizing method as discussed in the summary article "*RA17: Sizing buffers* on page 169. More precisely, it describes the Root Squared Error Method (RSEM) to determine the size of the project and feeding buffers and is based on the predefined uncertainty of the activity durations of the chain merging into the buffer.

Figure 8.8 on page 167 displays a project network with eight activities. The numbers above each node are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform

[7]This section is also available as an article entitled "Sizing CC/BM buffers: The root squared error method" published online at PM Knowledge Center.

the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6. More information on the determination of these buffers can be found in the article "*RA16: Buffering*" on page 166.

Table 8.2 displays the safe and aggressive time estimates of all project activities based on the example given in the article "*RA15: Latest start schedules*" on page 161.

## Root Squared Error Method

Rather than using a simple rule-of-thumb to size project and feeding buffers, the sum of squares method accounts for known variation in activity durations. This method is often referred to as the sum of squares method since it takes the sum of the squared differences between the low risk duration and the aggressive duration into account.

| |
|---|
| buffer size<br>=<br>the square root of the sum of the squares of the difference between the low risk duration and the aggressive duration for each activity along the chain leading to the buffer |

In this method, the buffers are sized as the square root of the sum of the squares of the difference between the low risk duration and the mean duration for each activity along the chain leading to the buffer, leading to the following values:

- project buffer: $\sqrt{(9-6)^2+(6-4)^2+(3-2)^2} = \sqrt{14} = 3.7$
- $FB_{4\text{-}6}$: $\sqrt{(2-1)^2+(2-1)^2} = \sqrt{2} = 1.4$
- $FB_{7\text{-}8}$: $\sqrt{(5-3)^2+(3-2)^2+(2-1)^2} = \sqrt{6} = 2.4$

## Rationale Behind this Method

It should be noted that the general underlying idea of this buffer method is that the size of these buffers is equal to two times the standard deviation of the path that merges into the buffer. Consequently, this buffer sizing rule depends on how the standard deviations of activities are defined.

In the article "*RA17: Sizing buffers*" on page 169, it has been assumed that the standard deviation of an activity is equal to 50 % of the difference between the safe and aggressive durations. Assume that $\delta_i$ is used to denote the difference between the safe and aggressive time estimate of activity $i$. The standard deviation of this activity is then equal to $(\delta_i/2)$. As an example, $\delta_2$ is equal to $(9-6)$ and the standard deviation of activity 2 is then equal to $\sigma_2 = (9-6)/2 = 1.5$.

Assume that the chain consists of $n$ activities $1, 2, \ldots, n$. Since the buffer is sized as two times the path standard deviation, the buffer size is equal to:

$$\begin{aligned} 2 * \sigma_{path} &= 2 * \sqrt{(\delta_1/2)^2 + (\delta_2/2)^2 + \ldots + (\delta_n/2)^2} \\ &= \sqrt{(\delta_1)^2 + (\delta_2)^2 + \ldots + (\delta_n)^2} \end{aligned}$$

which boils down to the buffer sizing rules used by the RSEM. As an example, the project buffer is sized as $2 * \sqrt{1.5^2 + 1^2 + 0.5^2} = 3.7$.

However, when other calculations are used to define the activity standard deviations, the RSEM should be defined differently. As an example, the standard deviation can also be based on three point estimates as is the case in the PERT method (see article "*BS15: PERT*" on page 59). Therefore, the general rule to size the buffers is to set them equal to twice the path standard deviation.

| |
|---|
| buffer size<br>=<br>two times the standard deviation of the longest path feeding the buffer |

It should be noted that the advantage of the RSEM is that it takes known variation in the activity duration into account, and it can therefore be considered as a more elaborated technique than the simple 50 % rule (see article "*RA18: Cut and paste method*" on page 172). However, this technique has also been criticized in literature since it may lead to undersized buffers when chains are relatively long due to the underestimation of activity variation. To that purpose, a variant of the method, known as the "bias plus sum of the squares" method has been proposed, which combines the 50 % rule and the RSEM. More details are outside the scope of this article. An overview of other buffer sizing methods is available in the article "*RA17: Sizing buffers*" on page 169.

## Questions

For the ease of the calculations, it is assumed that the project used for the questions below does not use any resources. Despite the inherent presence of resource constraints in the Critical Chain/Buffer Management (CC/BM) methodology, projects without resources can also be buffered using exactly the same CC/BM principle. The only difference is that the critical chain is now identical to the critical path.

1. Consider the project data given in Table 8.4 on page 177. What is the length of the project buffer when calculated with the root squared error method using the CC/BM methodology?

   (a) 4.9
   (b) 5.1

**Table 8.4** Example project data

| Activity | Safe duration | Aggressive duration | Successors |
|---|---|---|---|
| 1 | 6 | 4 | 2,3 |
| 2 | 5 | 3 | 4 |
| 3 | 10 | 6 | 5 |
| 4 | 5 | 3 | 6 |
| 5 | 2 | 1 | 6 |
| 6 | 2 | 1 | 7 |
| 7 | 6 | 4 | – |

(c) 14.6
(d) 26.0
(e) 26.4

2. What is the size of the feeding buffer when calculated with the root squared error method for the project data given in Table 8.4 on page 177?

(a) 1.41
(b) 2.30
(c) 2.82
(d) 7.51
(e) 8.00

3. The root squared error method can potentially lead to errors when . . .

(a) the chains are relatively short.
(b) the chains are relatively long.

4. The root squared error method assumes that the buffer size equals two times the standard deviation of the longest path that feeds the buffer. What is the likelihood that the project buffer will be exceeded, assuming that this assumption is correct? (choose closest number)

(a) 67.0 %
(b) 5.0 %
(c) 2.5 %
(d) 0.3 %

## *RA20: Adaptive Density Method*[8]

This article discusses a buffer sizing method as discussed in the summary article "*RA17: Sizing buffers* on page 169. More precisely, it describes the adaptive

[8]This section is also available as an article entitled "Sizing CC/BM buffers: The adaptive procedure with density" published online at PM Knowledge Center.

procedure with density method, in which the size of the buffers is based on the structure of the subnetwork that feeds buffer.

Figure 8.8 on page 167 displays a project network with eight activities. The numbers above each node are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6. More information on the determination of these buffers can be found in the article "*RA16: Buffering*" on page 166.

## The Adaptive Procedure with Density

The rationale behind buffer sizing methods based on network density is that subnetworks with an increasing number of precedence relations for a given number of activities have a high chance to cause delays. To that purpose, the so-called adaptive procedure with density (APD) has been proposed, which takes the density of the subnetworks that feed the buffer into account. Buffers are sized according to these subnetwork density values to assure that a higher number of precedence relations leads to a bigger buffer size to increase their power to absorb potential delays.

The buffer is sized as the standard deviation of the path leading to the buffer scaled by a factor that is calculated by taking the density of the subnetwork merging into the buffer into account, as follows:

Buffer size = K x $\sigma_{path}$

with

| | |
|---|---|
| K | The scaling factor based on the subnetwork density. |
| $\sigma_{path}$ | The standard deviation of the longest path feeding the buffer. |

buffer size
=
standard deviation of the path leading to the buffer scaled by a factor that is calculated by taking the resource density into account.

In the following sections, the calculations of the path standard deviations, the density of the subnetworks and the scale parameter K are illustrated on the example network of Fig. 8.8.

### Standard Deviation

The standard deviation of the longest path can be calculated in various ways. In this article, it is assumed that the standard deviations of the activities are equal to 50 % of the difference between the safe and aggressive durations (see Table 8.2). The standard deviation $\sigma_{path}$ of a path is then equal to the square root of the sum of the variances of the activities on that path, as follows:

- Critical chain 2-6-8: $\sqrt{1.5^2 + 1^2 + 0.5^2} = 1.87$
- Feeding chain 1-4: $\sqrt{0.5^2 + 0.5^2} = 0.71$
- Feeding chain 3-5-7: $\sqrt{1^2 + 0.5^2 + 0.5^2} = 1.22$

More detailed calculations are given in the article "*RA17: Sizing buffers*" on page 169.

### Network Density

The network density of a subnetwork merging into the buffer is a measure of how dense the network is in terms of the number of activities and precedence relations between these activities. The network density used in the adaptive procedure with density is equal to the number of precedence relations divided by the number of activities in the subnetwork, but obviously many alternatives could have been used. This density value is known as the coefficient of network complexity (CNC).

Figure 8.9 shows the three subnetworks for the three time buffers with their corresponding density values. Since the first node (S) and the last node (E) are

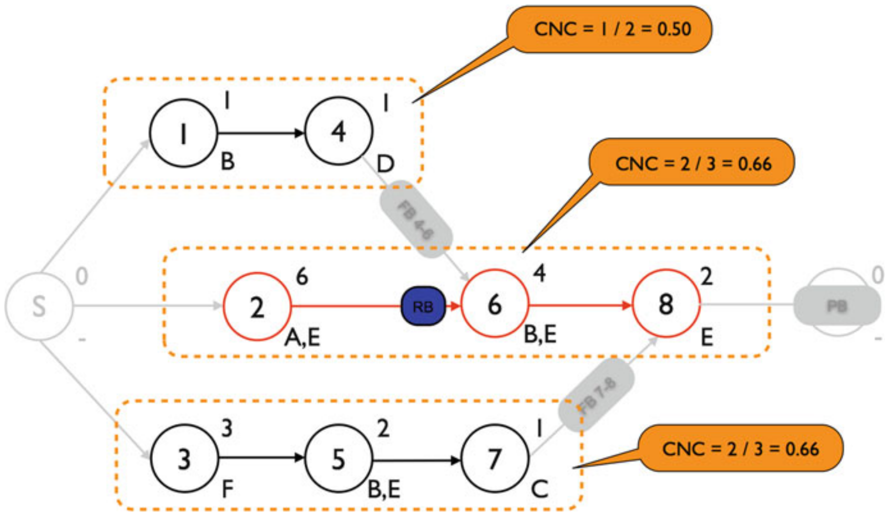



**Fig. 8.9** Three subnetworks merging into a buffer with their CNC value

dummy nodes with a zero duration and no resource demand, the precedence relations connected to these nodes are excluded from the calculations.

It should be noted that the three subnetworks in Fig. 8.9 are all single paths. In more complex project networks, these subnetworks can consist of networks with serial and parallel activities. In this case, the coefficient of network complexity used to measure the subnetwork density should take the complete network into account, and not only the longest path. The calculation of the path standard deviation, on the other hand, only takes the longest path of the subnetwork into account.

### Scale Parameter K

The network density is then used to calculate the scale factor K that is equal to:

K = 1 + network density

The buffer sizes based on the adaptive procedure with density are then equal to:

- Project buffer: 1.66 * 1.87 = 3.10
- $FB_{4\text{-}6}$: 1.50 * 0.71 = 1.07
- $FB_{7\text{-}8}$: 1.66 * 1.22 = 2.03

The rationale behind this buffer sizing method that a higher number of precedence relations for the same number of activities is possibly more sensitive to delays makes sense. When the coefficient of network complexity increases, it is indeed more likely that delays will occur more often and therefore larger buffers should be added that are able to absorb these delays. An overview of other buffer sizing methods is available in the article "*RA17: Sizing buffers*" on page 169.

### Questions

The questions below are based on the project network given in Fig. 8.10 on page 181. The numbers above the activities represent the activity durations, the letters below the activities represent the resource demand. The resource availability of all resources is equal to 1. The Critical Chain has already been marked in red or dark grey and consists of activities 1, 3 and 7.

1. On which arcs of the project network should the feeding buffers be placed using the Critical Chain/Buffer Management (CC/BM) methodology?

   (a) Both on arcs S-1 and 5-7
   (b) Both on arcs 1-3 and 3-7
   (c) Both on arcs 5-7 and 1-3
   (d) Only on arc 5-7
   (e) Only on arc 1-3
   (f) None of the above.

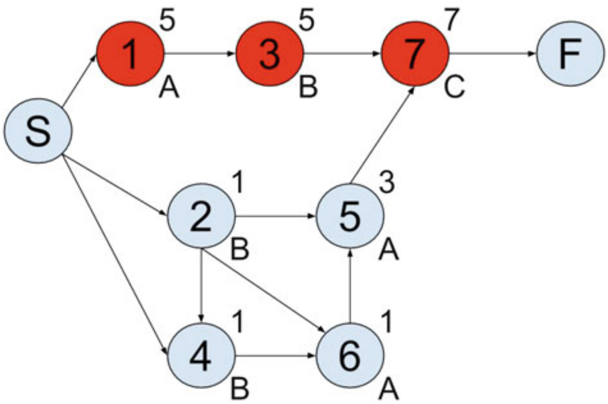



**Fig. 8.10** An example project network

2. How many resource buffers are needed?
   (a) 0
   (b) 1
   (c) 2
   (d) 3
3. What is the coefficient of network complexity (CNC) of the critical chain?
   (a) 1/3
   (b) 2/3
   (c) 3/2
   (d) 1
   (e) 3
4. What is the coefficient of network complexity (CNC) of the subnetwork containing the following activities: {2,4,5,6}?
   (a) 2/5
   (b) 2/3
   (c) 4/5
   (d) 5/4
   (e) 1
   (f) None of the above.
5. Assuming that the standard deviation of the activities is equal to 50 % of their duration, what is the size of the feeding buffer between activities 5 and 7 when

calculated using the adaptive procedure with density? (Assume that the scale parameter K = 1 + network density.)

(a) 1.73
(b) 2.16
(c) 3.90
(d) 4.16
(e) None of the above.

6. Assuming that the standard deviation of the activities is equal to 50 % of their duration, what is the size of the project buffer when calculated using the adaptive procedure with density? (Assume that the scale parameter K = 1 + network density.)

(a) 1.94
(b) 2.92
(c) 3.90
(d) 4.86
(e) 8.29

## *RA21: Adaptive Resource Tightness Method*[9]

This article discusses a buffer sizing method as discussed in the summary article "*RA17: Sizing buffers* on page 169. More precisely, it describes the adaptive procedure with resource tightness, in which the buffers are sized based on the scarceness of the resources used by the activities on the chain merging into the buffer.

Figure 8.8 on page 167 displays a project network with eight activities. The numbers above each node are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6. More information on the determination of these buffers can be found in the article "*RA16: Buffering*" on page 166.

[9]This section is also available as an article entitled "Sizing CC/BM buffers: The adaptive procedure with resource tightness" published online at PM Knowledge Center.

### The Adaptive Procedure with Resource Tightness

The rationale behind buffer sizing methods based on resource information is that resources that are scarce often have a huge impact on the duration of activities, and hence, on potential delays. To that purpose, the so-called adaptive procedure with resource tightness (APRT) has been proposed that takes the scarceness or tightness of resources into account. Buffers are sized according to these resource tightness values to assure that a higher degree of resource tightness leads to a bigger buffer size to increase their power to absorb potential delays.

The buffer is sized as the standard deviation of the path leading to the buffer scaled by a factor that is calculated by taking the resource tightness into account, as follows:

Buffer size = K x $\sigma_{path}$

with

| | |
|---|---|
| K | The scaling factor based on the resource tightness. |
| $\sigma_{path}$ | The standard deviation of the longest path feeding the buffer. |

> buffer size
> =
> standard deviation of the path leading to the buffer scaled by a factor that is calculated by taking the resource tightness into account.

In the following sections, the calculations of the path standard deviations, the resource tightness and the scale parameter K are illustrated on the example network of Fig. 8.8 on page 167.

### Standard Deviation

The standard deviation of the longest path can be calculated in various ways. In this article, it is assumed that the standard deviations of the activities are equal to 50 % of the difference between the safe and aggressive durations (see Table 8.2 on page 170). The standard deviation $\sigma_{path}$ of a path is then equal to the square root of the sum of the variances of the activities on that path, as shown in the following calculations:

- Critical chain 2-6-8: $\sqrt{1.5^2 + 1^2 + 0.5^2} = 1.87$
- Feeding chain 1-4: $\sqrt{0.5^2 + 0.5^2} = 0.71$
- Feeding chain 3-5-7: $\sqrt{1^2 + 0.5^2 + 0.5^2} = 1.22$

More detailed calculations are given in the article "*RA17: Sizing buffers*" on page 169.

### Resource Tightness

The resource tightness is a measure of the degree of resource use along the time horizon of all activities on the chain merging into the buffer. More precisely, it compares the total resource work content used by these activities with the total resource work content available during this time horizon for all resources.

- Total work content used by all activities on the chain: The work content is equal to the activity duration multiplied by the resource demand. As an example, the work content for activity 3 is equal to 3 * 1 = 3 for resource F.
- Total work content available along the length of the chain: The work content available is equal to the duration of the longest path feeding the buffer multiplied by the resource availability of the resource. For example, this is equal to 12 * 2 for resource E of the critical chain.

For more information on the difference and use of the resource demand, resource availability and work content, see e.g. the articles "*BS8: Linking resources*" on page 36 and article "*BS9: Activity costs*" on page 39.

The resource tightness is then calculated as the division of the two work contents, which leads to a value between 0 and 1, as follows:

- Resource tightness = total work content used/total work content available

Table 8.5 shows the resource tightness values for the different resources that are used for sizing the project buffer and the two feeding buffers.

These calculations are illustrated for resources A and E on the critical chain that has a length equal to 12 time units (e.g. days).

Resource A is only used by activity 2, and its total work content used is therefore equal to the duration of activity 2 multiplied by the resource demand, i.e. 6 * 1 = 6 (man days). The availability of resource A is equal to 1, and hence, its total work content available is equal to 12 * 1 = 12. Consequently, the periodic resource tightness for resource A along the critical chain is equal to 6/12 = 0.5.

The resource demand for resource E is equal to one for all activities (2, 6 and 8) on the critical chain and the total work content used is then equal to (6*1 + 4*1 + 2*1) = 12. Its availability is equal to 2, and hence, its total work content available is equal to 12 * 2 = 24. Consequently, the periodic resource tightness for resource E along the critical chain is equal to 12/24 = 0.5.

The resource tightness is then set as the maximum of all resource tightness values of all resources used by the activities in the chain (as displayed in the last row in Table 8.5).

### Scale Parameter K

The scaling factor K is equal to:

K = 1 + maximum resource tightness

**Table 8.5** Average resource use necessary to calculate the resource tightness

| Resource | PB | $FB_{4\text{-}6}$ | $FB_{7\text{-}8}$ |
|---|---|---|---|
| A | 6/12 | – | – |
| B | 4/12 | 1/2 | 2/6 |
| C | – | – | 1/6 |
| D | – | 1/2 | – |
| E | 12/24 | – | 2/12 |
| F | – | – | 3/6 |
| Maximum | 6/12 = 0.5 | 1/2 = 0.5 | 3/6 = 0.5 |

The buffer sizes based on the adaptive procedure with resource tightness are then equal to:

- Project buffer: 1.5 * 1.87 = 2.81
- $FB_{4\text{-}6}$: 1.5 * 0.71 = 1.07
- $FB_{7\text{-}8}$: 1.5 * 1.22 = 1.83

The rationale behind this buffer sizing method that a higher average resource use and consequently a higher resource tightness is more risky and could possibly be the cause of delays makes sense. When the total resource use is close to the total resource availability, it is indeed more likely that delays will occur more often and therefore larger buffers should be added that are able to absorb these delays. An overview of other buffer sizing methods is available in the article "*RA17: Sizing buffers*" on page 169.

## Questions

Consider the project data given in Table 8.6. The resources A, B and C are assumed to have an availability of 2, 1 and 2, respectively. In case multiple critical chains can be found, select the one with the shortest duration after inserting the buffers.

1. How many times should you calculate the resource tightness to determine the buffer sizes of all buffers?

   (a) 2
   (b) 3
   (c) 5
   (d) 6

2. What is the available work content for resource A on the critical chain?

   (a) 12
   (b) 19
   (c) 38
   (d) 76

**Table 8.6** Example project data

| Activity | Safe duration | Aggressive duration | Delta duration | Resource A | Resource B | Resource C | Standard deviation | Successors |
|---|---|---|---|---|---|---|---|---|
| 1 | 6 | 4 | 2 | 1 | 1 | 1 | 1 | 2,3 |
| 2 | 5 | 3 | 2 | 1 | 1 | 1 | 1 | 4 |
| 3 | 10 | 6 | 4 | 0 | 1 | 1 | 2 | 5 |
| 4 | 5 | 3 | 2 | 0 | 0 | 1 | 1 | 6 |
| 5 | 2 | 1 | 1 | 0 | 1 | 0 | 0.5 | 6 |
| 6 | 2 | 1 | 1 | 0 | 1 | 0 | 0.5 | 7 |
| 7 | 6 | 4 | 2 | 1 | 0 | 0 | 1 | – |

3. What is the work content used for resource C on the feeding chain?
   (a) 3
   (b) 6
   (c) 9
4. What is the size of the buffer for the critical chain when calculated using the resource tightness method?
   (a) 2.16
   (b) 3.53
   (c) 4.46
   (d) 4.90
   (e) 13.42
5. What is the size of the buffer for the feeding chain when calculated using the resource tightness method?
   (a) 0.50
   (b) 1.00
   (c) 1.50
   (d) 3.00

## RA22: Buffer Insertion[10]

The Critical Chain/Buffer Management (CC/BM) approach uses project and feeding buffers to add safety time to the project baseline schedule and to guarantee the timely completion of the project with a high probability. Inserting buffers into the project schedule creates a buffered project baseline schedule that can act as a tool to measure

[10]This section is also available as an article entitled "Critical Chain/Buffer Management: Inserting buffers in a resource constrained schedule" published online at PM Knowledge Center.

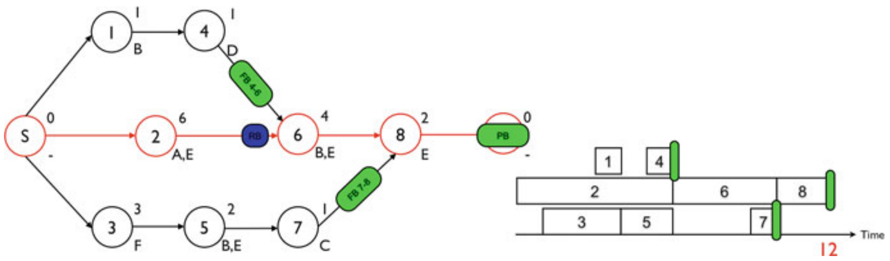



**Fig. 8.11** A buffered project network and a schedule with the buffer places indicated

performance and provides dashboards (i.e. the buffers) that need to be monitored to trigger corrective actions. In this article, the insertion of buffers is discussed and potential problems caused by activity shifts due to the time buffers are highlighted.

## *Buffer Insertion*

The project and the feeding buffers are treated as extra activities without predefined resource requirements and must be placed in the schedule to act as a protection for potential delays. The buffers are placed at the end of a chain for which the buffer is created, leading to an increase in the project duration compared to the unbuffered project schedule.

Figure 8.11 on page 187 shows the buffered project network (left) and a corresponding schedule (right) with the time buffer places indicated in the schedule based on the example discussed in the article "*RA16: Buffering*" on page 166. The numbers above each node in the project network are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6.

Each buffer needs to be sized according to a predefined sizing method (see article "*RA17: Sizing buffers*" on page 169) and will therefore consume time. In Fig. 8.12 on page 188, the buffered schedule is displayed using the cut and paste method (see article "*RA18: Cut and paste method*" on page 172) to determine the size of the buffers. Obviously, the buffered schedule has a longer duration (18 time units) than the unbuffered schedule displayed in Fig. 8.11 on page 187 (12 time units). Since the resource buffer is nothing more than a warning signal and does not consume time, it has not been displayed in the buffered schedule.

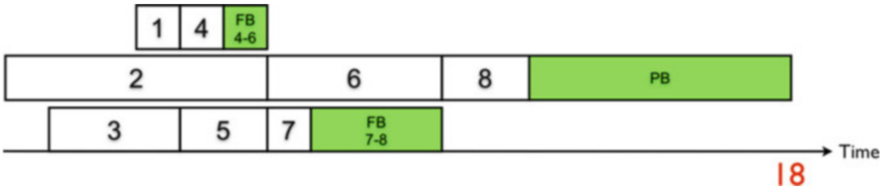



**Fig. 8.12** A resource feasible buffered project baseline schedule using the cut and paste method

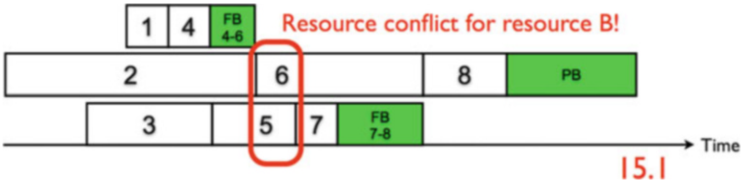



**Fig. 8.13** A buffered project baseline schedule using the adaptive procedure with density causing resource conflicts

Since activity durations are set to aggressive values (see article "*RA14: Aggressive estimates*" on page 158), these buffers act as a protection to changes in activity durations and should therefore be able to absorb potential delays.

### Resource Conflicts

Although the insertion of the buffers in the baseline schedule looks like a straightforward task, this insertion is often a cumbersome task, which leads to problems that harm the general philosophy of critical chain/buffer management. It is indeed not always easy to insert buffers in the schedule due to the shifts in activities. These activity shifts can lead to resource overallocations, known as resource conflicts (see article "*BS7: Critical path/chain*" on page 31) as illustrated in Fig. 8.13 on page 188. In this figure, buffers are sized according to the adaptive procedure with density (see article "*RA20: Adaptive density method*" on page 177) resulting in the following buffer sizes:

- Buffer size PB = 3.10
- Buffer size $FB_{4\text{-}6}$ = 1.07
- Buffer size $FB_{7\text{-}8}$ = 2.03

The project duration of the buffered schedule is equal to 15.1 time units (e.g. days) but shows a resource conflict for resource B. Since the availability of this resource is equal to 1 and the resource is used by both activities 5 and 6, there is a time period where resource B is used twice. This resource conflict caused by the shift of activity 5 leads to a buffered schedule that is no longer resource feasible.

Currently, there is no standardized approach available to cope with these resource conflicts generated by the insertion of the buffers. In the article "*RA23: Resource conflicts*" on page 189, various approaches to solve these resource conflicts are discussed.

**Questions**

1. Inserting buffers in a resource feasible schedule can result in a buffered schedule that is no longer resource feasible.
   (a) True
   (b) False
2. Resource buffers act as warning signals rather than as time buffers since they do not consume time.
   (a) True
   (b) False
3. A resource buffer is used to avoid resource conflicts for the activities on the critical chain.
   (a) True
   (b) False
4. The activities that constitute the critical chain before adding the buffers, will always lie on the longest chain after buffer insertion.
   (a) True
   (b) False

## RA23: Resource Conflicts[11]

Inserting buffers in a resource feasible project schedule is one of the last steps in the Critical Chain/Buffer Management (CC/BM) approach. Although the insertion of the buffers in the baseline schedule looks like a straightforward task, this insertion mechanism is often a cumbersome task, which leads to problems that harm the general philosophy of critical chain/buffer management. Inserting buffers often leads to resource conflicts in which some of the resources are overallocated (see article "*BS7: Critical path/chain*" on page 31). In this article, three simple

[11]This section is also available as an article entitled "Inserting buffers in a schedule: The problem with resource conflicts" published online at PM Knowledge Center.

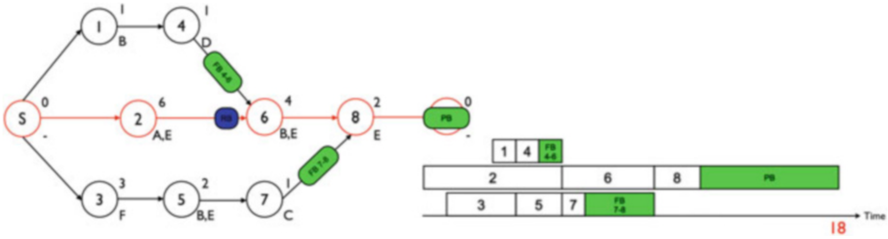


**Fig. 8.14** A buffered project network and schedule using the cut and paste method

techniques are described to cope with these resource conflicts in a CC/BM buffered schedule, namely:

- Allowing gaps in the chain
- Splitting buffers
- Resizing buffers

These techniques will be illustrated on an example project originally used in the article "*RA15: Latest start schedules*" on page 161. The left picture of Fig. 8.14 shows the buffered project network in which the numbers above each node are used to refer to the aggressive activity durations while the label below the node refers to a renewable resource that is required to perform the activity. The renewable resources A, B, C, D and F have an availability of one, while the renewable resource E availability is restricted to two units. The schedule contains three time buffers and one resource buffer. One project buffer is added to protect the critical chain S-2-6-8-E and two feeding buffers $FB_{4\text{-}6}$ and $FB_{7\text{-}8}$ are added to protect the feeding chains 1-4 and 3-5-7, respectively. The resource buffer RB is added to assure that resource B will be available on time to start with activity 6. The right part of Fig. 8.14 displays a corresponding buffered schedule using the cut and paste method. In this method, the size of the time buffers is equal to 50 % of the length of the chain as discussed in the article "*RA18: Cut and paste method*" on page 172.

In the remainder of this article, the size of the buffers will be changed to illustrate the presence of resource conflicts and to show some simple and straightforward ways to resolve these conflicts to make the buffered schedule resource feasible.

## *Allowing Gaps in the Chain*

The top schedule in Fig. 8.15 displays a buffered project baseline schedule using the adaptive procedure with density causing a resource conflict. In this procedure, buffers are sized according to the density of the subnetwork that feeds the chain.

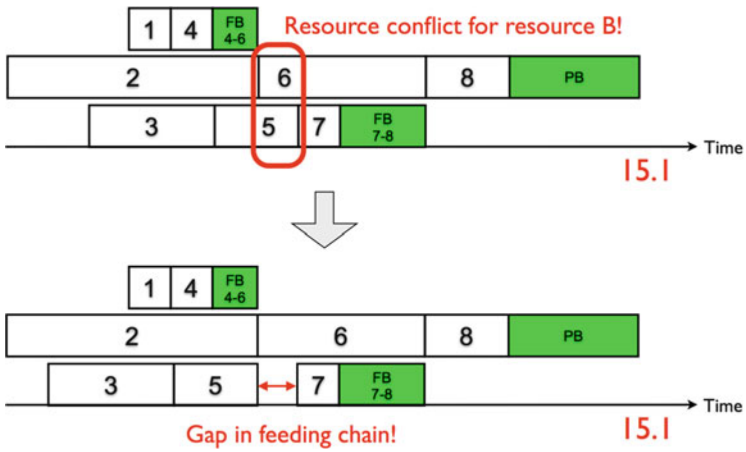



**Fig. 8.15** A buffered project baseline schedule with a feeding chain gap to resolve resource conflicts

This buffer sizing method is discussed in the article "*RA20: Adaptive density method*" on page 177 and leads to the following buffer sizes:

- Project buffer: 3.10
- $FB_{4\text{-}6}$: 1.07
- $FB_{7\text{-}8}$: 2.03

Thanks to a decrease in the size of the feeding buffer $FB_{7\text{-}8}$ and the project buffer compared to the buffers in Fig. 8.14, the total project duration has decreased to 15.1 time units (e.g. days). However, inserting these buffers in the project baseline schedule has led to a resource conflict for resource B used by activities 5 and 6 at the same time. A simple and pragmatic way to overcome this resource conflict is to allow a gap between activities 5 and 7 in the feeding chain, resulting in a resource feasible buffered project schedule, as displayed in the bottom picture of Fig. 8.15.

### Splitting Buffers

A second solution to cope with resource conflicts is illustrated in Fig. 8.16. In this figure, the sizes of the project buffer and feeding buffer $FB_{4\text{-}6}$ are equal to the sizes determined by the cut and paste method (6 time units and 1 time unit, respectively, as shown in Fig. 8.14). However, the size of buffer $FB_{7\text{-}8}$ has been increased. Since the size of the buffer should be a reflection of the uncertainty in the activity durations, it is assumed that the $FB_{7\text{-}8}$ size is increased from 3 to 3.5 time units in order to cope with a higher degree of uncertainty. Inserting this buffer in the project schedule

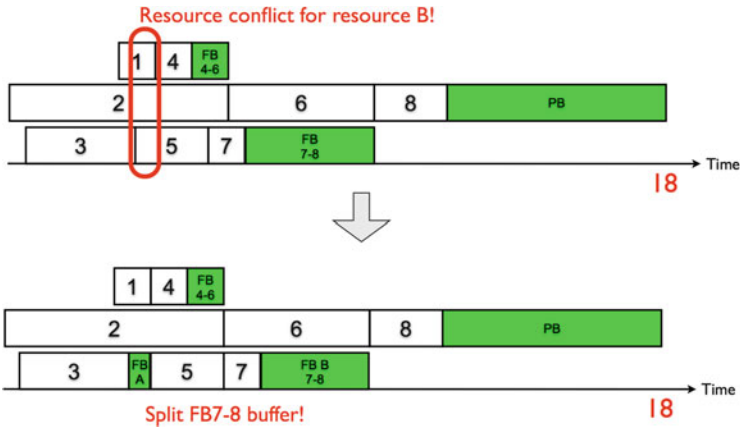



**Fig. 8.16** A buffered project baseline schedule with a split buffer to resolve resource conflicts

results in a new resource conflict, as displayed in the buffered schedule at the top of Fig. 8.16. Since activities 1 and 5 both rely on resource B that has an availability of 1 unit, this resource cannot be available for both activities at the same time. In order to avoid this resource conflict, the feeding buffer $FB_{7\text{-}8}$ has been split into parts A and B such that no resource conflicts occur in the buffered schedule, as shown in the bottom schedule of Fig. 8.16.

### Resizing Buffers

Resizing buffers to avoid resource conflicts can be considered as a last and probably the most simple and pragmatic technique to resolve resource conflicts. As an example, the buffer sizes obtained by the cut and paste method as displayed in Fig. 8.14 do not lead to resource conflicts.

It should be noted that all these methods to resolve resource conflicts might result in a buffered baseline schedule where resources are not overallocated, but the risk might be that the resulting buffering approach is no longer capable of absorbing potential delays in the chains feeding into these buffers. Therefore, these methods must be applied with utmost care. The sizes of the buffers should always be a reflection of the uncertainty of the activity duration in the chain leading to the buffer and the buffered schedule should therefore be constructed so that it is capable of coping with potential delays during the progress of the project. Simply resolving resource conflicts might put that objective into danger.

## Questions

1. Resolving a resource conflict by allowing gaps in the chain can result in a new resource conflict on the same or another resource.
   (a) True
   (b) False
2. The pragmatic ways of resolving resource conflicts in a buffered schedule, such as splitting buffers or allowing gaps in chains, might have the risk that the buffers loose their capability to protect against potential delays.
   (a) True
   (b) False
3. Allowing gaps in a feeding chain could result in a feeding path that is longer than the longest chain in the network, called the critical chain.
   (a) True
   (b) False
4. The size of the buffer should reflect the uncertainty of the activity durations in the chain leading to the buffer.
   (a) True
   (b) False
5. Splitting the feeding buffer is probably a good idea if activity durations near the end of the chain are more uncertain than those at the start of the chain.
   (a) True
   (b) False

# Part III
# Project Control

# Chapter 9
# Preface

## PC1: An Introduction to Project Control[1]

Project control (often referred to as project monitoring or project tracking) is the process performed to observe project execution in order to identify potential problems and/or opportunities in a timely manner such that corrective actions can be taken when necessary. The key benefit is that the current project status is observed on a regular basis, which enables the calculation of the project performance variance that is equal to the gap between actual performance and the baseline schedule. Since the current project performance is measured by variances from the project management plan, the baseline schedule plays a central and unambiguous role during the project tracking process.

The success of project monitoring and control relies on the quality of the baseline schedule data. Ideally, a project control approach should use all relevant information that project managers have obtained during the planning phase, and should rely on data from the baseline scheduling step (see article "*BS1: An introduction to baseline scheduling*" on page 9) as well as information from the schedule risk analysis phase (see article "*RA1: An introduction to risk analysis*" on page 109).

The project control topics of this book have been classified in the following categories:

- Earned Value Management
- Schedule control

For an overview of the three dynamic scheduling dimensions, see Fig. 1.1 on page 2 or the article "*Welcome to PM Knowledge Center*" on page 1. The different categories are briefly explained below.

[1]This section is also available as an article entitled "Dynamic scheduling: An introduction to project control" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_9

## *Earned Value Management*

Earned Value Management (EVM) is a methodology used to measure and communicate the real physical progress of a project and to integrate the three critical elements of project management (scope, time and cost management). It takes into account the work completed, the time taken and the costs incurred to complete the project and it helps to evaluate and control project risks by measuring project progress in monetary terms. The methodology has been used since the 1960s, when the USA department of defense proposed a standard method to measure a project's performance. The system relies on a set of often straightforward metrics to measure and evaluate the general health of a project. These metrics serve as early warning signals to timely detect project problems or to exploit project opportunities. The purpose of an EVM system is to provide answers to project managers on questions such as:

- What is the difference between budgeted and actual costs?
- What is the current project status? Ahead of schedule or schedule delay?
- Given the current project performance, what is the expected remaining time and cost of the project?

The research book "Measuring Time: Improving project performance using Earned Value Management" has been awarded the Research Collaboration Fund by the Belgian chapter of the Project Management Institute (PMI-Belgium) and the International Project Management Association (IPMA) (see Fig. 1.2 on page 4).

## *Schedule Control*

Controlling the performance of a project in progress is key to the success or failure of a project. Metrics provided by EVM systems measure the current progress and forecast the future expected project behavior such that the project manager is able to timely detect project problems and take corrective actions to bring the project back on track. In order to be able to take high quality corrective actions, the project manager should carefully control the project baseline schedule and set action thresholds that act as triggers to take these actions.

# Chapter 10
# Earned Value Management

## PC2: EVM Overview[1]

Controlling a project is key to the success or failure of the project. Measuring the project performance along the life of the project is a way to provide early warning signals that can be used as triggers for corrective actions in case the project is in danger. Earned Value Management (EVM) is a well-known technique to control the time and cost performance of a project. It is a methodology used since the 1960s, when the American Department of Defense proposed a standard method to measure a project's performance. The system relies on a set of often straightforward metrics to measure and evaluate the general health of a project. These metrics serve as early warning signals to timely detect problems or to exploit project opportunities. This article gives a brief overview of the main parameters and indicators used in an EVM approach, in order to give answers to the following questions, as summarized along the following lines:

- Key parameters: What is the difference between plan, cost and progress?
- Performance measures: What is the current project performance (time and cost)?
- Forecasting: What is the expected project duration and total cost (given current performance)?

Figure 10.1 shows the main components of an EVM analysis, divided in three different layers. For an overview of the EVM formulas, see article "*PC3: EVM formulary*" on page 203.

[1]This section is also available as an article entitled "Earned Value Management: An overview" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_10

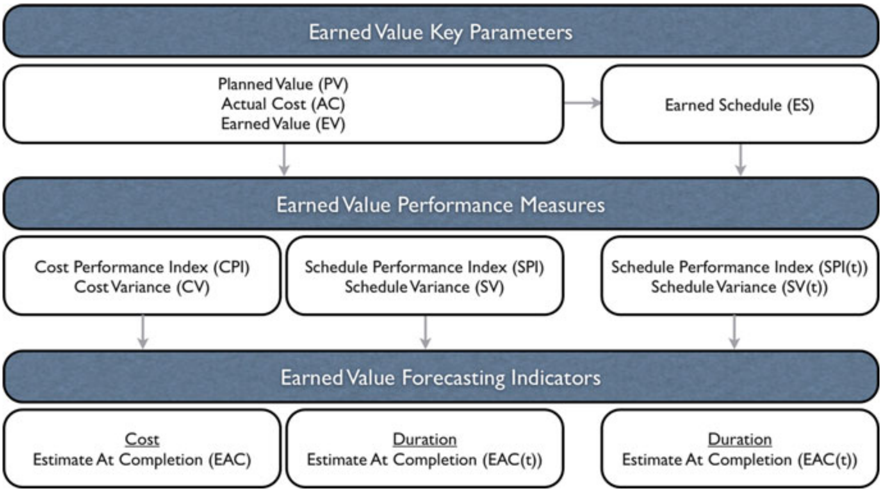



**Fig. 10.1** The three important components of Earned Value Management

## *Key Parameters*

The three key parameters of EVM are given along the following lines:

- Planned Value (PV): Time-phased budget baseline as an immediate result of the baseline schedule, often called the Budgeted Cost of Work Scheduled (BCWS).
- Actual Cost (AC): The cumulative actual cost spent at a given status date, often referred to as the Actual Cost of Work Performed (ACWP).
- Earned Value (EV): Represents the amount budgeted for performing the work that was accomplished by a given status date, often called the Budgeted Cost of Work Performed (BCWP) and equals the total activity (or project) budget at completion multiplied by the percentage activity (or project) completion (PC) at this particular point in time (= PC * BAC).

More information on the three key metrics of EVM can be found in the article "*PC5: Key metrics*" on page 208. Based on the Planned Value and Earned Value, a fourth key parameter can be automatically calculated as follows:

- Earned Schedule (ES): Translation of the EV of a given status date into time units by determining when this EV should have been earned in the baseline schedule.

The Earned Schedule metric measures your project progress in a time dimension and varies between 0 time units (at the start of the project) and the baseline Planned Duration (PD) at the end of the project. Hence, at the end of the project, EV = PV and ES = PD. More information can be found in the article "*PC6: Earned value and schedule*" on page 212.

## *Performance Measures*

Project performance, both in terms of time and costs, is determined by comparing the key parameters PV, AC, EV and ES, which results in the following performance measures.

- Time performance: The Schedule Performance Index (abbreviated as SPI or SPI(t) depending on whether EV or ES is used) is a measure to express the current time performance of the project, showing whether the project is ahead of schedule (>100 %), on time (=100 %) or late (<100 %).
- Cost performance: The Cost Performance Index (abbreviated as CPI) is a measure to express the current time performance of the project, showing whether the project cost is below budget (>100 %), on budget (=100 %) or above budget (<100 %).

Information on the calculation and interpretation of the performance measures is given in the article "*PC8: Project performance*" on page 217. A discussion on the two variants of the time performance measures is given in the article "*PC9: Time performance*" on page 220.

## *Forecasting Measures*

The project time and cost performance measures are assumed to be a representative indication for future project performance, and can therefore be used to forecast the final project duration and cost.

- Time forecasting: The Expected At Completion—Time (abbreviated as EAC(t)) is a forecast of the final project duration at the current status date, given the current project performance. Obviously, this forecast might differ from the baseline Planned Duration (PD).
- Cost forecasting: The Expected At Completion—Cost (abbreviated as EAC(€)) is a forecast of the total project cost at the current status date, given the current project performance. Obviously, this forecast might be different from the original budget or Budget At Completion (BAC).

Information on project forecasting can be found in the articles "*PC11: Forecasting time*" on page 228 and "*PC12: Forecasting cost*" on page 233.

## Questions

1. Earned Value Management (EVM) is applied prior to the project start and not during project execution.
   (a) True
   (b) False
2. The SPI expresses the current time performance of a project.
   (a) True
   (b) False
3. The planned value (PV) can also be called the budgeted cost of work performed (BCWP).
   (a) True
   (b) False
4. All EVM key parameters (PV, EV, AC and ES) are expressed in monetary units.
   (a) True
   (b) False
5. The percentage activity (or project) completion (PC) is needed to calculate the ...
   (a) EV
   (b) PV
   (c) AC
   (d) None of the above.
6. A project shows an SPI(t) of 85 % and a CPI of 105 %. This project is ...
   (a) under budget and behind schedule.
   (b) over budget and ahead of schedule.
   (c) under budget and ahead of schedule.
   (d) unknown, since no relevant information is available to come to a conclusion.
7. At the end of a project, it is always true that ...
   (a) EV = AC
   (b) ES = PD
   (c) EAC(t) = PD
   (d) None of the above.

## PC3: EVM Formulary[2]

Controlling a project is key to the success or failure of the project. Measuring the project performance along the life of the project is a way to provide early warning signals that can be used as triggers for corrective action in case the project is in danger. Earned Value Management (EVM) is a well-known technique to control the time and cost performance of a project. It is a methodology used since the 1960s, when the American Department of Defense proposed a standard method to measure a project's performance. The system relies on a set of often straightforward metrics to measure and evaluate the general health of a project. These metrics serve as early warning signals to timely detect project problems or to exploit project opportunities. This article gives an overview of the terminology used in EVM and in the different articles published in this book. In the list below, the main components of an EVM analysis are divided into three different layers. For an overview of the general EVM approach and references to further articles, see article "*PC2: EVM Overview*" on page 199.

- Earned Value Management: Key Metrics
  - S-curve: This graph displays the Planned Value (PV), the Actual Cost (AC) and Earned Value (EV) along the life of the project (Article: "*PC5: Key metrics*" on page 208).
  - PV curve: This graph displays the Planned Value (PV) as shown in the S-curve. Since the Planned Value curve is available at the construction of the baseline schedule (before the EVM tracking), this graph is accessible separately from the S-curve (Article: "*PC4: Planned value*" on page 205).
  - Earned Schedule (ES): This graph displays the Earned Schedule (ES) calculated from the Earned Value and Planned Value graph along the life of the project (Article: "*PC6: Earned value and schedule*" on page 212).
- Earned Value Management: Project Performance
  - EVM Performance Dashboard: This graph displays both the time and cost performance and divides the project performance into four regions showing time and cost performance.
  - Cost Variance (CV) : This graph displays the Cost Variance (CV = EV − AC) along the life of the project (Article: "*PC8: Project performance*" on page 217).
  - Cost Performance (CPI): This graph displays the Cost Performance Index (CPI = EV/AC) along the life of the project (Article: "*PC8: Project performance*" on page 217).

[2]This section is also available as an article entitled "Earned Value Management: The EVM formulary" published online at PM Knowledge Center.

- Schedule Variance (SV and SV(t)): This graph displays the Schedule Variance (SV or SV(t)) along the life of the project. Formulas used: $SV = EV - PV$ and $SV(t) = ES - AT$ (Article: "*PC9: Time performance*" on page 220).
- Schedule Performance (SPI and SPI(t)): This graph displays the Schedule Performance Index (SPI or SPI(t)) along the life of the project. Formulas used: $SPI = EV/PV$ and $SPI(t) = ES/AT$ (Article: "*PC9: Time performance*" on page 220).
- Schedule Adherence (p-factor): This graph displays how good the project progress follows the baseline schedule philosophy. This is known as schedule adherence and measured by the p-factor (100 % = perfect adherence). (Article: "*PC14: Schedule adherence*" on page 241).

- Earned Value Management: Project Forecasting
  - Cost Estimate At Completion (EAC(€)): This graph displays the estimated final cost at project completion (EAC(€)) predicted along the life of the project. Eight forecasting versions are used, in line with research from literature (Article: "*PC12: Forecasting cost*" on page 233).
  - Time Estimate At Completion (EAC(t)): This graph displays the estimated final duration at project completion (EAC(t)) predicted along the life of the project. Three methods are used (PVM, EDM and ESM), each using three variants (Article: "*PC11: Forecasting time*" on page 228).
  - MAPE: This graph displays the Mean Absolute Percentage Error as a measure of the forecast accuracy of time or cost predictions (Article: "*PC13: Forecast accuracy*" on page 236).
  - MPE: This graph displays the Mean Percentage Error as a measure of the forecast accuracy of time or cost predictions (Article: "*PC13: Forecast accuracy*" on page 236).

## Questions

1. The PV, AC and EV along the life of a project can all be represented by an S-curve.
   (a) True
   (b) False
2. The PV curve is already available before the project has actually started.
   (a) True
   (b) False
3. The schedule variance SV is expressed in monetary units.
   (a) True
   (b) False

4. The schedule adherence of a project can be expressed by the MAPE.
   (a) True
   (b) False
5. Which one of the following formulas is correct?
   (a) CPI = AC/EV
   (b) SV = EV − PV
   (c) SPI(t) = ES/PV
   (d) None of the above.
6. Which one of the following formulas is not correct?
   (a) CV = EV − AC
   (b) SV(t) = ES − AT
   (c) SPI = EV/PV
   (d) SPI(t) = ES/PV

## EVM Key Metrics

### *PC4: Planned Value*[3]

Controlling a project is key to the success or failure of the project. Earned Value Management (EVM) is a well-known technique to control the time and cost performance of a project and to predict the final project duration and cost. It is an easy tool to generate early warning signals to timely detect problems or to exploit project opportunities. An overview of the EVM metrics is given in the article "*PC2: EVM Overview*" on page 199 and the formulas are summarized in the article "*PC3: EVM formulary*" on page 203.

In this article, it will be shown that the project baseline schedule plays a crucial role in EVM, and that the planned value curve is a direct translation of this baseline schedule in monetary terms. Project progress, and the measured project performance, is based on the current planned value, actual cost and earned value.

Figure 10.2 displays a project network with eight activities and finish-start precedence relations between them. Each number above the node denotes the estimated duration of the activity while the number below the node is used to refer to the estimated activity cost. It is assumed, without loss of generality, that no renewable resource restrictions are imposed on the project.

[3]This section is also available as an article entitled "Earned Value Management: The project baseline schedule's planned value" published online at PM Knowledge Center.

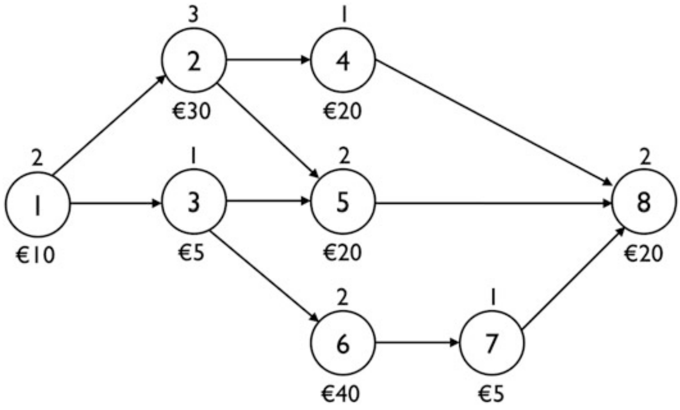



**Fig. 10.2** A fictitious example project network

## Project Baseline Schedule: The Planned Value

The Planned Value (PV) is the time-phased budget baseline as an immediate translation of the schedule constructed from the project network (without or with resources, as discussed in the article "*BS7: Critical path/chain*" on page 31). It is a cumulative increase in the total budgeted activity cost given the start and finish times stipulated in the baseline schedule. The planned value is often called budgeted cost of work scheduled (BCWS).

Figure 10.3 shows an earliest start schedule (see article "*BS10: Activity slack*" on page 43) of the example project of Fig. 10.2. The critical path is equal to the activity sequence 1-2-5-8 with a total duration of 9 weeks.

The assignment of the total budgeted cost of each activity depends on the EVM measurement method that will be employed for the measurement of project performance. Indeed, costs can be assigned to activities in various ways, e.g. all costs can be incurred upon the start of the activity, or only after a certain percentage is finished, or as a stepwise function (e.g. 20 % upon the start, 50 % after a first meeting with the client and a final 30 % upon the activity finish). In the Gantt chart of Fig. 10.3, it is assumed that the cost increase is linear over the activity duration, which is displayed by the cumulative cost values for each period (week) for each activity.

Figure 10.4 shows the total cumulative increase of all planned values given by the activities of the baseline schedule, resulting in a total budgeted cost of € 150. Using the EVM terminology (see article "*PC3: EVM formulary*" on page 203), the baseline schedule displays the following characteristics:

- Planned Duration (PD) = 9 weeks
- Budget at Completion (BAC) = € 150.00

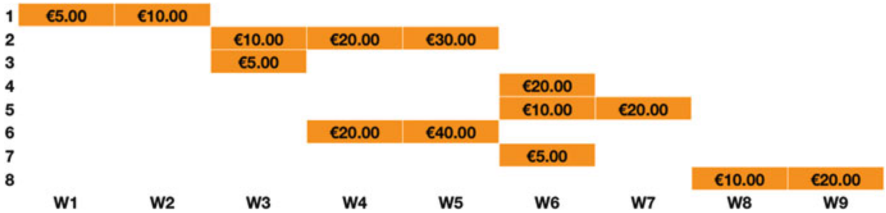



**Fig. 10.3** Earliest start schedule of the project of Fig. 10.2 with cumulative cost accrues

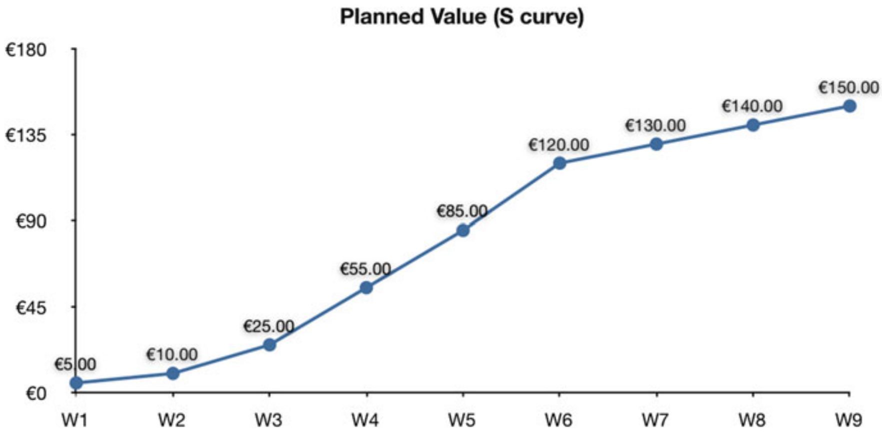



**Fig. 10.4** Planned value curve (S-curve)

## Questions

1. Constructing a reliable baseline schedule before project execution is important for applying EVM because the established schedule will be the basis for the planned value curve.

   (a) True
   (b) False

2. The critical path of a project network is always unique, that is, for every project network there is exactly one path that is defined as critical.

   (a) True
   (b) False

3. The planned duration (PD) of a project is calculated by summing the durations of all activities in the project network.

   (a) True
   (b) False

4. The budget at completion (BAC) of a project is calculated by summing the costs of all activities in the project network.

   (a) True
   (b) False

5. Consider the project network in Fig. 10.2 on page 206. If the duration of activity 5 changed from 2 to 1 time unit, all other things remaining equal, the following would happen:

   (a) The path 1-2-5-8 would remain the only critical path in the project network, the planned duration decreases by 1 time unit, the budget at completion remains unchanged.
   (b) All paths in the network are now critical, the planned duration decreases by 1 time unit, the budget at completion decreases by € 10.
   (c) All paths in the network are now critical, the planned duration as well as the budget at completion remain unchanged.
   (d) None of the above.

6. Consider the project network Fig. 10.2 on page 206. If the cost associated to activity 5 changed from € 20 to € 40, all other things remaining equal, the following would happen:

   (a) The path 1-2-5-8 is no longer critical.
   (b) The planned duration would increase by 2 time units.
   (c) The budget at completion would increase by € 20.
   (d) None of the above.

## *PC5: Key Metrics*[4]

This article builds further on the Earned Value Management (EVM) concepts summarized in the articles "*PC2: EVM Overview*" on page 199 and "*PC3: EVM formulary*" on page 203. It provides an overview of the way the earned value (EV) and actual costs (AC) are calculated and compared with the planned value (PV) in order to draw conclusions on the performance of a project in progress. All calculations are based on an example earliest start schedule Gantt chart of a fictitious project shown in Fig. 10.3.

[4]This section is also available as an article entitled "Earned Value Management: The three key metrics" published online at PM Knowledge Center.

### Baseline Schedule

The Planned Value (PV) is a time-phased budget baseline as an immediate result of the baseline schedule, translated in monetary terms. Table 10.1 displays the cumulative planned value of a project with a planned duration PD = 9 weeks and a budget at completion BAC = € 150. The numbers are calculated in the article "*PC4: Planned value*" on page 205 and are displayed in Table 10.1.

### Project Progress

Figure 10.5 displays the real life progress of the project up to week 7, with activity delays in activities 1 and 2 and a faster execution for activity 6. The periodic cumulative actual cost is given in each bar of these activities and might differ from the planned costs as displayed in Fig. 10.3. Activities 4, 5 and 8 are ready to be started and are therefore displayed with their baseline duration. During the project progress, the actual cost and the percentage completed is measured in order to obtain periodic data as given in Table 10.1.

Based on the progress up to today, the two remaining key parameters can be calculated. The Actual Cost (AC) is equal to the cumulative actual cost spent at a given status date (i.e. week 7) as the sum of the actual spendings of all activities. The Earned Value (EV) represents the amount budgeted for performing the work that was accomplished by a given status date (week 7). It is measured as the percentage completed, which is often a subjective estimate made by the project manager, multiplied by the baseline schedule cost for each activity. Example calculations for weeks 6 and 7 are given along the following lines.

**Table 10.1** Key metrics

| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 |
|---|---|---|---|---|---|---|---|---|---|
| PV | 5 | 10 | 25 | 55 | 85 | 120 | 130 | 140 | 150 |
| AC | 10 | 20 | 30 | 75 | 120 | 155 | 170 | | |
| EV | 3.33 | 6.67 | 10 | 22.5 | 70 | 82.5 | 90 | | |


**Fig. 10.5** Real life execution (project progress with cumulative actual cost values)

Example: week 6

- EV: 100 % * € 10 + 75 % * € 30 + 100 % * € 5 + 100 % * € 40 + 100 % * € 5 = € 82.5
- AC: € 30 + € 45 + € 30 + € 30 + € 20 = € 155

Example: week 7

- EV: 100 % * € 10 + 100 % * € 30 + 100 % * € 5 + 100 % * € 40 + 100 % * € 5 = € 90
- AC: € 30 + € 60 + € 30 + € 30 + € 20 = € 170

Figure 10.6 shows the S-curves for the three key parameters over time, up to week 7. Only the planned value is known from the project start until the end, as a result of the baseline schedule construction.

The values for the three key parameters at week 7 can be interpreted as follows:

- The PV curve measures how much value should have been earned according to the baseline schedule at W7, and is equal to € 130.00.
- The EV curve measures how much value has been earned at the current time W7 given the work that has been done up to now, which is equal to € 90.00.
- The AC curve measures the actual cost incurred up to the current time W7 given the work that has been done and equals € 170.00.

Based on this information, the metrics can be used to measure the current and past performance of the project in progress. Since the baseline schedule (PV) stipulates that an additional € 40.00 should have been earned in week 7 compared to what was actually earned (EV), the project is clearly late. Since the value of the work done up to now (AC) exceeds the value that should have been earned with the current work done as stipulated in the baseline schedule (EV), the project is clearly over budget. These time and cost deviations (underruns and overruns) can be expressed

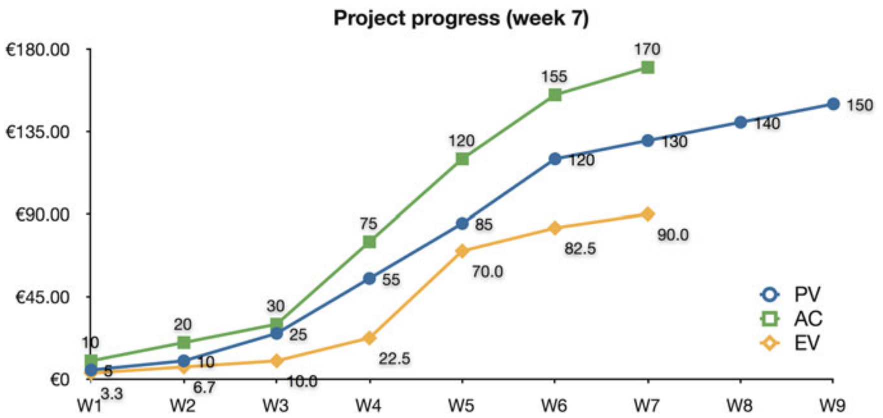



**Fig. 10.6** The PV, EV and AC S-curves

by performance indicators, as discussed in the article "*PC8: Project performance*" on page 217 and can be used to forecast the final project duration and total cost as explained in the articles "*PC11: Forecasting time*" on page 228 and "*PC12: Forecasting cost*" on page 233.

## Questions

1. If the actual cost AC is higher than the planned value PV at a certain evaluation moment, the earned value EV at that time instance is always lower than the planned value PV and vice versa.
   (a) True
   (b) False
2. The planned value PV can be established for each evaluation point before project initiation, while the actual cost and earned value are only calculated when the project progresses.
   (a) True
   (b) False
3. The calculation of the earned value EV is often based on subjective judgement since it can be difficult to estimate the exact portion of the work that has been completed.
   (a) True
   (b) False
4. The earned value EV at time instance $t$ is calculated by multiplying ...
   (a) the planned percentage complete and the planned cost.
   (b) the planned percentage complete and the actual cost.
   (c) the actual percentage complete and the planned cost.
   (d) the actual percentage complete and the actual cost.
5. Consider the graphs in Fig. 10.6 on page 210. Because the actual cost curve and the earned value curve are situated above and below the planned value curve respectively, the following situation might occur if no actions are taken to bring the project back on schedule.
   (a) The project is likely to finish late with cost overrun.
   (b) The project is likely to finish early with cost overrun.
   (c) The project is likely to finish late with cost underrun.
   (d) The project is likely to finish early with cost underrun.
6. Consider the Gantt chart in Fig. 10.5 on page 209 that indicates the project progress as indicated by the blue or dark grey part at week 7. Assume that the remaining work will be done as planned (see Fig. 10.3 on page 207). Which of the following statements is true?

(a) The project will finish at time instant 9 with a cost of € 230.
(b) The project will finish at time instant 9 with a cost of € 250.
(c) The project will finish at time instant 11 with a cost of € 230.
(d) The project will finish at time instant 11 with a cost of € 250.

## *PC6: Earned Value and Schedule*[5]

In the article "*PC5: Key metrics*" on page 208, it has been discussed that EVM relies on three key input metrics, known as the planned value (PV), the actual cost (AC) and the earned value (EV). In this article, a fourth key metric is discussed, which is based on the periodic values of the PV and EV. Figure 10.6 shows the S-curves for the three key parameters over time, up to week 7. Only the planned value is known from the project start until the end, as a result of the baseline schedule construction. The actual cost and the earned value are based on weekly progress measurements during project execution.

The fourth key metric, known as Earned Schedule (ES), is a simple translation of the EV of a given status date into time units by determining when this EV should have been earned in the baseline schedule. A formal definition of the ES metric can be given as follows:

Find $t$ such that $\mathrm{EV} \geq \mathrm{PV}_t$ and $\mathrm{EV} < \mathrm{PV}_{t+1}$

$$\mathrm{ES} = t + \frac{\mathrm{EV} - \mathrm{PV}_t}{\mathrm{PV}_{t+1} - \mathrm{PV}_t}$$

With

| | |
|---|---|
| $\mathrm{PV}_t$ | The planned value at time $t$ |
| EV | The earned value at the current week |

Example week 7:

EV at week 7 = € 90
$t$ = week 5 since $85 \leq 90$ and $120 > 90$
ES = 5 + (90 − 85)/(120 − 85) = 5.14

This value is illustrated graphically in Fig. 10.7.

While EVM measures schedule performance not in units of time, but rather in costs, the Earned Schedule metric, instead, measures your project progress in a time dimension and varies between 0 time units (at the start of the project) and the baseline Planned Duration (PD) at the end of the project. Hence, at the end of the project, EV = PV and ES = PD. In the article "*PC9: Time performance*" on page 220, it will be shown that the ES is also used to measure the time performance

[5]This section is also available as an article entitled "Measuring Time: Earned value or earned schedule?" published online at PM Knowledge Center.

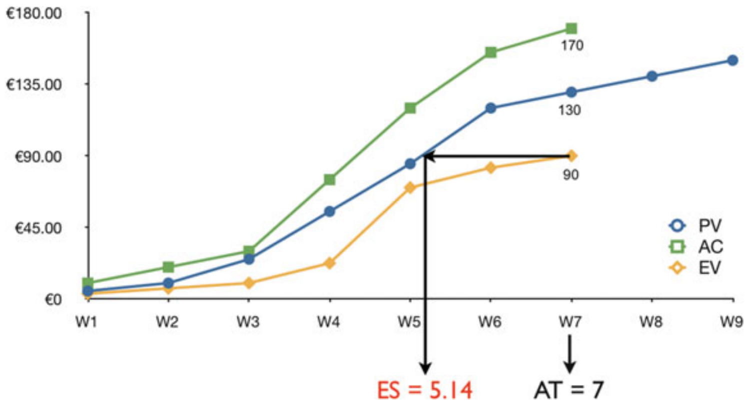



**Fig. 10.7** The Earned Schedule (ES) at week 7

of a project using an alternative schedule performance index, which is more reliable than the traditional SPI = EV/PV.

## Questions

1. In order to calculate the earned schedule ES, no information on the actual costs is needed.
   (a) True
   (b) False
2. If the earned value curve is situated strictly below the planned value curve, the earned schedule is always smaller than actual time and vice versa.
   (a) True
   (b) False
3. The earned schedule ES can be read from the horizontal time-axis and is thus expressed in time units.
   (a) True
   (b) False
4. At the end of the project, it is possible that the earned schedule ES is smaller than the planned duration PD, which indicates that the project is finished ahead of schedule.
   (a) True
   (b) False

5. Linear interpolation is often necessary to find the value for the earned schedule.
   (a) True
   (b) False
6. If, at a particular time instance, the earned value is equal to the earned value of the previous time instance, the values for the earned schedule of both time instances will always be equal.
   (a) True
   (b) False

## EVM Performance Measurement

### *PC7: Performance Scenarios*[6]

The time and cost performance of a project in progress can be determined by comparing the three key parameters of an earned value management system, known as planned value (PV), actual cost (AC) and earned value (EV). These three metrics can be used to calculate two time variance measures SV and SV(t) and one cost variance CV, as follows:

Schedule Variance SV = EV − PV

$<0$: project delay
$=0$: project on schedule
$>0$: project ahead of schedule

Schedule Variance with earned schedule SV(t) = ES − AT

$<0$: project delay
$=0$: project on schedule
$>0$: project ahead of schedule

Cost Variance CV = EV − AC

$<0$: budget overrun
$=0$: project on budget
$>0$: budget underrun

with AT the current or actual time.

For more information on the key metrics, the reader is referred to the article "*PC5: Key metrics*" on page 208. Information on the SV and CV metrics can be found in the article "*PC8: Project performance*" on page 217 while the SV(t) metric is explained in the article "*PC9: Time performance*" on page 220. An overview of

[6]This section is also available as an article entitled "Earned Value Management: An overview of project performance scenarios" published online at PM Knowledge Center.

these metrics in an earned value management system is given in the article "*PC2: EVM Overview*" on page 199.

Given the two dimensions (time and cost) and their possible variances (zero, positive or negative), nine different project performance situations might occur.

- Scenario 1: late project, over budget
- Scenario 2: late project, on budget
- Scenario 3: late project, under budget
- Scenario 4: on time, over budget
- Scenario 5: on time, on budget
- Scenario 6: on time, under budget
- Scenario 7: early project, over budget
- Scenario 8: early project, on budget
- Scenario 9: early project, under budget

When the scenarios "on time" and/or "on budget" are excluded, the four remaining scenarios 1, 3, 7 and 9 can be graphically displayed using example project data with a planned duration of 9 weeks and the current actual time AT is equal to week 6.

Figure 10.8 displays the three EVM key parameters and the SV and CV metrics under the four remaining time/cost scenarios. Depending on the SV values, the project is early or late. The CV values determine whether the project is over or under budget. Figure 10.9 displays the three EVM key parameters and the SV(t) metric for an early and a late project scenario.

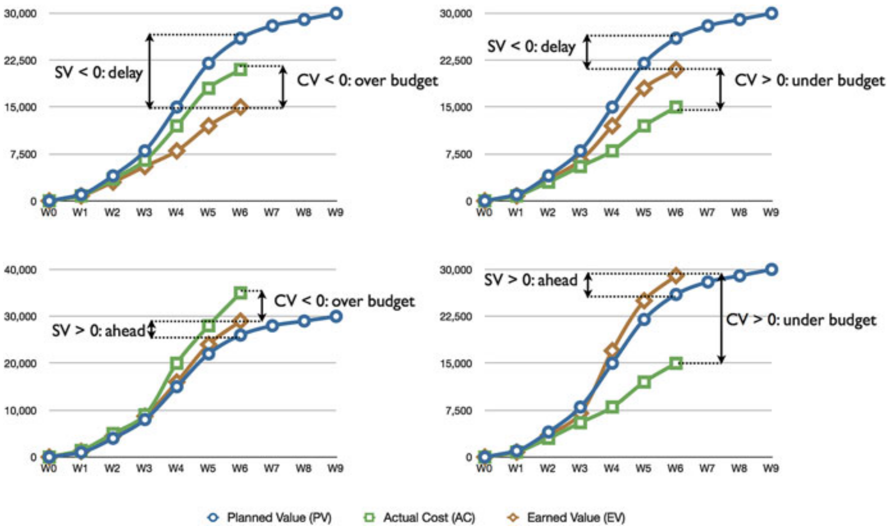



**Fig. 10.8** The EVM key metrics for early and late projects with cost under- and overruns

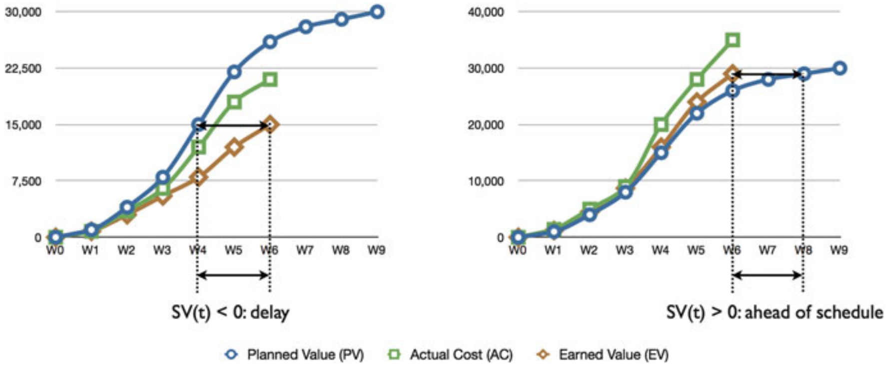



**Fig. 10.9** The PV and EV metrics for a late (*left*) and early (*right*) project

## Questions

1. The difference between the two time variance measures SV and SV(t) is that SV is expressed in monetary units, while SV(t) is expressed in time units.
   (a) True
   (b) False
2. Whether a project is likely to finish ahead of schedule or late can be seen by comparing the PV and the EV at that time instance, while cost overruns/underruns are indicated by the difference in AC and EV.
   (a) True
   (b) False
3. The schedule variance based on earned schedule (SV(t)) is more reliable than the one based on earned value (SV).
   (a) True
   (b) False
4. The schedule variance SV(t) is always measured on the vertical cost axis.
   (a) True
   (b) False
5. A project is likely to finish ahead of plan with a budget overrun if at time instant *t* the following holds true (measured on the vertical cost axis):
   (a) PV > EV > AC
   (b) EV > PV > AC
   (c) AC > PV > EV
   (d) AC > EV > PV

6. A project is likely to finish late with a budget overrun at time instant $t$. Which of the following inequalities is incorrect (values are measured on the vertical cost axis)?
   (a) PV > AC > EV
   (b) AC > PV > EV
   (c) AC > EV > PV

## *PC8: Project Performance*[7]

This article builds further on the Earned Value Management (EVM) concepts summarized in the articles "*PC2: EVM Overview*" on page 199 and "*PC3: EVM formulary*" on page 203. It measures the time and cost performance of a project in progress, based on periodic information of an example project given in Fig. 10.6. This figure displays the planned value line as well as the earned value and actual cost lines up to week 7. The project has a planned duration PD = 9 weeks and a budget at completion BAC = € 150.

### Project Performance

Project performance, both in terms of time and costs, is determined by comparing the key parameters PV, AC, EV and ES, resulting in a schedule (time) and cost variance, as follows:

- **Schedule Variance (SV)**: Shows the variance in time expressed in monetary terms EV − PV (see Fig. 10.10) and can be interpreted as follows:

  > 0: project ahead of schedule
  = 0: project on time
  < 0: project delay

- **Cost Variance (CV)**: Shows the variance in cost expressed in monetary terms as EV − AC and is displayed graphically in Fig. 10.10. The CV can be interpreted as follows:

  > 0: project under budget
  = 0: project on budget
  < 0: project over budget

[7] This section is also available as an article entitled "Earned Value Management: Measuring a project's performance" published online at PM Knowledge Center.

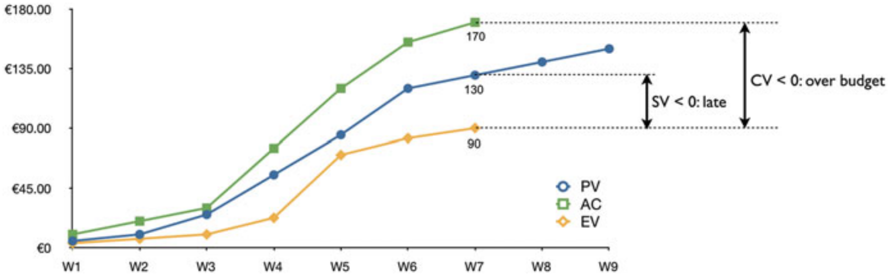



**Fig. 10.10** Real life execution (project progress at week 7)

**Table 10.2** The three EVM key metrics and the four performance measures

| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 |
|---|---|---|---|---|---|---|---|---|---|
| PV | 5 | 10 | 25 | 55 | 85 | 120 | 130 | 140 | 150 |
| AC | 10 | 20 | 30 | 75 | 120 | 155 | 170 | | |
| EV | 3.33 | 6.67 | 10 | 22.5 | 70 | 82.5 | 90 | | |
| SV | −1.67 | −3.33 | −15.00 | −32.50 | −15.00 | −37.50 | −40.00 | | |
| SPI | 0.67 | 0.67 | 0.40 | 0.41 | 0.82 | 0.69 | 0.69 | | |
| CV | −6.67 | −13.33 | −20.00 | −52.50 | −50.00 | −72.50 | −80.00 | | |
| CPI | 0.33 | 0.33 | 0.33 | 0.30 | 0.58 | 0.53 | 0.53 | | |

Both variances are expressed in monetary units. While this is obvious for a cost variance, the variance of time should be better expressed in a time dimension rather than in a monetary unit. To that purpose, the two variances can be translated to two well-known unitless performance indices, as follows:

- **Schedule Performance Index (SPI)**: Shows the performance of time (EV/PV) in a unitless dimension:

  $>$ 100 %: project ahead of schedule
  $=$ 100 %: project on time
  $<$ 100 %: project delay

- **Cost Performance Index (CPI)**: Shows the performance of cost (EV/AC) in a unitless dimension:

  $>$ 100 %: project under budget
  $=$ 100 %: project on budget
  $<$ 100 %: project over budget

Table 10.2 shows the three key metrics and the four performance measures for all periods until week 7. The three key metrics have been discussed in the article "*PC5: Key metrics*" on page 208. Figure 10.10 shows the S-curves of the three key metrics and the schedule and cost variances in a graphical way.

It should be noted that the schedule variance SV and schedule performance index SPI have been criticized since they are unreliable measures for the time performance of a project. An alternative variance and performance index has been proposed instead, as discussed in the article "*PC6: Earned value and schedule*" on page 212. An overview of different time and cost performance scenarios during a project's progress is given in the article "*PC7: Performance scenarios*" on page 214.

## Questions

1. If the schedule variance (SV) is lower than zero, the project is ...
   (a) behind schedule.
   (b) on time.
   (c) ahead of schedule.
2. If the cost performance index (CPI) is higher than 100 %, the project is ...
   (a) under budget.
   (b) on budget.
   (c) over budget.
3. The cost variance (CV) is expressed in monetary units, whereas the schedule variance (SV) is expressed in a time dimension.
   (a) True
   (b) False
4. Consider Table 10.2 on page 218 and assume that the AC and EV in period 8 are now known and equal to € 200 and € 120, respectively. What can you conclude about the project's schedule performance with respect to the previous period?
   (a) It improved.
   (b) It remained the same.
   (c) It deteriorated.
   (d) None of the above.
5. Consider Table 10.2 on page 218 and assume that the AC and EV in period 8 are now known and equal to € 200 and € 120, respectively. What can you conclude about the project's cost performance with respect to the previous period?
   (a) It improved.
   (b) It remained the same.
   (c) It deteriorated.
   (d) None of the above.
6. Consider Fig. 10.6 on page 210. The cost variance (CV) for period 5 is ...
   (a) € −50.
   (b) € −35.
   (c) € 35.
   (d) € 50.

7. Consider Fig. 10.6 on page 210. Which statement is true?
   (a) The project's schedule performance has been lower than the cost performance throughout the entire project.
   (b) The project's cost performance has been lower than the schedule performance throughout the entire project.
   (c) None of the above.

## *PC9: Time Performance*[8]

In the article "*PC5: Key metrics*" on page 208, it has been discussed that EVM relies on three key input metrics, known as the planned value (PV), the actual cost (AC) and the earned value (EV). In the article "*PC6: Earned value and schedule*" on page 212, it has been shown that a fourth method, known as Earned Schedule (ES), works better for measuring the time progress of a project.

In this article, the schedule performance index SPI is criticized for its unreliable behavior towards the end of a project and an alternative schedule performance index SPI(t) is presented. Example calculations will be used to illustrate this issue, based on the baseline schedule and fictitious project progress of a project that ends 2 weeks later than planned, as shown in Fig. 10.11. The baseline schedule displays the cumulative planned values for each activity and the real-life execution shows the cumulative actual costs for each activity.

### Project Performance

A project's time performance is measured by the schedule variance SV or the schedule performance index SPI, as discussed in the article "*PC8: Project performance*" on page 217. However, the Earned Schedule technique allows to measure alternative versions of the schedule performance index and variance, as follows:

- **Schedule Variance (SV(t))**: Shows the variance in time (ES − AT) expressed in time units

  $> 0$: project ahead of schedule
  $= 0$: project on time
  $< 0$: project delay

[8] This section is also available as an article entitled "Earned Value Management: Reliable time performance measurement" published online at PM Knowledge Center.

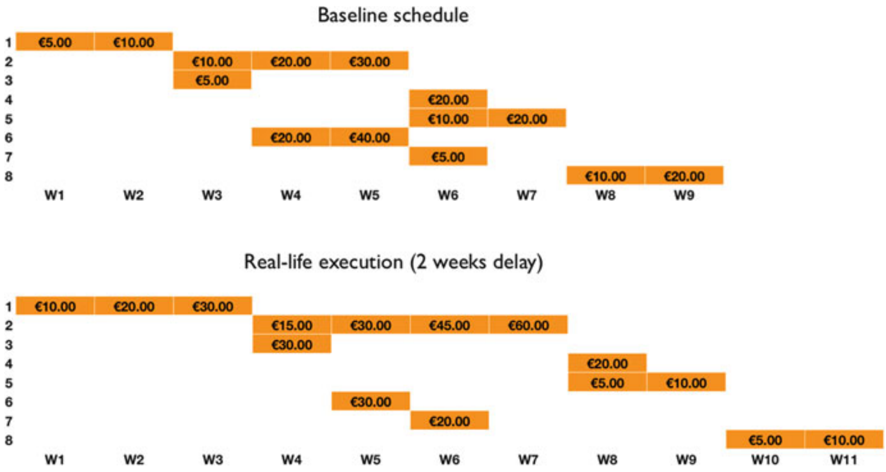



**Fig. 10.11** An example project baseline schedule (*top*) and real-life progress (*bottom*) with 2 weeks delay

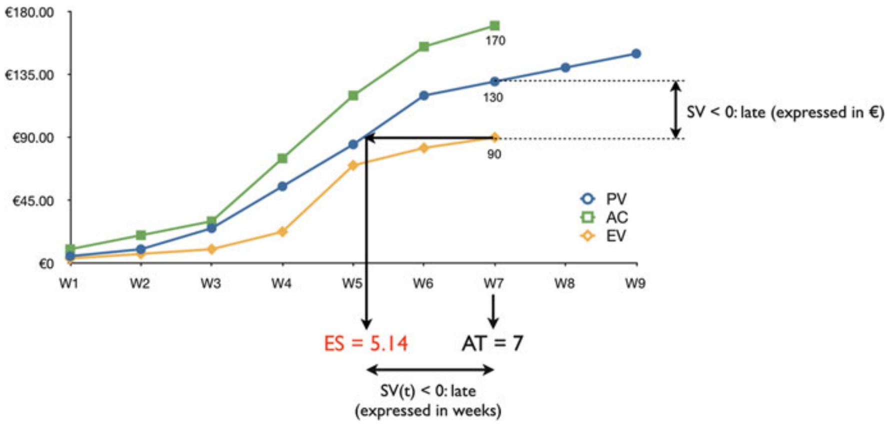



**Fig. 10.12** Real life execution (project progress at week 7)

- **Schedule Performance Index (SPI(t))**: Shows the performance of time (ES/AT) in a unitless dimension

  $>$ 100 %: project ahead of schedule
  $=$ 100 %: project on time
  $<$ 100 %: project delay

with AT = the current Actual Time (e.g. week 7 in Fig. 10.12).

Figure 10.12 shows the schedule variances SV and SV(t) on the example project of Fig. 10.11 at week 7. While SV measures the schedule performance not in units of time, but rather in costs, the SV(t), instead, measures the project progress in a time dimension. However, the earned schedule technique, and its alternative performance measures, have another advantage that is more important than the simple expression in time units instead of costs: It solves the unreliable behavior of the traditional SPI metric.

Table 10.3 shows the EVM metrics along the project duration (11 weeks) of Fig. 10.11. The table shows the following observations at the end (week 11) of the project:

- PV varies between € 0 (at the start of the project) and the budget at completion (BAC) at the end of the project.
- EV varies between € 0 (at the start of the project) and the budget at completion (BAC) at the end of the project.
- ES varies between 0 time units (at the start of the project) and the baseline Planned Duration (PD) at the end of the project.

Hence, at the end of the project, EV = PV and ES = PD. Based on this fact, the performance indicators show the following behavior at the project end:

- SV = EV − PV = BAC − BAC = 0 (when project is on time or late)
- SPI = EV/PV = BAC/BAC = 1 (when project is on time or late)
- SV(t) = ES − AT = PD − AT

  > 0, if AT < ES (project is early)
  = 0, if AT = ES (project is on time)
  < 0, if AT > ES (project is late)

- SPI(t) = ES/AT = PD/AT

  > 1, if AT < ES (project is early)
  = 1, if AT = ES (project is on time)
  < 1, if AT > ES (project is late)

Consequently, since the Earned Value metric EV is always equal to the Planned Value metric PV at the end of the project, the SPI always ends at 100 %, regardless of the real project state (on time or late). This unreliable trend of the SPI towards 1 at the end of the project has been the topic of many discussions and research projects, which is solved when using the alternative SV(t) and SPI(t) performance measures. Figure 10.13 displays the SPI and SPI(t) for the example project.

## Questions

1. If the schedule performance index SPI(t) is lower than 100 %, the project is . . .

   (a) behind schedule.
   (b) on time.
   (c) ahead of schedule.

**Table 10.3** The EVM data for the example project of Fig. 10.11

| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| AT | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| PV | 5.00 | 10.00 | 25.00 | 55.00 | 85.00 | 120.00 | 130.00 | 140.00 | 150.00 | 150.00 | 150.00 |
| AC | 10.00 | 20.00 | 30.00 | 75.00 | 120.00 | 155.00 | 170.00 | 195.00 | 200.00 | 205.00 | 210.00 |
| EV | 3.33 | 6.67 | 10.00 | 22.50 | 70.00 | 82.50 | 90.00 | 120.00 | 130.00 | 140.00 | 150.00 |
| ES | 0.67 | 1.33 | 2.00 | 2.83 | 4.50 | 4.92 | 5.14 | 6.00 | 7.00 | 8.00 | 9.00 |
| SV | −1.67 | −3.33 | −15.00 | −32.50 | −15.00 | −37.50 | −40.00 | −20.00 | −20.00 | −10.00 | 0.00 |
| SPI | 0.67 | 0.67 | 0.40 | 0.41 | 0.82 | 0.69 | 0.69 | 0.86 | 0.87 | 0.93 | 1.00 |
| SV(t) | −0.33 | −0.67 | −1.00 | −1.17 | −0.50 | −1.08 | −1.86 | −2.00 | −2.00 | −2.00 | −2.00 |
| SPI(t) | 0.67 | 0.67 | 0.67 | 0.71 | 0.90 | 0.82 | 0.73 | 0.75 | 0.78 | 0.80 | 0.82 |

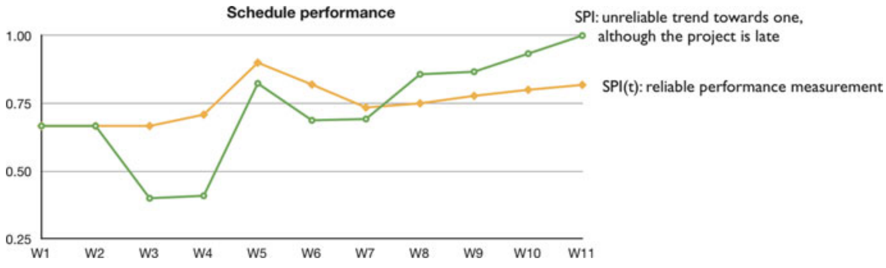



**Fig. 10.13** SPI versus SPI(t)

2. If the schedule variance SV(t) is equal to one, the project is . . .
   (a) behind schedule.
   (b) on time.
   (c) ahead of schedule.
3. Which one of the following statements is true?
   (a) The SV is expressed in units of time, whereas the SV(t) is not.
   (b) The SV(t) is expressed in units of time, whereas the SV is not.
   (c) Both the SV and the SV(t) are expressed in units of time.
   (d) None of the above.
4. Consider Fig. 10.6 on page 210. At week 5, the earned schedule (ES) is . . .
   (a) lower than 3.
   (b) between 3 and 4.
   (c) between 4 and 5.
   (d) higher than 5.
5. When a project is late, the schedule variance SV at the end of the project is always equal to zero.
   (a) True
   (b) False
6. Which one of the following statements is true?
   (a) If SPI $= 1$ at the end of a project, SPI(t) $= 1$ as well.
   (b) If SPI(t) $= 1$ at the end of a project, SPI $= 1$ as well.
   (c) SPI(t) is always equal to one at the end of an early project.
   (d) None of the above.

# EVM Forecasting

All calculations for this section have been made in MS Excel without rounding. However, since the results in Tables 10.5, 10.6, 10.7, and 10.8 are displayed with rounding, some slightly different numbers might appear when using manual calculations. To that purpose, these tables are available in MS Excel at www.or-as.be/books/pmkc.

## *PC10: Forecasting*[9]

This article builds further on the Earned Value Management (EVM) concepts summarized in the articles "*PC2: EVM Overview*" on page 199 and "*PC3: EVM formulary*" on page 203. More precisely, the EVM forecasting approach to predict the final project duration and cost, given the current project performance, is discussed. The basic assumption of project forecasting in EVM is that the current project performance, measured by the SPI and SPI(t) metrics for time and the CPI for cost, is representative for future performance and is therefore used to predict the final duration and cost of a project. The general abbreviations to forecast time and cost are as follows:

- Expected At Completion—Cost (EAC(€)): Forecast of total project cost at the current status date, which might be different from the Budget At Completion (BAC).
- Expected At Completion—Time (EAC(t)): Forecast of final project duration at the current status date, which might differ from the baseline Planned Duration (PD).

In this article, three main topics on EVM forecasting are discussed as follows:

- General cost forecasting formula
- General time forecasting formula
- Performance factor

### Cost Forecasting EAC(€)

Forecasting the final cost of the project is key to the success or failure of a project since it allows to take corrective actions when the predicted EAC(€) exceeds a certain threshold. This forecast is based on the actual costs already spent and a

[9]This section is also available as an article entitled "Earned Value Management: Forecasting project outcome" published online at PM Knowledge Center.

prediction of the future spendings for the remaining portion of work to be done. The general forecasting formula is equal to:

EAC(€) = AC + PCWR
with

| | |
|---|---|
| AC | The actual cost at the current time moment (i.e. AT = actual time) |
| PCWR | Planned Cost of Work Remaining as an estimate for the future |

More information on the cost forecasting formulas and techniques are discussed in the article "*PC12: Forecasting cost*" on page 233.

### Time Forecasting EAC(t)

Forecasting the final duration of the project is key to the success or failure of a project since it allows to take corrective actions when the predicted EAC(t) exceeds a certain threshold. This forecast is based on the actual time already spent on the work done and a prediction of the duration of the remaining portion of work to be done. The general forecasting formula is equal to:

EAC(t) = AT + PDWR
with

| | |
|---|---|
| AT | The actual time moment (i.e. today) |
| PDWR | Planned Duration of Work Remaining |

More information on the time forecasting formulas and techniques can be found in the article "*PC11: Forecasting time*" on page 228.

Figure 10.14 displays the three key metrics, Planned Value, Actual Cost and Earned Value of a project at a certain current moment in time AT. The EV is equal to the PV at the end of the project and hence, the current EV (which shows that the project is late) is used to predict at what moment in time the EV equals PV. The total project duration prediction is shown in the figure by

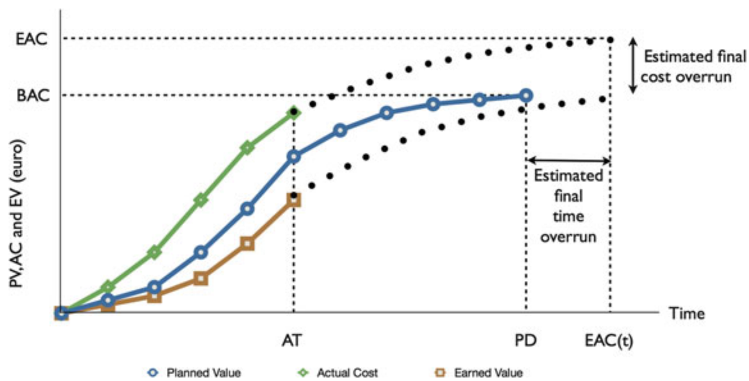



**Fig. 10.14** Forecasting cost (EAC(€)) and time (EAC(t))

the EAC(t), which shows the project is expected to be late. Likewise, the current actual costs are higher than the EV, which shows a budget overrun at the current moment AT. The AC is expected to grow at the same rate until the project is expected to be finished and results in an EAC(€) showing an expected final cost overrun.

### Performance Factor PF

The performance factor (PF) is a way to express that the rate in which the increase in AC or EV (the dotted lines in Fig. 10.14) occurs might depend on the assumption of the performance of the remaining work to be done. Consequently, it refers to the assumption about the expected performance of the future work PDWR or PCWR, as follows:

- PF = 1: Future performance is expected to follow the baseline schedule.
- PF = SPI or SPI(t): Future performance is expected to follow the current time performance.
- PF = SCI or SCI(t): Future performance is expected to follow the current time and cost performance.

with the Schedule Cost Index defined as SCI = SPI * CPI or SCI(t) = SPI(t) * CPI.

All these methods provide an estimate for the total project duration and cost, and offer a range of possibilities, and hence, a lower and upper bound on your predicted total time and cost. The choice of a specific forecasting method depends on the project, the expertise of the project manager and many other unknown factors.

### Questions

1. Which statement is most accurate?
   (a) The main objective of project control is to generate early warning signals to timely detect unfavorable project performance, such that actions can be taken to bring the project back on schedule.
   (b) The main objective of project control is to generate early warning signals to timely detect favorable or unfavorable project performance such that actions can be taken to address problems or to exploit opportunities.
2. The expected cost at completion EAC(€) is always equal to the budget at completion BAC.
   (a) True
   (b) False

3. Graphically, the expected cost and time overruns or underruns can be observed ...
   (a) on the vertical and the horizontal axis respectively.
   (b) on the horizontal and vertical axis respectively.
4. The performance of cost and time are per definition inversely related. For example, if cost is expected to exceed the budget, the project is always estimated to finish ahead of schedule.
   (a) True
   (b) False
5. Assuming that future performance is expected to follow the current time performance when forecasting the total cost. This approach ...
   (a) will always give inaccurate results.
   (b) can be a correct approach, and depends on the project characteristics.
6. The schedule cost index SCI is based on both cost and time performance.
   (a) True
   (b) False

## *PC11: Forecasting Time*[10]

In the article "*PC10: Forecasting*" on page 225, it has been shown that the general forecasting formula for predicting the final duration, known as the Expected At Completion—Time (EAC(t)), is equal to:

EAC(t) = AT + PDWR
with

| | |
|---|---|
| AT | The actual time moment (i.e. today) |
| PDWR | Planned Duration of Work Remaining |

The way the PDWR is calculated depends on the performance factor PF, which refers to the assumption made about the expected performance of the future work, as follows:

- PF = 1: Future performance is expected to follow the baseline schedule.
- PF = SPI or SPI(t): Future performance is expected to follow the current time performance.
- PF = SCI or SCI(t): Future performance is expected to follow the current time and cost performance.

with SCI the Schedule Cost Index (SCI = SPI * CPI and SCI(t) = SPI(t) * CPI).

[10]This section is also available as an article entitled "Earned Value Management: Forecasting time" published online at PM Knowledge Center.

**Table 10.4** Overview of EVM time forecasts (3 versions x 3 performance factors)

| Forecasting method | Version 1: According to plan | Performance factor (PF) |
|---|---|---|
| Planned value method | $EAC(t)_{PV_1}$ | 1 |
| Earned duration method | $EAC(t)_{ED_1}$ | 1 |
| Earned schedule method | $EAC(t)_{ES_1}$ | 1 |
| Forecasting method | Version 2: According to current time performance | Performance factor (PF) |
| Planned value method | $EAC(t)_{PV_2}$ | SPI |
| Earned duration method | $EAC(t)_{ED_2}$ | SPI |
| Earned schedule method | $EAC(t)_{ES_2}$ | SPI(t) |
| Forecasting method | Version 3: According to current time/cost performance | Performance factor (PF) |
| Planned value method | $EAC(t)_{PV_3}$ | SCI |
| Earned duration method | $EAC(t)_{ED_3}$ | SCI |
| Earned schedule method | $EAC(t)_{ES_3}$ | SCI(t) |

In this article, more detailed information is given on the time forecasting formulas and techniques. More precisely, three different EVM time forecasting techniques are discussed as follows:

- Planned value method
- Earned duration method
- Earned schedule method

Table 10.4 gives an overview of the three forecasting methods and the three different versions. Each version differs in the calculation of the PDWR and the performance factor. The nine different EAC(t) formulas are explained in the remainder of this article and illustrated on project data summarized in Table 10.5.

## Planned Value Method

The planned value method calculates the PDWR somewhat differently, resulting in three different EAC(t) calculations that do not follow the general EAC(t) = AT + PDWR formula.

The first version of the EAC(t) formula is based on two new concepts, known as the planned value rate $PV_{rate}$ and the Time Variance TV, as follows:

$PV_{rate} = BAC/PD$
Time Variance $TV = SV/PV_{rate}$

Version 1: $EAC(t)_{PV_1} = PD - TV$

**Table 10.5** Overview of EVM time forecasts (example)

| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PV | 5.00 | 10.00 | 25.00 | 55.00 | 85.00 | 120.00 | 130.00 | 140.00 | 150.00 | 150.00 | 150.00 |
| EV | 3.33 | 6.67 | 10.00 | 22.50 | 70.00 | 82.50 | 90.00 | 120.00 | 130.00 | 140.00 | 150.00 |
| ES | 0.67 | 1.33 | 2.00 | 2.83 | 4.50 | 4.92 | 5.14 | 6.00 | 7.00 | 8.00 | 9.00 |
| ED | 0.67 | 1.33 | 1.20 | 1.64 | 4.12 | 4.13 | 4.85 | 6.86 | 7.80 | 9.33 | 11.00 |
| TV | −0.10 | −0.20 | −0.90 | −1.95 | −0.90 | −2.25 | −2.40 | −1.20 | −1.20 | −0.60 | 0.00 |
| SPI | 0.67 | 0.67 | 0.40 | 0.41 | 0.82 | 0.69 | 0.69 | 0.86 | 0.87 | 0.93 | 1.00 |
| SPI(t) | 0.67 | 0.67 | 0.67 | 0.71 | 0.90 | 0.82 | 0.73 | 0.75 | 0.78 | 0.80 | 0.82 |
| CPI | 0.33 | 0.33 | 0.33 | 0.30 | 0.58 | 0.53 | 0.53 | 0.62 | 0.65 | 0.68 | 0.71 |
| $EAC(t)_{PV_1}$ | 9.10 | 9.20 | 9.90 | 10.95 | 9.90 | 11.25 | 11.40 | 10.20 | 10.20 | 9.60 | 9.00 |
| $EAC(t)_{PV_2}$ | 13.50 | 13.50 | 22.50 | 22.00 | 10.93 | 13.09 | 13.00 | 10.50 | 10.38 | 9.64 | 9.00 |
| $EAC(t)_{PV_3}$ | 40.50 | 40.50 | 67.50 | 73.33 | 18.73 | 24.60 | 24.56 | 17.06 | 15.98 | 14.12 | 12.60 |
| $EAC(t)_{ED_1}$ | 9.33 | 9.67 | 10.80 | 11.36 | 9.88 | 10.88 | 11.15 | 10.14 | 10.20 | 10.67 | 11.00 |
| $EAC(t)_{ED_2}$ | 13.50 | 13.50 | 22.50 | 22.00 | 10.93 | 13.09 | 13.00 | 10.50 | 10.38 | 10.71 | 11.00 |
| $EAC(t)_{ED_3}$ | 38.50 | 36.50 | 61.50 | 64.00 | 15.16 | 19.32 | 18.33 | 12.06 | 11.13 | 11.05 | 11.00 |
| $EAC(t)_{ES_1}$ | 9.33 | 9.67 | 10.00 | 10.17 | 9.50 | 10.08 | 10.86 | 11.00 | 11.00 | 11.00 | 11.00 |
| $EAC(t)_{ES_2}$ | 13.50 | 13.50 | 13.50 | 12.71 | 10.00 | 10.98 | 12.25 | 12.00 | 11.57 | 11.25 | 11.00 |
| $EAC(t)_{ES_3}$ | 38.50 | 36.50 | 34.50 | 33.02 | 13.57 | 15.36 | 16.92 | 14.50 | 12.96 | 11.83 | 11.00 |

The second and third version re-calculates the planned duration PD by taking the SPI and SCI = SPI * CPI into account, as follows:

Version 2: $EAC(t)_{PV_2} = PD/SPI$
Version 3: $EAC(t)_{PV_3} = PD/SCI$

### Earned Duration Method

This forecasting method relies on the Earned Duration metric, which is equal to ED = AT * SPI. The final project duration using the earned duration method is calculated as

$$EAC(t)_{ED} = AT + \frac{\max(PD, AT) - ED}{PF}$$

The use of max(PD, AT) can be explained as follows:

- Project progress is still early: if AT < PD then PD is used
- Project progress is already late: if AT > PD then AT is used

The three versions of this prediction technique only differ in their performance factor PF that can be equal to 1, SPI or SPI * CPI.

### Earned Schedule Method

This forecasting method relies on the Earned Schedule metric (see article "*PC6: Earned value and schedule*" on page 212) and calculates the final duration prediction as:

$$EAC(t)_{ES} = AT + \frac{PD - ES}{PF}$$

Consequently, the three versions of this prediction technique only differ in their performance factor PF that can be equal to 1, SPI(t) or SPI(t) * CPI.

### Project Example

In Table 10.5, all forecasting methods have been calculated based on fictitious project data with a planned duration PD = 9 weeks, a project finish with 2 weeks delay and a total budget BAC = € 150, as discussed in the article "*PC9: Time performance*" on page 220. The $PV_{rate}$ is equal to € 150/9 = € 16.67 per week. The SPI, SPI(t) and CPI values are also given to allow the reader to calculate the EAC(t) predictions.

It should be noted that the earned schedule methods provide duration forecasts that lie closer to the real duration (11 weeks) than the other methods. However, this table only serves as an example table, and therefore, results cannot be generalized.

More information on the accuracy of forecasting methods can be found in the article "*PC13: Forecast accuracy*" on page 236.

## Questions

1. The schedule variance SV and time variance TV produce the same values if the performance factor equals one.
   (a) True
   (b) False
2. The Schedule Performance Index SPI is a dimensionless metric.
   (a) True
   (b) False
3. The second part in the estimate at completion formula of the earned duration method can never be negative.
   (a) True
   (b) False
4. The performance factor for the earned schedule method can be 1, SPI or SPI * CPI.
   (a) True
   (b) False
5. Which formula is always correct?
   (a) $EAC(t)_{PV_2} = PD/CPI$
   (b) $EAC(t)_{ED_1} = AT + (PD - ED)$
   (c) $EAC(t)_{ES_3} = AT + (PD - ES)/SCI$
   (d) None of the above.
6. Which formula is false?
   (a) $EAC(t)_{PV_1} = PD * (1 - SV/BAC)$
   (b) $EAC(t)_{ED_2} = AT/SPI$ if $AT > PD$
   (c) $EAC(t)_{ES_3} = AT + (PD - EV)/SCI(t)$
   (d) They are all correct.

## *PC12: Forecasting Cost*[11]

In the article "*PC10: Forecasting*" on page 225, it has been shown that the general forecasting formula for predicting the final cost, known as the Expected At Completion—Cost (EAC(€)), is equal to:

EAC(€) = AC + PCWR

with

| | |
|---|---|
| AC | The actual cost at the current time moment (i.e. AT = actual time) |
| PCWR | Planned Cost of Work Remaining as an estimate for the future |

The way the PCWR is calculated depends on the performance factor PF, which refers to the assumption made about the expected performance of the future work, as follows:

- PF = 1: Future performance is expected to follow the baseline schedule (version 1).
- PF = CPI: Future performance is expected to follow the current cost performance (version 2).
- PF = SPI or SPI(t): Future performance is expected to follow the current time performance (version 3).
- PF = SCI or SCI(t): Future performance is expected to follow the current time and cost performance. This method can be used under two versions, i.e. PF is equal to the current SCI performance (version 4) or to a weighted time and cost performance (version 4').

In this article, more detailed information on the cost forecasting formulas and techniques is given. All formulas rely on the general formula where PCWR = BAC − EV corrected for the performance factor, as follows:

$$\text{EAC}(€) = \text{AC} + \frac{(\text{BAC} - \text{EV})}{\text{PF}}$$

Table 10.6 gives an overview of eight forecasting methods. Each version differs in the calculation of the PCWR and the performance factor. The eight different EAC(€) formulas are explained in the remainder of this article and illustrated on project data summarized in Table 10.7. Note that versions 3, 4 and 4' can be used under the traditional Schedule Performance Index (SPI) assumptions, or with the new Earned Schedule based Schedule Performance Index (SPI(t)).

### Project Example

In Table 10.7, all forecasting methods have been calculated based on fictitious project data with a planned duration PD = 9 weeks, a project finish with 2 weeks

[11]This section is also available as an article entitled "Earned Value Management: Forecasting cost" published online at PM Knowledge Center.

**Table 10.6** Overview of EVM cost forecasts

| Version (V) | | Performance factor (PF) |
|---|---|---|
| V1: According to plan | | |
| | $EAC(€)_{V1}$ | 1 |
| V2: According to current cost performance | | |
| | $EAC(€)_{V2}$ | CPI |
| V3: According to current time performance | | |
| | $EAC(€)_{V3-SPI}$ | SPI |
| | $EAC(€)_{V3-SPI(t)}$ | SPI(t) |
| V4: According to current time/cost performance | | |
| | $EAC(€)_{V4-SCI}$ | SCI |
| | $EAC(€)_{V4-SCI(t)}$ | SCI(t) |
| V4': According to weighted time/cost performance | | |
| | $EAC(€)_{V4'-SPI}$ | 0.8*CPI + 0.2 SPI |
| | $EAC(€)_{V4'-SPI(t)}$ | 0.8*CPI + 0.2 SPI(t) |

delay, a total budget BAC = € 150.00 and a real total project cost of € 210.00, as discussed in the article "*PC9: Time performance*" on page 220. The $PV_{rate}$ is equal to € 150/9 = € 16.66 per week. The SPI, SPI(t) and CPI values are also given to allow the reader to calculate the EAC(t) predictions.

It should be noted that the EAC(€) method with a PF = 1 provides cost forecasts that lie closer to the real project cost (€ 210.00) than the other methods. However, this table only serves as an example table, and therefore, results cannot be generalized. More information on the accuracy of forecasting methods can be found in the article "*PC13: Forecast accuracy*" on page 236.

## Questions

1. The planned cost of work remaining PCWR is always expected to follow the baseline schedule.

   (a) True
   (b) False

2. The only difference between SPI and SPI(t) is that EV is replaced by ES in the latter formula.

   (a) True
   (b) False

3. The performance metrics SPI(t) and CPI are both expressed in monetary terms.

   (a) True
   (b) False

**Table 10.7** Overview of EVM cost forecasts (example)

| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PV | 5.00 | 10.00 | 25.00 | 55.00 | 85.00 | 120.00 | 130.00 | 140.00 | 150.00 | 150.00 | 150.00 |
| AC | 10.00 | 20.00 | 30.00 | 75.00 | 120.00 | 155.00 | 170.00 | 195.00 | 200.00 | 205.00 | 210.00 |
| EV | 3.33 | 6.67 | 10.00 | 22.50 | 70.00 | 82.50 | 90.00 | 120.00 | 130.00 | 140.00 | 150.00 |
| ES | 0.67 | 1.33 | 2.00 | 2.83 | 4.50 | 4.92 | 5.14 | 6.00 | 7.00 | 8.00 | 9.00 |
| SPI | 0.67 | 0.67 | 0.40 | 0.41 | 0.82 | 0.69 | 0.69 | 0.86 | 0.87 | 0.93 | 1.00 |
| SPI(t) | 0.67 | 0.67 | 0.67 | 0.71 | 0.90 | 0.82 | 0.73 | 0.75 | 0.78 | 0.80 | 0.82 |
| CPI | 0.33 | 0.33 | 0.33 | 0.30 | 0.58 | 0.53 | 0.53 | 0.62 | 0.65 | 0.68 | 0.71 |
| EAC(€)$_{V1}$ | 156.67 | 163.33 | 170.00 | 202.50 | 200.00 | 222.50 | 230.00 | 225.00 | 220.00 | 215.00 | 210.00 |
| EAC(€)$_{V2}$ | 450.00 | 450.00 | 450.00 | 500.00 | 257.14 | 281.82 | 283.33 | 243.75 | 230.77 | 219.64 | 210.00 |
| EAC(€)$_{V3-SPI}$ | 230.00 | 235.00 | 380.00 | 386.67 | 217.14 | 253.18 | 256.67 | 230.00 | 223.08 | 215.71 | 210.00 |
| EAC(€)$_{V3-SPI(t)}$ | 230.00 | 235.00 | 240.00 | 255.00 | 208.89 | 237.37 | 251.67 | 235.00 | 225.71 | 217.50 | 210.00 |
| EAC(€)$_{V4-SCI}$ | 670.00 | 665.00 | 1,080.00 | 1,113.89 | 286.53 | 339.46 | 333.70 | 251.88 | 235.50 | 220.69 | 210.00 |
| EAC(€)$_{V4-SCI(t)}$ | 670.00 | 665.00 | 660.00 | 675.00 | 272.38 | 309.76 | 324.26 | 260.00 | 239.56 | 223.30 | 210.00 |
| EAC(€)$_{V4'-SPI}$ | 376.67 | 378.33 | 433.85 | 471.19 | 246.71 | 274.83 | 276.76 | 240.20 | 228.85 | 218.64 | 210.00 |
| EAC(€)$_{V4'-SPI(t)}$ | 376.67 | 378.33 | 380.00 | 409.06 | 243.71 | 269.47 | 275.18 | 241.71 | 229.61 | 219.16 | 210.00 |

4. In order to make cost predictions, the $PV_{rate}$ should first be calculated.
   (a) True
   (b) False
5. Which formula is correct?
   (a) $EAC(€)_{V1} = AC + BAC - ES$
   (b) $EAC(€)_{V3-SPI(t)} = AC + (BAC - EV)/SPI(t)$
   (c) $EAC(€)_{V4-SCI(t)} = AC + (BAC - EV)/SCI$
   (d) None of the above.
6. Which formula is false?
   (a) $EAC(€)_{V2} = AC + (BAC - EV)/CPI$
   (b) $EAC(€)_{V3-SPI(t)} = AC + (BAC - ES)/SPI(t)$
   (c) $EAC(€)_{V4-SCI} = AC + (BAC - EV)/SCI$
   (d) They are all correct.

## *PC13: Forecast Accuracy*[12]

Predicting the final project duration and/or cost of a project in progress, given the current project performance, is a crucial step during project control. In an EVM analysis, quite a number of time and cost forecasting techniques are available, however, it is a cumbersome task to select the right technique for the project under study.

In this article, a simple simulation approach is presented to guide the project manager in his/her selection of the most accurate time and cost forecasting technique of the project in progress. The technique evaluates the accuracy of all EVM predictions of a project by means of a computerized simulation model. An overview of the EVM metrics is given in the article "*PC2: EVM Overview*" on page 199 and the formulas are summarized in the article "*PC3: EVM formulary*" on page 203. The approach to evaluate the accuracy of the forecasting techniques available in EVM consists of a three-step procedure, as summarized along the following lines:

- Step 1. Define uncertainty
- Step 2. Simulate project progress
- Step 3. Evaluate the forecast accuracy

[12]This section is also available as an article entitled "Predicting project performance: Evaluating the forecasting accuracy" published online at PM Knowledge Center.

### Define Uncertainty

Uncertainty on the activity durations or costs, resource use, the presence of precedence relations or even on the existence of an activity in the project network is what typifies projects. Therefore, single point estimates should be replaced by probability distributions to incorporate this uncertainty. In doing so, project parameters that are considered to be deterministic are modeled as random variables that enable the project manager to develop a computerized simulation model that imitates project progress. More information can be found in the article "*RA4: Activity distributions*" on page 119.

### Simulate Project Progress

Random numbers are generated from the probability distributions defined earlier to imitate a fictitious project progress run. This process is repeated a (huge) number of times such that each simulated project run is different. The general idea is that each simulation run reflects a realistic project run that can possibly occur when the project is in real progress. More information on the use of simulation runs in project scheduling is given in the article "*RA2: Simulating project progress*" on page 111. A simple simulation approach consisting of eight special simulation scenarios is proposed in the article "*RA3: CPM schedule control*" on page 115.

### Evaluate the Forecast Accuracy

During each project simulation run, the periodic time and cost performance of the project is measured using Earned Value Management data, which are then used to predict the final duration and total real cost of the project. At the end of each project run, the simulated project duration and cost is known and is compared to the planned duration and budgeted cost in order to evaluate the accuracy of these periodic predictions. More information on the general concept of making predictions using EVM data is given in the article "*PC10: Forecasting*" on page 225.

The experienced project manager who is knowledgeable about the use of EVM metrics and predictions can choose among a set of time and cost prediction techniques, as discussed in the articles "*PC11: Forecasting time*" on page 228 and article "*PC12: Forecasting cost*" on page 233. However, despite the rich availability, no guidelines are given on which techniques to use for a specific project.

This is where the Monte Carlo simulation runs come into play. The periodic forecasts can be compared with the known project durations and costs and the accuracy can be evaluated by means of the two following measures:

- Mean Absolute Percentage Error (MAPE): The average of the absolute values of the relative deviations between the periodic time/cost predictions and the final project duration/cost, as follows:
  - $\text{MAPE (time)} = \frac{1}{\#\text{Periods}} * \sum_{\text{All Periods}} \frac{|\text{EAC(t)} - \text{RD}|}{\text{RD}}$
  - $\text{MAPE (cost)} = \frac{1}{\#\text{Periods}} * \sum_{\text{All Periods}} \frac{|\text{EAC(€)} - \text{RC}|}{\text{RC}}$
- Mean Percentage Error (MPE): The average of the relative deviations between the periodic time/cost predictions and the final project duration/cost, as follows:
  - $\text{MPE (time)} = \frac{1}{\#\text{Periods}} * \sum_{\text{All Periods}} \frac{\text{EAC(t)} - \text{RD}}{\text{RD}}$
  - $\text{MPE (cost)} = \frac{1}{\#\text{Periods}} * \sum_{\text{All Periods}} \frac{\text{EAC(€)} - \text{RC}}{\text{RC}}$

with

| | |
|---|---|
| EAC(t) | Time forecast (at each periodic review period) |
| EAC(€) | Cost forecast (at each periodic review period) |
| RD | Real duration (known upon completion of each simulation run) |
| RC | Real cost (known upon completion of each simulation run) |
| # Periods | Number of periodic reviews in EVM<br>(= number of EAC(€) and EAC(t) values) |

Consequently, the MAPE and MPE are calculated in a similar way. While the MAPE gives an indication about the average deviations as a nonnegative value, the MPE calculations can result in both positive and negative values and measure over- or underestimations of the final project duration or cost.

The MAPE and MPE calculations can be easily illustrated on the nine time forecasts of the example given in the article "*PC11: Forecasting time*" on page 228 and the eight cost forecasts given in the article "*PC12: Forecasting cost*" on page 233. For the sake of simplicity, these forecasts are copied in Table 10.8. The planned duration of the project is equal to 9 weeks while the real duration equals 11 weeks (2 weeks delay). The planned cost is equal to € 150 while the real cost equals € 210. In the last two columns, the MPE and MAPE values are calculated for all methods. These calculations are described in detail below for the $\text{EAC(t)}_{\text{PV}_1}$ time forecasting method and the $\text{EAC(€)}_{\text{V1}}$ cost forecasting method.

- MPE (time) = $\{(9.10 - 11) + (9.20 - 11) + (9.90 - 11) + (10.95 - 11) + (9.90 - 11) + (11.25 - 11) + (11.40 - 11) + (10.20 - 11) + (10.20 - 11) + (9.60 - 11) + (9.00 - 11)\}/(11 * 11) = -0.09$

**Table 10.8** Nine periodic time forecasts and eight periodic cost forecasts

| | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | MPE | MAPE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Time forecasting | | | | | | | | | | | | | |
| $EAC(t)_{PV_1}$ | 9.10 | 9.20 | 9.90 | 10.95 | 9.90 | 11.25 | 11.40 | 10.20 | 10.20 | 9.60 | 9.00 | −0.09 | 0.10 |
| $EAC(t)_{PV_2}$ | 13.50 | 13.50 | 22.50 | 22.00 | 10.93 | 13.09 | 13.00 | 10.50 | 10.38 | 9.64 | 9.00 | 0.22 | 0.30 |
| $EAC(t)_{PV_3}$ | 40.50 | 40.50 | 67.50 | 73.33 | 18.73 | 24.60 | 24.56 | 17.06 | 15.98 | 14.12 | 12.60 | 1.89 | 1.89 |
| $EAC(t)_{ED_1}$ | 9.33 | 9.67 | 10.80 | 11.36 | 9.88 | 10.88 | 11.15 | 10.14 | 10.20 | 10.67 | 11.00 | −0.05 | 0.06 |
| $EAC(t)_{ED_2}$ | 13.50 | 13.50 | 22.50 | 22.00 | 10.93 | 13.09 | 13.00 | 10.50 | 10.38 | 10.71 | 11.00 | 0.25 | 0.27 |
| $EAC(t)_{ED_3}$ | 38.50 | 36.50 | 61.50 | 64.00 | 15.16 | 19.32 | 18.33 | 12.06 | 11.13 | 11.05 | 11.00 | 1.47 | 1.47 |
| $EAC(t)_{ES_1}$ | 9.33 | 9.67 | 10.00 | 10.17 | 9.50 | 10.08 | 10.86 | 11.00 | 11.00 | 11.00 | 11.00 | −0.06 | 0.06 |
| $EAC(t)_{ES_2}$ | 13.50 | 13.50 | 13.50 | 12.71 | 10.00 | 10.98 | 12.25 | 12.00 | 11.57 | 11.25 | 11.00 | 0.09 | 0.11 |
| $EAC(t)_{ES_3}$ | 38.50 | 36.50 | 34.50 | 33.02 | 13.57 | 15.36 | 16.92 | 14.50 | 12.96 | 11.83 | 11.00 | 0.97 | 0.97 |
| Cost forecasting | | | | | | | | | | | | | |
| $EAC(€)_{V1}$ | 156.67 | 163.33 | 170.00 | 202.50 | 200.00 | 222.50 | 230.00 | 225.00 | 220.00 | 215.00 | 210.00 | −0.04 | 0.10 |
| $EAC(€)_{V2}$ | 450.00 | 450.00 | 450.00 | 500.00 | 257.14 | 281.82 | 283.33 | 243.75 | 230.77 | 219.64 | 210.00 | 0.55 | 0.55 |
| $EAC(€)_{V3-SPI}$ | 230.00 | 235.00 | 380.00 | 386.67 | 217.14 | 253.18 | 256.67 | 230.00 | 223.08 | 215.71 | 210.00 | 0.23 | 0.23 |
| $EAC(€)_{V3-SPI(t)}$ | 230.00 | 235.00 | 240.00 | 255.00 | 208.89 | 237.37 | 251.67 | 235.00 | 225.71 | 217.50 | 210.00 | 0.10 | 0.10 |
| $EAC(€)_{V4-SCI}$ | 670.00 | 665.00 | 1,080.00 | 1,113.89 | 286.53 | 339.46 | 333.70 | 251.88 | 235.50 | 220.69 | 210.00 | 1.34 | 1.34 |
| $EAC(€)_{V4-SCI(t)}$ | 670.00 | 665.00 | 660.00 | 675.00 | 272.38 | 309.76 | 324.26 | 260.00 | 239.56 | 223.30 | 210.00 | 0.95 | 0.95 |
| $EAC(€)_{V4'-SPI}$ | 376.67 | 378.33 | 433.85 | 471.19 | 246.71 | 274.83 | 276.76 | 240.20 | 228.85 | 218.64 | 210.00 | 0.45 | 0.45 |
| $EAC(€)_{V4'-SPI(t)}$ | 376.67 | 378.33 | 380.00 | 409.06 | 243.71 | 269.47 | 275.18 | 241.71 | 229.61 | 219.16 | 210.00 | 0.40 | 0.40 |

- MAPE (time) $= \{|9.10-11|+|9.20-11|+|9.90-11|+|10.95-11|+|9.90-11|+|11.25-11|+|11.40-11|+|10.20-11|+|10.20-11|+|9.60-11|+|9.00-11|\}/(11*11) = 0.10$
- MPE (cost) $= \{(156.67-210.00)+(163.33-210.00)+(170.00-210.00)+(202.50-210.00)+(200.00-210.00)+(222.50-210.00)+(230.00-210.00)+(225.00-210.00)+(220.00-210.00)+(215.00-210.00)+(210.00-210.00)\}/(11*210.00) = -0.04$
- MAPE (cost) $= \{|156.67-210.00|+|163.33-210.00|+|170.00-210.00|+|202.50-210.00|+|200.00-210.00|+|222.50-210.00|+|230.00-210.00|+|225.00-210.00|+|220.00-210.00|+|215.00-210.00|+|210.00-210.00|\}/(11*210.00) = 0.10$

These calculations show that the time forecast has an absolute deviation of 10 % with an average underestimation (MPE = −9 %). Similarly, the cost forecast is on average slightly underestimated (MPE = −0.04) with an absolute deviation of 10 %. The MAPE and MPE values in Table 10.8 show that the best time forecasting method is $EAC(t)_{ES_1}$ and best performing cost forecasting method is $EAC(€)_{V3-SPI(t)}$, which is obviously not a general conclusion but only holds for the data in the table.

It should be noted that this illustration is somewhat biased and can certainly not be generalized since it does not rely on a simulation with multiple project progress runs, but instead the values in the table are given for only one project progress run. In order to evaluate the forecasting accuracy in a more clever way, multiple project progress runs should be performed, such that the averages of all MAPE and MPE values can lead to a much more reliable decision to select the technique that fits best for the project.

## Questions

1. Single point estimates are able to incorporate uncertainty inherent to each project.
   (a) True
   (b) False
2. The MAPE indicates average deviations as a nonnegative value, while the MPE can be both positive as negative.
   (a) True
   (b) False
3. The MAPE for a certain number of periods will always be larger than or equal to the absolute value of the corresponding MPE.
   (a) True
   (b) False

4. Assume a project that is finished with a real duration of 10 days. During the three periodic review periods, the EAC(t) values are equal to 9, 10 and 9 for periods 1, 2 and 3 respectively. Which of the following calculations is correct?
   (a) MAPE = 0.0667 and MPE = 0.
   (b) MAPE = 0.0667 and MPE = 0.0667.
   (c) MAPE = 0.0667 and MPE = −0.0667.
   (d) None of the above is true.
5. Assume a project that is finished with a real cost of € 150. During the three periodic review periods, the EAC(€) values are equal to 140, 150 and 160 for periods 1, 2 and 3 respectively. Which of the following calculations is correct?
   (a) MAPE = 0.0444 and MPE = 0.
   (b) MAPE = 0.0444 and MPE = 0.0444.
   (c) MAPE = 0 and MPE = −0.0444.
   (d) None of the above is true.

## Schedule Adherence

### *PC14: Schedule Adherence*[13]

This article builds further on the Earned Value Management (EVM) concepts summarized in the articles "*PC2: EVM Overview*" on page 199 and "*PC3: EVM formulary*" on page 203. More precisely, a concept known as the p-factor will be discussed to measure the schedule adherence of a project in progress. The concept is based on the earned schedule metric as discussed in article "*PC6: Earned value and schedule*" on page 212.

#### Schedule Adherence

While the performance measures (SPI, SPI(t) and CPI) used in Earned Value Management compare the current project progress relative to the baseline schedule, the schedule adherence concept takes a different view on the current progress of the project. It does not evaluate the project performance against a static baseline schedule, but instead, relies on the dynamic earned schedule metric to measure the progress.

The rationale behind the schedule adherence concept lies in the observation that performing work not according to the baseline schedule often indicates activity

[13]This section is also available as an article entitled "Earned Value Management: Measuring schedule adherence" published online at PM Knowledge Center.

impediments or is likely a cause of rework. Consequently, the basic idea is that whenever impediments occur (activities that are performed relatively less efficiently compared to the project progress), resources are shifted from these constrained activities to other activities where they could gain earned value. However, this results in a project execution that deviates from the original baseline schedule and might involve a certain degree of risk. Indeed, the latter activities are performed without the necessary inputs, and could result into a certain portion of rework. Based on these observations, the p-factor has been introduced as a measure to provide the connection of project output to EVM. It measures the portion of earned value accrued in congruence with the baseline schedule, i.e. the tasks that ought to be either completed or in progress. A formal definition of the p-factor can be given as follows:

$$\text{p-factor} = \frac{\sum_{i \in N} \min(\text{PV}_{i,\text{ES}}, \text{EV}_{i,\text{AT}})}{\sum_{i \in N} \text{PV}_{i,\text{ES}}}$$

with

| | |
|---|---|
| $N$ | The set of all activities of the project network |
| $\text{PV}_{i,\text{ES}}$ | The planned value of activity $i$ at time ES |
| $\text{EV}_{i,\text{AT}}$ | The earned value of activity $i$ at time AT |

and

ES > AT (early), ES = AT (on time) or ES < AT (late).

Figure 10.3 on page 207 displays the cumulative planned values of the baseline schedule with a total planned duration PD = 9 weeks and a budget at completion BAC = € 150.00 (see the article "*PC4: Planned value*" on page 205 for the calculations).

Figure 10.15 displays the project progress up to week 7 (i.e. actual time AT = week 7), with activity delays in activities 1 and 2 and a faster execution for activity 6. The periodic cumulative earned value is given in each bar of these activities. More detailed calculations can be found in the article "*PC5: Key metrics*" on page 208.

In the article "*PC6: Earned value and schedule*" on page 212, it has been shown that the earned schedule metric can be easily calculated as ES = 5.14, which means that the project is clearly late.

Based on the data given in Fig. 10.3 (PV) and Fig. 10.15 (EV), the p-factor can be calculated at the actual time AT = week 7 as p-factor = {min(10,10) + min(30,


**Fig. 10.15** Real project progress up to week 7 (numbers are periodic earned values)

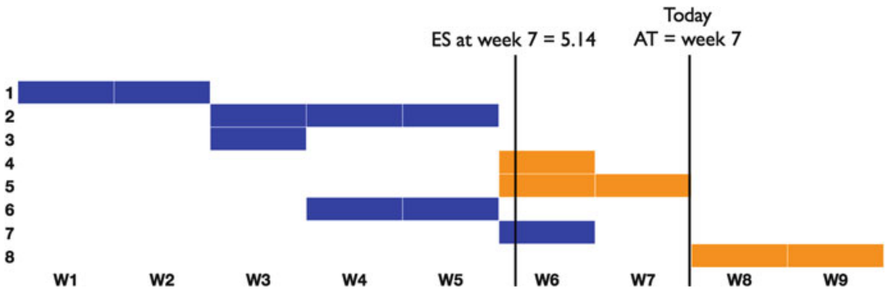



**Fig. 10.16** The schedule adherence concept shown on the baseline schedule

30) + min(5, 5) + min(0.14*20, 0) + min(0.14*10, 0) + min(40, 40) + min(0.14*5, 5) + min(0, 0)}/{10 + 30 + 5 + 0.14*20 + 0.14*10 + 40 + 0.14*5 + 0} = 85.7/89.9 = 0.95.

Figure 10.16 displays the baseline schedule with the activity progress in blue or dark grey. The actual time AT = 7 while the ES = 5.14. Since the earned schedule ES expresses the time where the current amount of earned value should have been earned, activities 4 and 5 have made relatively little progress compared to the normal progress expressed by the ES metric while activity 7 has made more progress than originally captured in the baseline schedule at time 5.14. This is the reason why schedule adherence is only equal to 95 %, which is less than 100 % (perfect schedule adherence).

According to the schedule adherence concept, it is said that activities 4 and 5 show certain impediments while activity 7 has been performed under a certain degree of risk and might be the subject of rework. More information on the p-factor approach can be found in the articles "*PC15: Effective earned value*" on page 244 and "*PC16: Schedule inadherence*" on page 247.

## Questions

1. The higher the value for the p-factor, the more the actual project execution deviates from the original baseline schedule.
   (a) True
   (b) False
2. The p-factor can never be higher than 1.
   (a) True
   (b) False
3. At the end of a project, the p-factor is always equal to 1.
   (a) True
   (b) False

4. Consider the Gantt chart in Fig. 10.15 on page 242 that indicates the project progress as indicated by the blue or dark grey part at week 7. Assume that the remaining work will be done as planned (see Fig. 10.3 on page 207). The p-factor at the end of week 7 is equal to 0.95, but what is the p-factor at the end of week 5? (Tip: The ES at the end of week 5 was equal to 4.50.)

   (a) 0.54
   (b) 0.66
   (c) 0.71
   (d) 0.86

5. Consider Fig. 10.16 on page 243 and assume that the current date is equal to the end of week 7. Assume two small changes to the figure, i.e. (1) activity 7 has not started yet, and (2) € 5.00 was earned by completing a portion (i.e. 25 %) of activity 4. These two changes will not change the values for EV and ES at week 7, but the p-factor will . . .

   (a) increase.
   (b) decrease.
   (c) remain the same.

## *PC15: Effective Earned Value*[14]

In this article, the schedule adherence concept proposed in the article "*PC14: Schedule adherence*" on page 241 is used to calculate the so-called effective earned value EV(e). To that purpose, the following two topics will be discussed:

- Schedule adherence: Calculation of the p-factor.
- Effective earned value: Adapt current earned value for work done under risk.

### Schedule Adherence

Measuring the adherence of a baseline schedule is presented in the article "*PC14: Schedule adherence*" on page 241 where the p-factor has been introduced as a tool to identify activity impediments or likely causes of rework. The idea is based on the earned schedule concept and compares the progress of the project measured by the accrue in earned value against the earned schedule at the current moment in time AT. The calculation of the p-factor will be illustrated on a fictitious

[14]This section is also available as an article entitled "Earned Value Management: Schedule adherence and the effective earned value" published online at PM Knowledge Center.

example baseline schedule of Fig. 10.16, where the values for ES and AT are equal to:

- Actual time AT = 7 weeks
- Earned value EV = € 90.00
- Earned schedule ES = 5.14 weeks
- p-factor = 0.95

The p-factor will be used to calculate the effective earned value EV(e).

**Effective Earned Value EV(e)**

The p-factor assumes that lack of schedule adherence is caused by a combination of the presence of impediments or constraints and work performed under risk. Figure 10.16 shows an intermediate project progress state at the Actual Time AT = 7. The EV accrued at the current time AT = 7 is given in blue or dark grey and the ES = 5.14. The figure visualizes the p-factor as follows:

- The portion of the work to the left of the ES line is assumed to be performed without risk and indicates the presence of an impediment or project constraint.
- The portion of work to the right of the ES line indicates work that is ahead of the normal project performance and is assumed to hold a certain degree of risk.
- The p-factor is equal to the EV (blue or dark grey bars) to the left of ES divided by the total EV (all blue or dark grey bars to the left and right of the ES line).

The activity impediments and work performed under risk are illustrated in Fig. 10.17.

It is assumed that this degree of risk is the result of an inefficient use of resources, which were shifted from the constrained activities to less constrained activities where the resources could gain earned value. However, these shifted resources work without the necessary inputs possibly resulting in a certain portion of rework (i.e. risk). The p-factor is a measure to express the portion of the EV without risk (referred to as EV(p), while the remaining portion is denoted as EV(r)).

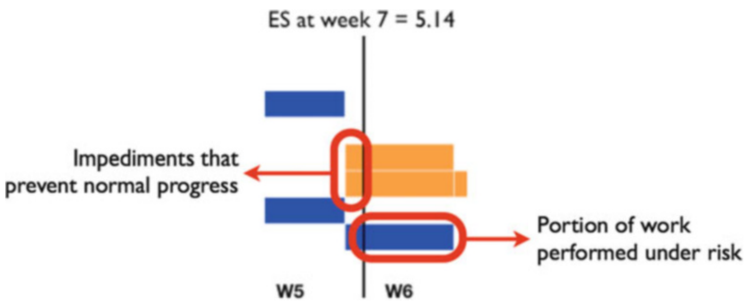



**Fig. 10.17** Activity impediments and work performed under risk as identified by the p-factor

A project manager should realize that the remaining EV(r) portion might be subject to risk and possibly results in rework. The effective earned value EV(e) is defined as the risk-adapted portion of earned value that is performed within the expected baseline schedule performance, taking into account that only R % of the EV(r) will be accounted as risk-free. Mathematically, these p-factor assumptions can be summarized as follows:

$$\mathrm{EV} = \mathrm{p} * \mathrm{EV} + (1 - \mathrm{p}) * \mathrm{EV} = \mathrm{EV(p)} + \mathrm{EV(r)}$$
$$\rightarrow \mathrm{EV(e)} = \mathrm{EV(p)} + \mathrm{R}\,\% * \mathrm{EV(r)}$$

with

| | |
|---|---|
| EV | Earned value |
| EV(p) | Risk-free earned value |
| EV(r) | Remaining earned value portion performed under risk |
| EV(e) | Effective earned value |
| R % | Estimated portion of EV(r) that is usable and requires no rework |

When R % is assumed to be 10 %, the effective earned value EV(e) is equal to 0.95 * € 90 + 0.10 * 0.05 * € 90 = € 85.95. In the article "*PC16: Schedule inadherence*" on page 247, possible causes for lack of schedule adherence are discussed.

## Questions

1. A p-factor below unity indicates a lack of perfect schedule adherence.
   (a) True
   (b) False
2. When a project exhibits a p-factor of 1, the earned value EV is equal to the effective earned value EV(e).
   (a) True
   (b) False
3. The effective earned value EV(e) can never be higher than the earned value EV.
   (a) True
   (b) False
4. The higher the risk of rework in a project, the higher R % should be set in the calculation of EV(e).
   (a) True
   (b) False

5. Consider Fig. 10.16 on page 243. Which one of the following statements is false?
   (a) The portion of the work to the left of the ES line is assumed to be performed without risk and indicates the presence of an impediment or project constraint.
   (b) The portion of work to the right of the ES line indicates work that is ahead of the normal project performance and is assumed to hold a certain degree of risk.
   (c) The p-factor is equal to the EV (blue or dark grey bars) to the left of ES divided by the total EV (all blue or dark grey bars to the left and right of the ES line).
   (d) All statements are true.
6. In Fig. 10.16, the activity/activities for which a portion of the work is performed under risk (at week 7) is/are . . .
   (a) activities 4 and 5.
   (b) activity 7.
   (c) activity 8.
   (d) activities 4, 5 and 8.

## *PC16: Schedule Inadherence*[15]

This article builds further on the Earned Value Management (EVM) concepts summarized in the articles "*PC2: EVM Overview*" on page 199 and "*PC3: EVM formulary*" on page 203. EVM has been the topic of many extensions, among which the p-factor is a relatively recent one that is used to calculate the so-called schedule adherence of a project in progress. In this article, the following two topics will be discussed:

- Schedule adherence: Brief review of the concept measured by the p-factor and tool to calculate the effective earned value.
- Project progress scenarios: Three example scenarios are given why the schedule adherence could be lower than 100 % (perfect).

### Schedule Adherence

Measuring the adherence of a baseline schedule is presented in the article "*PC14: Schedule adherence*" on page 241 where the p-factor concept has been introduced

[15]This section is also available as an article entitled "Earned Value Management: Identifying the lack of schedule adherence" published online at PM Knowledge Center.

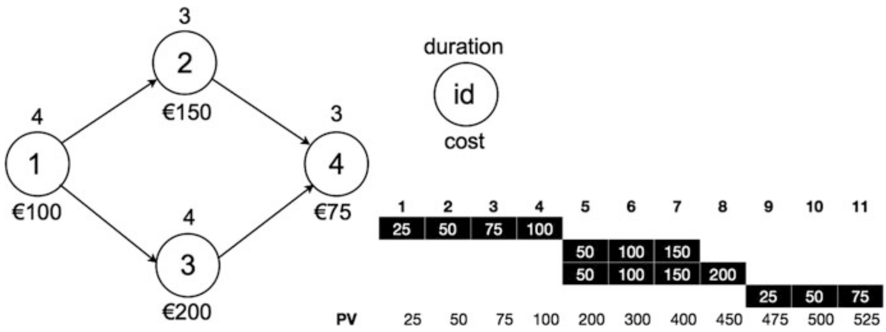



**Fig. 10.18** A four-activity project network and its baseline schedule

as a tool to identify activity impediments or likely causes of rework. In the article "*PC15: Effective earned value*" on page 244, this p-factor is used to adapt the earned value by taking into account the work performed under risk. More precisely, it incorporates the possibility of rework and therefore calculates the effective earned value as the risk-free earned value and the remaining earned value portion performed under risk adapted with the probability for rework.

In this article, three possible scenarios will be discussed as illustrations of why the schedule adherence could deviate from the ideal situation (i.e. p-factor values less than 100 %). To that purpose, the p-factor is calculated for a fictitious four-activity project network as displayed in Fig. 10.18, with the number above each node denoting the activity duration estimate and the number below the node indicating the associated cost. The figure also displays the baseline schedule with a total project duration PD = 11 weeks and a budget at completion BAC = € 525.

## Imperfect Schedule Adherence

The schedule adherence concept as measured by the p-factor is based on the earned schedule concept and compares the progress of the project measured by the accrue in earned value against the earned schedule at the current moment in time AT. P-factor values lower than 100 % denote a lack of perfect schedule adherence and can be caused by various deviations from the original baseline schedule logic. In the remainder of this article, fictitious project progress is simulated using three scenarios to show possible reasons for a lack of perfect schedule adherence.

**Activity Overlapping:** The p-factor concept does not take the project network and its precedence relations into account, but instead takes a general project view on the schedule adherence based on earned value, planned value and earned schedule calculations. Nevertheless, the example in Fig. 10.19 shows that activity overlaps between activities 1 and 2 and between activities 3 and 4 lead to p-values lower than 100 %. Activity overlaps give rise to activities that start earlier than allowed since

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 25 | 50 | 75 | 100 | | | | | | | |
| | | | 50 | 100 | 150 | | | | | | |
| | | | | | | 50 | 100 | 150 | 200 | | |
| | | | | | | | | | 25 | 50 | 75 |
| **EV** | 25 | 50 | 125 | 200 | 250 | 300 | 350 | 400 | 475 | 500 | 525 |
| **ES** | 1 | 2 | 4.25 | 5 | 5.5 | 6 | 6.5 | 7 | 9 | 10 | 11 |
| **p** | 1 | 1 | 0.7 | 0.75 | 0.7 | 0.83 | 0.93 | 1 | 1 | 1 | 1 |
| **Time** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** |

**Fig. 10.19** Activity overlapping between activities 1 and 2 and 3 and 4

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 15 | 35 | 65 | 100 | | | | | | | |
| | | | | | 25 | 75 | 150 | | | | |
| | | | | | 100 | 175 | 200 | 200 | | | |
| | | | | | | | | | 50 | 60 | 75 |
| **EV** | 15 | 35 | 65 | 100 | 225 | 350 | 450 | 450 | 500 | 510 | 525 |
| **ES** | 0.6 | 1.4 | 2.6 | 4 | 5.25 | 6.5 | 8 | 8 | 10 | 10.4 | 11 |
| **p** | 1 | 1 | 1 | 1 | 0.83 | 0.86 | 1 | 1 | 1 | 1 | 1 |
| **Time** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** |

**Fig. 10.20** Example where the accrue in PV and EV differs

their predecessor activity is not finished yet. This is probably the cause of a lack of information and an increased risk of rework. Figure 10.19 shows that the p-factor is able to detect this situation.

**PV/EV Accrue Deviations:** Activities that are completed within their estimated time and budget are not necessarily performed in congruence with their predefined planned value. Since the p-factor is a concept to measure the degree of adherence to the baseline schedule, expressed as a relation between the project progress (Earned Value) and the baseline schedule (Planned Value), it should be able to give an indication of the deviation between PV and EV. In Fig. 10.20, the EV accrued during the real life project progress is not in line with the PV curve of the baseline schedule, although the project finishes exactly on time. The PV is assumed to increase linearly along the activity duration (see Fig. 10.18), which is somewhat different for the activity progress in Fig. 10.20. The p-factor measures this lack of schedule adherence.

**Ahead of Schedule and/or Delayed Project Execution:** Obviously, deviations from the original baseline schedule during project progress lead to a project ending ahead or late. This lack of schedule adherence should be measured and reported by the p-factor concept in order to serve as a dynamic tool to forecast the final project status. Figure 10.21 shows a situation of a delayed project with activities finishing ahead of and behind schedule, resulting in p-factor values lower than 1.

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 25 | 50 | 75 | 100 | | | | | | | | |
| | | | | | 30 | 60 | 90 | 120 | 150 | | | |
| | | | | | 100 | 200 | | | | | | |
| | | | | | | | | | | 25 | 50 | 75 |
| **EV** | 25 | 50 | 75 | 100 | 230 | 360 | 390 | 420 | 450 | 475 | 500 | 525 |
| **ES** | 1 | 2 | 3 | 4 | 5.3 | 6.6 | 6.9 | 7.4 | 8 | 9 | 10 | 11 |
| **p** | 1 | 1 | 1 | 1 | 0.85 | 0.81 | 0.86 | 0.93 | 1 | 1 | 1 | 1 |
| **Time** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |

**Fig. 10.21** Project progress with a 1 week delay

## Questions

1. The effective earned value EV(e) is equal to the sum of the risk-free and risky (i.e. with a probability of rework) earned value portions.
   (a) True
   (b) False
2. Fast tracking (i.e. starting an activity while at least one of its predecessors is not finished yet) leads to a deteriorating schedule adherence value.
   (a) True
   (b) False
3. Activities that are completed within their estimated time and budget can never cause a decrease in schedule adherence (p-factor).
   (a) True
   (b) False
4. When a project suffers from a delay, the p-factor will always be less than 1 somewhere during the project (i.e. at least for one period).
   (a) True
   (b) False
5. Which one of the following is no possible reason for imperfect schedule adherence?
   (a) Fast tracking.
   (b) Deviations between PV and EV accrue.
   (c) Deviations between PV and AC.
   (d) All previously mentioned reasons can cause imperfect schedule adherence.

# Chapter 11
# Schedule Control

## PC17: Bottom-Up Control[1]

The interest in the sensitivity of project activities from both academics and practitioners lies in the need to focus a project manager's attention on those activities that influence the (time and cost) performance of the project. When management has a certain indication of the potential impact of the various activities on the project objective, a better management focus and a more accurate response during project control should positively contribute to the overall performance of the project.

Schedule Risk Analysis (SRA) is a simple yet effective technique to connect the risk information of project activities to the baseline schedule, in order to provide sensitivity information of individual project activities to assess the potential impact of uncertainty on the final project duration and cost. A traditional schedule risk analysis requires four steps, as described in the article "*RA5: Schedule sensitivity*" on page 123, to report activity sensitivity measures that evaluate each activity's time and cost estimate on a scale of risk.

These sensitivity measures can be used by the project manager to distinguish between risky and less risky activities in order to focus on those activities that might have an impact on the overall project objective during the project progress. In this article, the following three topics will be briefly explained:

- Sensitivity measures: Risk classification for project activities.
- Action threshold(s): Interpretation of the measures to restrict a project manager's focus.
- Bottom-up project control: A project control approach.

[1]This section is also available as an article entitled "Bottom-up project control: Setting action thresholds using schedule risk analysis" published online at PM Knowledge Center.



M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_11

### *Sensitivity Measures*

The output of a schedule risk analysis is a set of measures that define a degree of activity criticality and sensitivity. The values of the activity sensitivity measures give an indication of the potential impact of changes in the baseline activity durations and costs on the overall project objective (project duration or total cost). The values for the sensitivity measures for each activity need to be interpreted by the project manager such that they can be used to distinguish between risky and less risky activities and to set action thresholds that act as a mechanism to take corrective actions in case the project in progress runs into trouble. Well known sensitivity measures used in a schedule risk analysis are the criticality index, the significance index, the schedule sensitivity index and the cruciality index, which are discussed in the article "*RA6: Time sensitivity*" on page 126. In the remainder of this article, a fictitious sensitivity measure is used that expresses the sensitivity of each activity on a scale between 0 % (insensitive) and 100 % (highly sensitive).

### *Setting Action Thresholds*

Action thresholds should be set by the project manager using the activity sensitivity information in order to put his/her focus on the most risky activities of the project. A predefined action threshold defines the intensity of control measured by the portion of project activities that require high attention. Consequently, knowledge of activity sensitivity information is crucial to steer a project manager's attention towards a subset of the project activities that have a high expected impact on the overall project performance and objectives (time and cost). These highly sensitive activities are subject to more intensive control, while other less sensitive activities require less or no attention during project progress. In the illustrative Fig. 11.1, the set of activities is split into three classes, based on the values of the sensitivity measures obtained from a schedule risk analysis (the horizontal bars), as follows:

- If "sensitivity measure $<$ 30 %" then "safe zone": These activities are considered to be insensitive to the project objective, and hence, they do not require special attention from the project manager once the project is in progress.
- If "30 % $\leq$ sensitivity measure $<$ 60 %" then "watch out zone": These activities are considered to have a low to moderate potential impact on the project objective. They do not require special actions, but a higher degree of attention during project progress could prevent future problems.
- If "sensitivity measure $\geq$ 60 %" then "dangerous zone": These project activities are highly sensitive and are assumed to have a high potential impact on the project objective. These activities must be under constant attention of the project manager during their progress and require immediate corrective actions in case these activities show deviations from their baseline values.

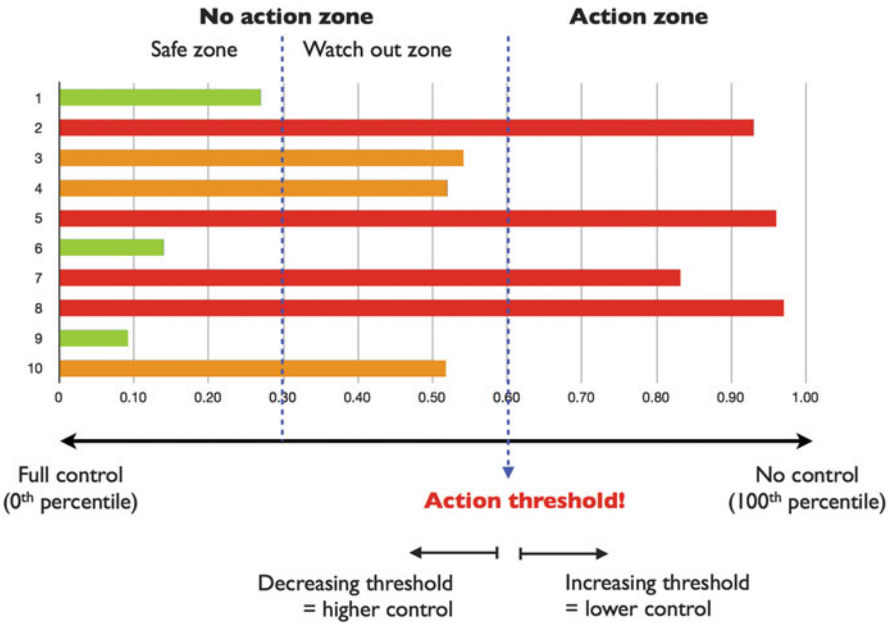



**Fig. 11.1** Action threshold setting to distinguish between risky and less risky activities

In the example Fig. 11.1, the action threshold has been set at a sensitivity value of 60 % such than only four activities (2, 5, 7 and 8) require intensive attention and control and a possible action. The other six activities are said to have a low or moderate sensitivity and require less or no attention during project progress. A further division is made within the "no action zone" to make a distinction between "completely safe" and "moderately safe" activities in order to illustrate that the use of multiple thresholds is possible.

## *Bottom-Up Project Control*

Figure 11.2 displays a work breakdown structure (WBS) of the example project, which splits the project objective into smaller pieces. It graphically illustrates the bottom-up control approach on the three sets of activities and the impact of actions on the overall project objective. Since the action threshold(s) define(s) the intensity of control at the lowest level of the WBS (activity level) in order to control and manage the overall project objective at the highest WBS level, this approach is referred to as bottom-up project control. Since the set of project activities is split into various classes where each class has its own intensity of control, this approach aims at reducing the project manager's effort such that he/she can focus on only the risky activities that might have an impact on the project objective.

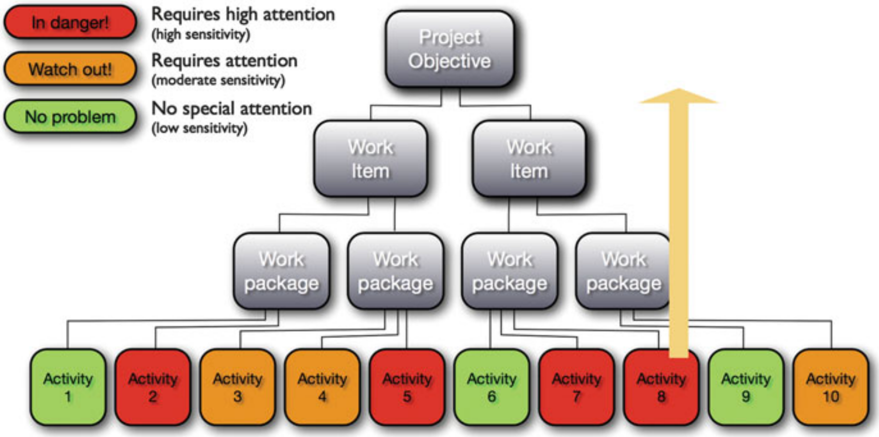



**Fig. 11.2** Bottom-up project control using schedule risk analysis

Research studies have shown that the bottom-up activity based control approach using sensitivity information of activities obtained through a standard schedule risk analysis is particularly useful when projects contain a lot of parallel activities. More information can be found in the article "*PC19: Why it works/fails*" on page 258.

## Questions

1. During project progress, it is quite likely that the activities on the critical path change.

   (a) True
   (b) False

2. Risk sensitivity measures are defined on the project level (i.e. highest level of the WBS).

   (a) True
   (b) False

3. An activity can be considered as highly sensitive if that activity has a high impact on the overall cost and/or time performance. These activities require more attention during project progress.

   (a) True
   (b) False

4. If the sensitivity value of an activity reaches the action threshold, immediate corrective action needs to be taken without hesitation.
   (a) True
   (b) False
5. Project control based on activity level information (i.e. bottom-up project control) is most effective when dealing with projects that contain a lot of parallel activities.
   (a) True
   (b) False
6. The primary aim of bottom-up project control is ...
   (a) to break up the project in manageable parts, such that several different project managers can each manage a part of the project in isolation.
   (b) to reduce the project manager's control effort by focusing only on the risky activities.
   (c) to enable the project manager to reduce the number of activities in the project network, and hence, to make the project much smaller.
   (d) None of the above is true.

## PC18: Top-Down Control[2]

Measuring and controlling the performance of a project in progress is key to the success or failure of a project. It requires a set of metrics to measure the current progress and to forecast the future expected project behavior such that the project manager is able to timely detect project problems and take corrective actions to bring the project back on track.

Earned Value Management (EVM) provides such performance measures at periodic time periods to compare the initial baseline schedule with the current project performance. It is based on three key metrics and simple formulas and provides the project manager with a set of measures that act as an early warning signal to detect project problems in an early stage. More information on these EVM performance measures can be found in the article "*PC2: EVM Overview*" on page 199. In this article, the following three topics will be briefly explained:

- Earned value management: Controlling the performance of projects in progress.
- Action threshold(s): Interpretation of the performance measures as early warnings for action.
- Top-down project control: A project control approach.

[2]This section is also available as an article entitled "Top-down project control: Setting action thresholds using earned value management" published online at PM Knowledge Center.

## *Earned Value Management*

Earned value management is a project control technique to measure the performance of projects in progress using three key metrics. The key metrics are used to measure the (time and cost) performance at the current moment in time where the project manager is right in the middle of the project progress and still has room to take corrective actions in case the project is in trouble. The current project performance can be evaluated by the following performance measures: Two versions of the schedule performance index (SPI or SPI(t)) to measure the current time performance and the cost performance index (CPI) to measure the current cost performance. In the remainder of this article, the performance of a project in progress over time will be measured on fictitious project data with a duration of 11 weeks using a performance measure that reports the value 1 if the project is on schedule (SPI or SPI(t)) or on cost (CPI). More information on these EVM performance measures can be found in the article "*PC8: Project performance*" on page 217.

## *Setting Action Thresholds*

The EVM performance measures that are periodically reported to the project manager must act as early warning signals to timely detect project problems and to trigger actions in case the project performance runs out of control. Consequently, they provide the project manager with useful information of the overall project performance at a certain moment in time, but they do not provide detailed information about the performance of the individual project activities. Figure 11.3 illustrates the project performance over time (weeks 1–11) and the action threshold set by the project manager. This action threshold defines the intensity of control and is shown by the increasing dotted line on the figure. When the project performance measure drops below the action threshold at a certain moment, it is assumed that the current performance is too far away from the initial expectations, which might put the overall project success in danger. Consequently, the project performance behavior at weeks 7 and 8 must act as a trigger to take corrective actions in order to bring the project performance back on track.

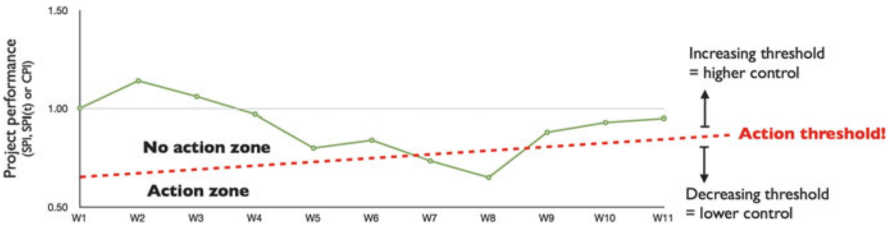



**Fig. 11.3** Action threshold setting to distinguish between risky and less risky activities

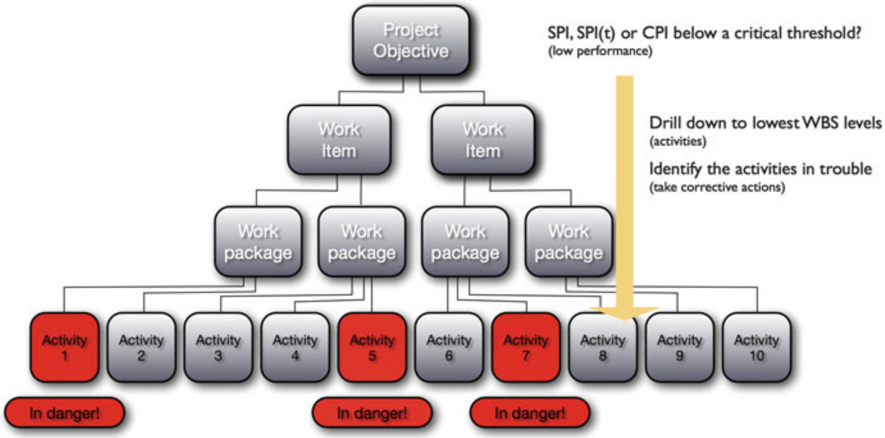



**Fig. 11.4** Top-down project control using earned value management

## *Top-Down Project Control*

Figure 11.4 displays a work breakdown structure (WBS) of the example project, which splits the project objective into smaller pieces. It graphically illustrates the top-down control approach where project based performance measures give an indication of the current performance that act as a trigger to drill down to the lowest WBS levels to find out where the problems really are. In the example figure, activities 1, 5 and 7 are assumed to be the activities in progress with problems that require corrective action by the project manager to bring the overall project performance back on track.

Research studies have shown that the top-down project based control approach using schedule and cost performance metrics obtained by using earned value management is particularly useful when projects contain a lot of serial activities. More information can be found in the article "*PC19: Why it works/fails*" on page 258.

### Questions

1. The EVM performance measures provide information about the performance of the individual project activities at a certain moment in time.
   (a) True
   (b) False

2. The action thresholds set during a top-down project control approach have to remain constant throughout the entire project.
   (a) True
   (b) False
3. Figure 11.3 on page 256 shows the periodic values for EVM performance measures and a dotted line as action threshold for top-down project control. The zone for taking corrective actions is situated . . .
   (a) below the dotted line.
   (b) above the dotted line.
4. Assume that the performance measure in Fig. 11.3 is equal to the SPI(t). If we increase the action threshold represented by the dotted line, it will result in an increase in the intensity of top-down project control.
   (a) True
   (b) False
5. A top-down control approach does not provide an indication of which specific activities are causing the problems in the project.
   (a) True
   (b) False
6. Does a top-down project control approach require less/same/more control effort than a bottom-up control approach?
   (a) Less effort
   (b) Same effort
   (c) More effort
   (d) This depends on the nature of the project and the way its progress evolves over time.

## PC19: Why It Works/Fails[3]

Controlling the performance of a project in progress is key to the success or failure of a project. It requires a set of metrics to measure the current progress and to forecast the future expected project behavior such that the project manager is able to timely detect project problems and take corrective actions to bring the project back on track. In order to be able to take high quality corrective actions, the project

[3]This section is also available as an article entitled "Why project control works so well for some projects and fails so miserably for others" published online at PM Knowledge Center.

manager should set action thresholds that act as triggers to take these actions, which can be done in two ways:

- A bottom-up control approach using schedule risk analysis
- A top-down control approach using earned value management

In this article, the efficiency of both approaches is analyzed and results are reported.

## *Project Control*

In a bottom-up control approach, the lowest levels of the work breakdown structure (WBS) serve as static predictors of the impact changes might have on the project objective. More precisely, sensitivity information for each activity obtained from a schedule risk analysis (SRA) enables the project manager to distinguish between sensitive and insensitive activities. Action thresholds can then be set at a certain sensitivity value such that the activities with a higher sensitivity value than the action threshold value are said to be risky and are assumed to have a huge potential impact on the project objective. This limited set of activities requires an intensive control during project progress and might be subject to corrective actions in case these activities might endanger the project. This bottom-up control approach is explained in detail in the article "*PC17: Bottom-up control*" on page 251.

A top-down project control approach requires a dynamic knowledge of the overall project performance at a current moment in time, which is provided by earned value management (EVM) indicators such as the schedule and cost performance indices. These indicators serve as early warning signals and must trigger corrective actions when they drop below a predefined acceptable threshold. In this case, the project manager must drill down to the lower levels of the work breakdown structure to find the set of activities that are in trouble and that need to be put back on the right track. This drill-down project control is described in the article "*PC18: Top-down control*" on page 255.

## *Control Efficiency*

The action thresholds in both control approaches define the points of focus and attention of the project during project progress and serve as triggers to take corrective actions to bring project in trouble back on the right path. Consequently, in order to efficiently control projects in progress, the project manager must have access to reliable data provided by SRA or EVM metrics. This should enable the manager to focus on the activities in trouble in order to take high quality corrective actions that bring the project performance back to normal. Therefore, setting action thresholds involves a trade-off between the intensity of control at the lowest WBS levels and the potential impact actions might have on the overall project objective (highest WBS level).

- Intensity of control: The value of the action thresholds defines the intensity of control and consequently the effort a project manager has to make during project progress. A low control intensity runs the risk of masking problems in certain activities in progress, which will only be detected much later in the project life. A higher control intensity reduces the risk that problems in activities will not be detected but results in a higher effort for the project manager.
  - Bottom-up control: Low (high) thresholds result in a bigger (smaller) set of activities that are assumed to be sensitive, and hence, result in a higher (lower) intensity of control.
  - Top-down control: Low (high) thresholds mean that the probability to drill down decreases (increases), and hence, this results in a lower (higher) intensity of control.
- Impact of actions: When actions are taken in order to bring activities in progress back to their expected behavior, the project manager aims at bringing the project performance back on track as stipulated in the baseline schedule.
  - Bottom-up control: Corrective actions taken on very sensitive activities probably might have a bigger beneficial impact on the project objective than actions taken on activities that are assumed to have a lower impact. Selecting the right activities during a schedule risk analysis is therefore crucial in a bottom-up control approach.
  - Top-down control: Selecting the activities at the right moment is crucial to detect the activities in progress that really cause the project problems. Therefore, reliable performance data during project progress to drill down to the project activities at the crucial moment in the life of the project are essential for the quality of the corrective action decision making process.

The efficiency of both control approaches is measured as a combination of these two factors. Figure 11.5 shows the control efficiency tested on a wide and diverse set of projects along the following dimensions.

- Project network structure (x-axis): More than 4,000 project networks have been used to test the control efficiency. These networks are fundamentally different from each other as measured by the topological structure. In Fig. 11.5, the projects range from projects with a lot of parallel activities to projects with a more serial structure.
- Control efficiency (y-axis): The efficiency of control is measured by a combination of the intensity of control and the impact of actions. A low level of control intensity and a high positive impact of the actions on the project objective result in a high control efficiency. Figure 11.5 compares the efficiency of two top-down control approaches, one bottom-up approach and one combined control approach.

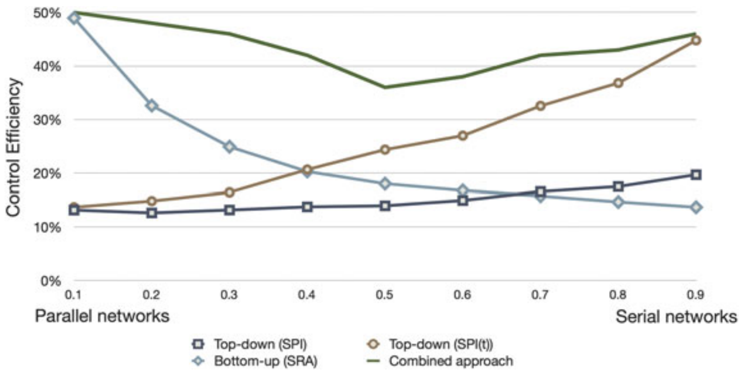



**Fig. 11.5** The control efficiency

The results of Fig. 11.5 can be summarized as follows:

- The top-down approach using the schedule performance index SPI shows a relatively low efficiency. This can be mainly contributed to the low reliability of the SPI at the final stages of the project (see the article "*PC9: Time performance*" on page 220).
- The top-down approach using the new schedule performance index SPI(t) is much more efficient, and this efficiency increases as the project becomes more and more serial.
- The bottom-up approach using schedule risk analysis (SRA) has a high efficiency when projects contain a lot of activities in parallel.
- The combined approach where both activity sensitivity information as well as project based performance indicators are used performs best and leads to a significant increase in the control efficiency.

## Questions

1. Schedule risk analysis (SRA) is used in a bottom-up control approach while earned value management (EVM) is used as a top-down control approach.

   (a) True
   (b) False

2. When a project is monitored using a top-down control approach, the project manager will never have to drill down to the activity level details.

   (a) True
   (b) False

3. When the action threshold values used in a bottom-up control approach are lowered, the intensity of control . . .
   (a) increases.
   (b) decreases.
   (c) remains the same.
4. When the action threshold values used in a top-down control approach are lowered, the intensity of control . . .
   (a) increases.
   (b) decreases.
   (c) remains the same.
5. In a bottom-up control approach, corrective actions should be taken on the most sensitive activities.
   (a) True
   (b) False
6. In order to control a project with a very serial network, it is better to use SRA than EVM.
   (a) True
   (b) False
7. It is always more efficient to use a control approach that combines SRA and EVM than to apply the approaches separately.
   (a) True
   (b) False

## PC20: Retained and Overridden Logic[4]

One of the primary tasks during project control is periodically updating the baseline schedule to reflect the actual progress of the work done and to present a realistic forecast of the remaining work. A project in progress can be monitored using Earned Value Management (EVM). In this technique, the schedule predictions using the performance metrics available in EVM completely rely on estimates about the percentage completion of the activities and give a helicopter view on the performance of the project at the current status day (see article "*PC10: Forecasting*" on page 225).

[4]This section is also available as an article entitled "Schedule prediction: From a retained to an overridden logic" published online at PM Knowledge Center.

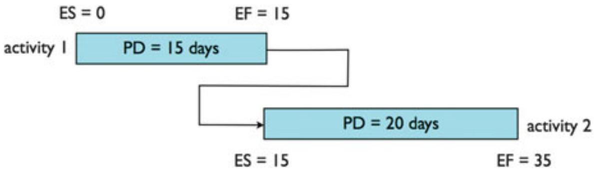



**Fig. 11.6** An example earliest start baseline schedule

A more time-consuming and detailed alternative is to update the baseline Gantt chart at the activity level. In this so-called tracking Gantt chart, the schedule prediction is made by taking the actual and estimated remaining durations of each activity into account and by updating the Gantt chart. In doing so, the tracking Gantt chart gives a prediction of the future schedule based on the inputs of actual and remaining durations/costs of activities. In this article, periodic updates of a project in progress using the tracking Gantt chart in order to obtain predictions of timing of the remaining work is illustrated using three options, as follows:

- Overridden logic: When work is performed out-of-sequence, some of the precedence relations of the remaining work are no longer respected.
- Retained logic: All remaining work is assumed to follow the logic of the original baseline precedence relations.
- Percentage overridden/retained logic: When work is performed out-of-sequence, the logic of the precedence relations is only partially violated as an option in-between the overridden and retained logic.

Figure 11.6 displays an illustrative baseline schedule Gantt chart with two activities. Activity 1 has an estimated duration of 15 days while its successor activity 2 has an estimated duration of 20 days. The precedence relation between the two activities is assumed to be a finish-start relation with a minimal time-lag equal to zero. The earliest start (ES) and earliest finish (EF) times of each activity are indicated, resulting in a total project planned duration of 35 days. For more information on the use of precedence relations, see the article "*BS3: Precedence relations*" on page 16. Detailed information on the earliest and latest start calculations can be found in the article "*BS10: Activity slack*" on page 43.

## *Retained or Overridden Logic*

Since work is often performed out-of-sequence, the original logic captured by the precedence relations between activities as specified in the baseline schedule is often violated. This situation can cause unrealistic deviations between the baseline scheduling logic and the project tracking Gantt chart, and often leads to unnecessary adaptations and modifications to the baseline schedule.

Generally, there are two options available in software tools to handle out-of-sequence progress during the tracking phase, as follows:

- Retained logic assumes that the original precedence relations are still valid, even when activity overlaps during progress have taken place. This logic respects all precedence relations of the remaining work, but often leads to unrealistically long project duration forecasts.
- Overridden logic assumes that an activity that started with a certain overlap will violate the original precedence relation logic completely. This logic assumes that the remaining work of an activity in progress can be done without being affected by its incomplete predecessor activities, but it often leads to unrealistically short project duration forecasts. This logic is also known as out-of-sequence progress.

These two extreme logics are illustrated in Fig. 11.7 where it is assumed that the current status day is equal to day 8. Activity 1 has been in progress for 8 days, and the project manager estimates that it will take another 12 days to finish this activity (i.e. the remaining duration (RD) is equal to 12 days). Even though activity 1 has not finished yet, activity 2 has already started and has been in progress for 2 days. Consequently, the original precedence relation between both activities has been violated. The remaining duration is estimated at 18 days. More information on setting actual and remaining durations can be found in the article "*PC21: Updating schedules*" on page 267.

Since the relations between the two activities have not been respected, the overridden logic assumes that this will not be the case in the future, and hence, the remaining work of activity 2 can start immediately at the status day, leading to an earliest finish of 26 days. The retained logic keeps respecting the precedence relations incorporated by the baseline scheduling logic (Fig. 11.6) and assumes that the remaining work of activity 2 can only start after the expected finish of activity 1. This logic leads to an expected project duration equal to 38 days.

Figure 11.7 shows that the expected delay of activity 1 will be recovered by the out-of-sequence work of activity 2, leading to a shorter project duration than stipulated by the baseline schedule ($26 < 35$). In the retained logic, the delay in activity 1 will cause a delay in the expected project duration ($38 > 35$), despite the out-of-sequence work of activity 2.

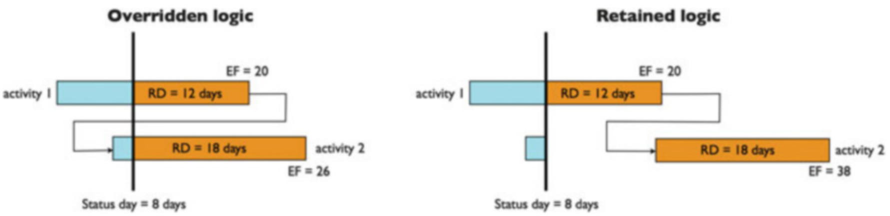



**Fig. 11.7** Overridden (*left*) and retained (*right*) logic for the remaining work at status day = 8

## *Percentage Overridden/Retained Logic*

Since both the overridden and the retained logic display a schedule forecast taking both the actual activity durations (the work done) and the remaining activity durations (the work yet to be done) into account, a third option is available that constructs a schedule forecast for the remaining work in between the overridden and retained logic.

Motivated by the observation that the tracking Gantt chart is nothing more than a schedule forecast, the option to shift between the two extreme logics should allow the user to adjust the forecast according to his/her own wishes.

The percentage overridden/retained logic displays a tracking Gantt chart as a percentage in between the overridden (0 %) and retained (100 %) logic, as follows:

- 0 %: In the overridden logic, the remaining work of activity 2 can be started immediately (i.e. 0 days after the current status day).
- 100 %: In the retained logic, the remaining work of activity 2 can be started when its predecessor is finished, i.e. $20 - 8 = 12$ days after the current status day.
- x %: The remaining work of activity 2 can be started in-between the start defined by the overridden logic and the start defined by the retained logic. More precisely, x % of the retained logic start minus the current status day (x % * 12 days) is added to the status day to define the start of the remaining work.

The percentage between the overridden and retained logic obviously affects the total project duration estimate. In Fig. 11.8, the percentage overridden/retained logic is set at 33 % (left) and 66 % (right), leading to an expected project duration equal to 30 and 34, respectively. As an example, a percentage of 33 % leads to an intermediate tracking Gantt chart where activity 2 will start $0.33 * 12$ days $= 4$ days after the current status day.

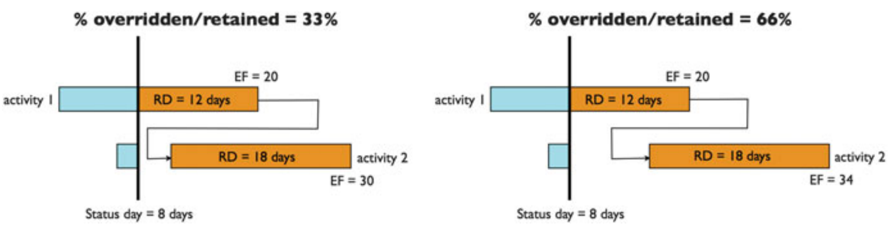



**Fig. 11.8** Percentage retained/overridden logic equal to 33 % (*left*) or 66 % (*right*) at status day = 8

## Questions

1. When applying an overridden logic, all precedence relations of the remaining work are violated.
   (a) True
   (b) False
2. For a completely parallel project (i.e. all activities can be performed in parallel according to the baseline schedule), the choice between overridden and retained logic does not matter.
   (a) True
   (b) False
3. A retained logic will never yield a shorter project duration forecast than an overridden logic.
   (a) True
   (b) False
4. Activity preemption (i.e. the splitting of activities) occurs in an overridden logic, but not in a retained logic.
   (a) True
   (b) False
5. The higher the percentage overridden/retained logic is set, the longer the forecasted project duration will be.
   (a) True
   (b) False
6. Consider Fig. 11.7 on page 264 showing both an overridden and a retained logic. If the percentage overridden/retained logic would be set to 75 %, the project duration forecast would be . . .
   (a) 29.
   (b) 35.
   (c) 36.
   (d) 42.

## PC21: Updating Schedules[5]

One of the primary tasks during project control is periodically updating the baseline schedule to reflect the actual progress of the work done and to present a realistic forecast of the remaining work. Measuring the progress of a project can be done in various ways. The two most well-known and widely used techniques are:

- Earned Value Management (EVM): In this technique, the schedule predictions using the performance metrics available in EVM completely rely on estimates about the percentage completion of the activities and give a helicopter view on the performance of the project at the current status day. This technique is outside the scope of this article. Further references can be found in the article "*PC2: EVM Overview*" on page 199.
- Critical Path Method (CPM): A more time-consuming and detailed alternative is to update the baseline Gantt chart at the activity level, which is often constructed using CPM techniques (or its extensions). In this so-called tracking Gantt chart, the schedule prediction is made by taking the actual and estimated remaining durations of each activity into account and by updating the Gantt chart. In doing so, the tracking Gantt chart gives a prediction of the future schedule based on the inputs of actual and remaining durations/costs of activities.

In this article, periodic updates of a project in progress using the tracking CPM Gantt chart are discussed. More precisely, the necessary inputs to periodically update the baseline schedule are described and illustrated by means of a simple two-activity project. In this article, the following three topics are discussed:

- Baseline schedule: Construct the baseline schedule to get the planned duration and budget at completion.
- Schedule update: Set the actual start, actual and remaining duration and actual and remaining cost values to update the schedule.
- Two remarks: Explain the difference between EVM and CPM schedule control and see whether the inclusion of resource costs will change the general approach discussed in this article.

### *Baseline Schedule*

In Fig. 11.6 on page 263, a two activity project baseline schedule is shown with a total duration of 35 days. The precedence relation between the two activities is assumed to be a finish-start relation with a minimal time-lag equal to zero. The earliest start (ES) and earliest finish (EF) times of each activity are indicated. The

[5]This section is also available as an article entitled "Updating the schedule: Setting actual and remaining cost values of activities in progress" published online at PM Knowledge Center.

project example is borrowed from the article "*PC20: Retained and overridden logic*" on page 262. It is assumed that the activities do not need renewable or consumable resources and that the activity durations and costs are set to the following values:

- Activity 1:
  - Planned duration = 15 days
  - Fixed cost = € 1,000
  - Variable cost = € 50 per day
- Activity 2:
  - Planned duration = 20 days
  - Fixed cost = € 2,000
  - Variable cost = € 50 per day

Given the baseline information in Fig. 11.6, the total cost of the activities and the total project cost can be calculated following the guidelines in the article "*BS9: Activity costs*" on page 39, as follows:

- Activity 1: € 1,000 + 15 * € 50 = € 1,750
- Activity 2: € 2,000 + 20 * € 50 = € 3,000

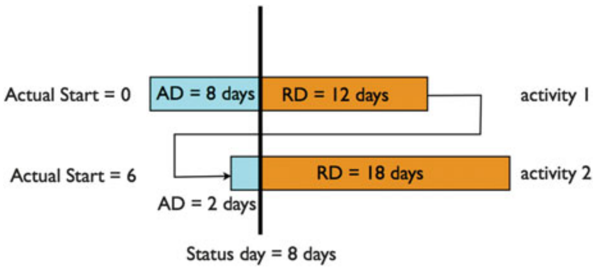
resulting in a total budget at completion (BAC) equal to € 4,750.

## *Project in Progress*

A project in progress needs a periodic update of the baseline schedule to be able to see the impact of previous changes on the total expected project duration and cost. Therefore, progress input data must be given at regular times to update the portion of work already done and to predict the future changes and the impact on the total project duration and cost.

Figure 11.9 displays the project tracking Gantt chart at status day 8. Even though activity 1 has not finished yet, activity 2 has already started despite the precedence relation between the two activities. This is known as an overridden schedule logic as explained in the article "*PC20: Retained and overridden logic*" on page 262

**Fig. 11.9** Project in progress at status day = 8 (overridden logic)



## *Actual Start*

Updating the schedule during project progress results in activities that have already started (finished or in progress) and activities that still have to start. Therefore, a first crucial input metric to update the schedule is the actual start.

- Actual Start: Actual start time of an activity on or before the status date that might differ from the original baseline start time.

The actual start times are set to 0 (activity 1, i.e. the project start) and day 6 for activity 2 to denote that both activities are in progress at day 8.

## *Actual and Remaining Duration*

Activities in progress need an update about their actual and remaining durations that might differ from their baseline duration values. Therefore, two crucial input metrics need to be given to update the schedule, as follows:

- Actual Duration (AD): The time spent on an activity between its actual start time and the current status date. It represents the workload already spent on this activity until now (i.e. day 8).
- Remaining Duration (RD): A forecast of how many time units (hours, days, weeks, years) an activity will need from the status date forward to finish its remaining work.

Activity 1 has been in progress for 8 days (actual duration), and the project manager estimates that it will take another 12 days to finish this activity (remaining duration). Likewise, the actual duration for activity 2 is equal to 2 days while the remaining duration is estimated at 18 days.

## *Actual and Remaining Cost*

Calculating the actual cost of the portion of work that is finished at the current time (day 8) and the remaining cost of the work that still has to be done is based on the estimates for the actual and remaining durations.

Estimating the actual cost of the portion of work finished for the activity can be done using the following three metrics:

- Actual Cost (AC): The actual cost for the work already done. Ideally, this actual cost should have a clear relation with the workload already done by the various resources assigned to this activity. It can be done based on time sheets for the various resources or based on knowledge about the actual spendings already made for the portion of work done for the activity. Since updating the resource time sheets to get the actual cost of an activity in progress can be a time-consuming task, a simple and effective alternative way to estimate the actual cost of an activity in progress can be done using the following formula:

| |
|---|
| Actual Cost<br>=<br>Planned Actual Cost + Planned Actual Cost Deviation |

- Planned Actual Cost (PAC): The planned actual cost is calculated based on the baseline schedule information and is equal to the total planned activity cost given by the baseline schedule when the activity is in progress for AD time units. Consequently, the actual duration estimate to estimate the total workload spent on an activity in progress determines the planned actual cost for this activity. It is assumed that all costs are as planned and also the resources work as planned. For more information about calculating the total activity cost, see the article "*BS9: Activity costs*" on page 39.
- Planned Actual Cost Deviation (PACDev): Since uncertainty typifies projects or resources shift from time to time and work more or less efficient than originally planned, the actual cost of an activity in progress can differ from the planned actual cost. Consequently, the planned actual cost deviation is a way to express that difference, and hence, positive or negative values can be set to denote that the actual cost of an activity in progress is higher or lower than the planned actual cost. This PACDev replaces the various time sheets of resources.

In the example of Fig. 11.9, the planned actual cost is equal to € 1,000 + 8 * € 50 = € 1,400 for activity 1 and € 2,000 + 2 * € 50 = € 2,100 for activity 2. Consequently, the total actual cost of the portion of work done at day 8 is equal to € 3,500. In case the real actual cost is lower or higher than this estimated planned cost, the PACDev should be set to negative or positive cost values to update the actual cost.

Estimating the remaining cost of the remaining portion of work for the activity follows a similar approach, using the following three metrics:

- Remaining Cost (RC): A forecast of how many additional monetary resources (euro, dollars, …) an activity requires to finish the portion of remaining work. Similar to the actual cost estimate, the remaining cost estimate can be determined in the following way:

| |
|---|
| Remaining Cost<br>=<br>Planned Remaining Cost + Planned Remaining Cost Deviation |

- Planned Remaining Cost (PRC): The planned remaining cost is, similar to the planned actual cost, calculated automatically based on the remaining duration (RD) input.

- Planned Remaining Cost Deviation (PRCDev): In order to give a realistic estimate of the remaining cost, the planned remaining cost value can be modified using the optional planned remaining cost deviation field.

In the example of Fig. 11.9, the planned remaining cost is equal to 12 * € 50 = € 600 for activity 1 and 18 * € 50 = € 900 for activity 2. Consequently, the total remaining cost of the remaining portion of work to be done from day 8 is equal to € 1,500. In case the remaining cost is expected to be lower or higher than this estimated planned cost, the PRCDev should be set to negative or positive cost values to update the remaining cost.

It should be noted that the total expected cost at completion is now equal to € 3,500 + € 1,500 = € 5,000 (when both PACDev and PRCDev are equal to zero), which exceeds the BAC of € 4,750 and therefore a cost overrun is expected.

## *Remarks*

It should be noted that the EVM approach also has the objective of efficient project control, similar to the CPM schedule update method discussed in this article, but is however based on a Percentage Completed (PC) estimate. The PC estimate reflects the portion of the total workload that is estimated to be finished. The Earned Value metric EV is based on this estimate as EV = PC * BAC. Note that neither the actual/remaining duration nor the actual/remaining cost need to have a link with the PC estimate and will not be explicitly used for the calculation of the EV metric. More information can be found in the article "*PC5: Key metrics*" on page 208.

It is also worth noting that in the example of Fig. 11.9, no costs for renewable and consumable resources have been taken into account. If resource costs would have been included, the schedule update principles would remain the same. The only difference would lie in the calculations of the PAC and PRC values, as discussed in the article "*BS9: Activity costs*" on page 39.

### Questions

1. The critical path method (CPM) is a more time-consuming technique for making schedule predictions than earned value management (EVM).
   (a) True
   (b) False
2. The fixed cost of an activity is distributed evenly over the duration of the activity.
   (a) True
   (b) False

3. If the planned actual cost deviation (PACDev) of an activity is negative, the actual cost (AC) of that activity is lower than its planned actual cost (PAC).
   (a) True
   (b) False

4. Consider Fig. 11.9 on page 268 with two activities in progress at day 8. Assume that activity 1 has a fixed cost of € 2,000 and a daily variable cost of € 75 and that activity 2 has a fixed cost of € 1,500 and a daily variable cost of € 100. What is the planned remaining cost (PRC) at the status date?
   (a) € 2,700
   (b) € 3,500
   (c) € 5,250
   (d) € 6,200

5. If one would like to include resource costs, these could simply be added to the fixed and variable activity costs in the calculation of PAC and PRC without having to change the schedule update principles.
   (a) True
   (b) False

# Part IV
# Solutions

# Chapter 12
# Solutions

This brief chapter gives an overview of the solutions to all questions of this book. The solutions are available as summary tables split up in three sections representing the three main topics of this book. The solutions are also available as a PDF file and an MS Excel file and can be downloaded from www.or-as.be/pmkc. For some of the questions, the answers are subject to interpretation and discussion, and therefore, answers might be adapted based on input from the readers. This is exactly the reason why the answer have also been put online rather than only as a hard copy in the next sections of this chapter. It allows for modifications and extra explanation, if necessary. Therefore, the MS Excel file will contain a label mentioning the date of the last update such that the reader can check at any time whether he/she has the latest version. Some of the answers will contain some extra comments to highlight how the calculations have been carried out or to mention possible misinterpretations. Any comments of readers is welcome by an email to the author of the book, and the remark will possibly be taken into account in the next update of the solutions files.

The "Baseline Scheduling" section provides answers for 137 questions, with extra comments for 50 of the questions. The "Schedule Risk Analysis" section provides answers for 124 questions, with extra comments for 79 of the questions. The "Project Control" section provides answers for 119 questions, with extra comments for 72 of the questions.

## Baseline Scheduling

The solutions for the *baseline scheduling* questions are displayed in Table 12.1. For the answers with an asterisk (*), a more detailed explanation is given in this section with reference to the article abbreviation and question number.


M. Vanhoucke, *Integrated Project Management Sourcebook*,
DOI 10.1007/978-3-319-27373-0_12

**Table 12.1** Solutions for the baseline scheduling questions

| | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 |
|---|---|---|---|---|---|---|---|
| BS1 | – | – | – | – | – | – | – |
| BS2 | b | a | a | c | b | b | – |
| BS3 | b | b* | a | d | b* | b | b* |
| BS4 | b | a | b* | d | d | a, d | – |
| BS5 | b | a | a | b | b | b | – |
| BS6 | a | a | b | b | d | – | – |
| BS7 | b | a | a* | d* | a* | – | – |
| BS8 | a | a | a | – | – | – | – |
| BS9 | b | a | a | c* | c | – | – |
| BS10 | a | b* | a | a | – | – | – |
| BS11 | b | b* | a | a | a* | c* | – |
| BS12 | a | b | c* | c* | c* | c* | e* |
| BS13 | a | b | a* | d* | – | – | – |
| BS14 | b | b* | a | a | – | – | – |
| BS15 | b | c | a* | a* | c* | d* | – |
| BS16 | a | c | b | b* | – | – | – |
| BS17 | b | a | a | b | b | – | – |
| BS18 | b* | b | a* | a | c* | e* | – |
| BS19 | a | b | b* | a | c* | a* | – |
| BS20 | a | b | a | b* | a | b | f* |
| BS21 | a* | b* | a | b | b* | b | – |
| BS22 | a | b* | a | b | b, d | – | – |
| BS23 | a | b | c | b* | c* | – | – |
| BS24 | b | a | b | a | b* | a* | – |
| BS25 | b* | b* | a* | c* | d* | – | – |
| BS26 | c | d | b | a | – | – | – |
| BS27 | a | b* | a* | b* | – | – | – |

The list of comments is given along the following lines:

BS3–Q2: The statement should be true for a Finish-Start (FS) but not for Start-Finish (SF).
BS3–Q5: The start-start relationship enforces that the start of both activities can be separated by no more than 3 time periods.
BS3–Q7: As long as activity A finishes before activity B, all precedence relations are satisfied, and hence, activity A can have multiple start times.
BS4–Q3: The statement is not always true, and depends on the durations of the activities.
BS7–Q3: The critical path consists of activities 4, 7 and 8.
BS7–Q4: The critical path consists of activities 1, 2, 5, 8 and 10.
BS7–Q5: Paths B and C include activity 9, which is not on the CC since it can be shifted backwards. Path D does not include activity 8 which cannot be moved without increasing the duration of the complete project.

BS9–Q4: Answer C is the correct solution if it is assumed that a full-time worker is not paid by the project budget when he/she is not active. Otherwise, answer D is correct. See MS Excel solution file tab “BS9”.
BS10–Q2: The earliest start time of an activity is the maximum of the earliest finishing time of all its predecessors.
BS11–Q2: Decreasing the duration of an activity inflates the associated cost (more manpower, more expensive equipment, etc.)
BS11–Q5: The critical path with normal data is equal to A-C-E-F with length 13 and the critical path with crash data is equal to A-B-F or A-C-E-F with length 7. See MS Excel solution file, tab “BS11”.
BS11–Q6: The minimum cost (CP = 13) is equal to 62,000 which is simply the sum of all costs for normal data. The minimum cost (CP = 7) is equal to 80,000 and NOT 81,000. 81,000 is simply the sum of all costs for crash data but when you look carefully, activity D can be put at duration = 3 instead of its crash duration of 2, saving 1,000 euro. See MS Excel solution file, tab “BS11”.
BS12–Q3: The critical path length is equal to 17 time units. The earliest start of activity 5 equals 7 and the latest start equals 11, which results in a slack value of 4. See MS Excel solution file tab “BS12”.
BS12–Q4: All the predecessors of activity 5 are on the critical path, hence the safety slack is equal to the total slack.
BS12–Q5: See MS Excel solution file tab “BS12”.
BS12–Q6: The introduction of a buffer has no impact on the free slack of activity 2. See MS Excel solution file tab “BS12”.
BS12–Q7: See MS Excel solution file tab “BS12”.
BS13–Q3: The activity start times are equal to 0, 2, 5, 6, 8, 7 for activities 1 to 6, and the activity finish times are equal to 3, 4, 9, 9, 9, 9. See MS Excel solution file tab “BS13”.
BS13–Q4: See MS Excel solution file tab “BS13”.
BS14–Q2: The aim of the game is to cope with uncertainty in a reactive way.
BS15–Q3: The critical path contains the activities 1, 2, 4 and 6.
BS15–Q4: Calculate as the square root of the sum of the variances of the activities on the CP $(1+0.25+4+0.25)$. See MS Excel solution file tab “BS15”.
BS15–Q5: Use the normal distribution calculations $(21-19)/2.35$ to calculate the z-value and use MS Excel or normal tables to calculate the probability. See MS Excel solution file tab “BS15”. (using MS Excel NORMDIST function).
BS15–Q6: Use the average and standard deviation values of previous questions and find the 90th percentile. See MS Excel solution file tab “BS15”. (using MS Excel NORMINV function)
BS16–Q4: For independent paths, answer B is the correct one. For dependent paths it is not correct.
BS18–Q1: The word “total” assumes that both direct (immediate) and indirect successors are taken into account, so the statement is false.
BS18–Q3: Activity 5 is unrelated to activity 6.
BS18–Q5: The earliest finish times are equal to 1, 2, 4, 5, 7 and 8 for activities 1 to 6. See MS Excel solution file tab “BS18”.

BS18–Q6: The cumulative resource work content is equal to 6, 13, 15, 12 9 and 3 for activities 1–6 so the list is normally equal to [3, 2, 4, 5, 1, 6]. However, some activities come earlier in the list than predecessor activities (e.g. 3 comes before 1) which will never be possible. Taking both the precedence relations and the cumulate resource work content into account will transform the list to list [2, 4, 1, 3, 5, 6]. See MS Excel solution file tab “BS18”.

BS19–Q3: This is an advanced questions, and the answer is illustrated in the MS Excel solution file, tab “BS19”.

BS19–Q5: The schedule has finish times equal to 1, 2, 4, 10, 7 and 13 for activities 1 to 6, respectively. See MS Excel solution file, tab “BS19”.

BS19–Q6: Since activity 4 can now be scheduled in parallel with activity 3, a reduction of at least 2 time periods will be obtained. Due to the reduction, activities 5 and 6 can solve be scheduled in parallel, resulting in a total reduction of 5 time units. See MS Excel solution file tab, “BS19”.

BS20–Q4: The actual duration will be larger than or equal to the lower bound.

BS20–Q7: The CPLB = 17, the RBLB = 19 and the CSLB = 21. See MS Excel file, tab “BS20” or the book “Project Management with Dynamic Scheduling: Baseline Scheduling, Risk Analysis and Project Control” from which this example was taken.

BS21–Q1: An upper bound for a maximization problem is identical to a lower bound for a minimization problem, and ignores some constraints (e.g. the resource constraints). It does not provide a feasible solution.

BS21–Q2: A lower bound for a minimization problem ignores some constraints, but does not lead to a feasible solution. A heuristic search would provide an upper bound, and not a lower bound.

BS21–Q5: A heuristic solution provides a lower bound for a maximization problem.

BS22–Q2: This statement is false. While it is true for the minimization of the total project duration, it can be false for other scheduling objectives such as the net present value maximization.

BS23–Q4: This statement is false. While it is true for the minimization of the total project duration, it can be false for other scheduling objectives such as the net present value maximization.

BS23–Q5: One need more information on the schedule to determine which schedule is the best. The question does not follow the definition of regular measure of performance.

BS24–Q5: Activities that are scheduled in parallel might still have a total resource use that is higher than the availability.

BS24–Q6: The critical path is equal to activities 1, 2, 5, 8 and 10 with a duration of 13. Activities 2 and 4 cannot be scheduled in parallel and one of these activities must be delayed, leading to an increase in the total project duration of 2 time units. See MS Excel solution file tab “BS24”. Note that there is also an alternative critical path equal to 1, 4, 7, 9, 10.

BS25–Q1: The net present value should be higher than zero, not lower.

BS25–Q2: These activities should be scheduled as late as possible.

BS25–Q3: A lump sum payment is a huge positive cash flow, which should be scheduled as soon as possible.
BS25–Q4: See MS Excel solution file tab "BS25".
BS25–Q5: The new net present values are equal to € 59.15, € 58.49, € 62.18 and € 85.05 for options (a)–(d), respectively. See MS Excel solution file tab "BS25".
BS27–Q2: While many project scheduling software tools often focus on the efficient use of resources (i.e. levelling), they mostly take the minimization of the total project duration as the main objective.
BS27–Q3: This statement is true since the network is completely serial and hence both schedules will be identical.
BS27–Q4: An earliest start schedule can have a very irregular resource use, despite the fact that no resource conflicts occur.

## Schedule Risk Analysis

The solutions for the *schedule risk analysis* questions are displayed in Table 12.2. For the answers with an asterisk (*), a more detailed explanation is given in this section with reference to the article abbreviation and question number.
The list of comments is given along the following lines:

RA2–Q4: During Monte Carlo simulations, the critical path might change for every simulation run.
RA2–Q6: When the logic of the random generator discussed in the article is followed, the generated number will be equal to 5 since $0.20 < 0.379403 < 0.7$.
RA2–Q7: Each simulation runs generates numbers independent from the previous run.
RA3–Q2: The SPI(t) indicator should be compared against the value 1, and not zero.
RA3–Q5: The SPI(t) indicators reports a project delivery ahead of schedule, and the final result shows that RD < PD, illustrating a final early delivery.
RA3–Q6: The SPI(t) indicators reports a project delivery ahead of schedule, but the final result shows that RD > PD, illustrating a final late delivery. Exactly the opposite!
RA4–Q2: The standard deviation of a triangular distribution is equal to $\frac{a^2+m^2+b^2-am-ab-mb}{18}$ so it can be easily seen using values for $a$, $b$ and $c$ that the variances for distributions 1 and 3 will be the same, and bigger than the variance for distribution 2. See MS Excel solution file, tab "RA4".
RA4–Q3: Using interval estimates will produce several critical paths, i.e. one for each simulation run. In doing so, sensitivity metrics can be produced that express the likelihood of being a critical path. Finding a new deterministic path is not the goal of simulation.
RA4–Q4: The PERT technique makes some assumptions that are not 100 % correct and therefore the results of PERT might slightly differ from simulated

**Table 12.2** Solutions for the schedule risk analysis questions

| | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 |
|---|---|---|---|---|---|---|---|---|
| RA1 | – | – | – | – | – | – | – | – |
| RA2 | b | c | b | b* | b | b* | c* | – |
| RA3 | b | d* | c | b | a* | a* | – | – |
| RA4 | a | e* | b* | b* | – | – | – | – |
| RA5 | b | d | c | c | a | b* | – | – |
| RA6 | c | b | a* | c | b | b | – | – |
| RA7 | b* | d | c* | a* | b* | f* | b | a* |
| RA8 | f* | f | b* | c* | e* | f* | d* | – |
| RA9 | f | f | b* | c* | c* | b* | – | – |
| RA10 | f | f | b* | d* | d* | e* | e* | – |
| RA11 | c* | e | b* | c* | d* | f* | a* | – |
| RA12 | b* | d | d* | c* | c* | e* | – | – |
| RA13 | b | a | b | b | a* | – | – | – |
| RA14 | c | b | c | b | – | – | – | – |
| RA15 | d* | c | c* | d* | d* | b | – | – |
| RA16 | c* | b* | b* | d | b* | – | – | – |
| RA17 | b | c* | e* | b* | b* | – | – | – |
| RA18 | a* | b* | c* | c* | a* | – | – | – |
| RA19 | b* | c* | b* | c* | – | – | – | – |
| RA20 | d* | c* | b* | d* | c* | e* | – | – |
| RA21 | d* | c* | a* | d* | c* | – | – | – |
| RA22 | a* | a | a | b* | – | – | – | – |
| RA23 | a | a | a | a | b* | – | – | – |

results, even when the same distributions are used. See e.g. article "BS16" for more information.

RA5–Q6: Such an activity has a high probability of being critical, but the impact of unexpected changes in the durations will not have a huge impact on the project duration, and hence, it does not deserve much attention.

RA6–Q3: Since the CRI is measured as an absolute value, both values close to 1 and $-1$ will results in values close to 1.

RA7–Q1: In a serial network, every activity lies on the critical path, regardless of its duration.

RA7–Q3: The number of runs that activity A is critical $= 3$ and total number of runs $= 4$ so CI $= \frac{3}{4} = 0.75$. See MS Excel solution file, tab "RA7".

RA7–Q4: The number of runs that activity B is critical $= 2$ and total number of runs $= 4$ so CI $= \frac{2}{4} = 0.50$. See MS Excel solution file, tab "RA7".

RA7–Q5: When an activity is added to the network, the number of times the old activities A and B lie on the critical path can remain the same or reduce, but never increase.

RA7–Q6: The number of times that activity A will lie on the critical path will probably increase due to the longer path as a result of successor C. The number of times activity B lies on the critical path will decrease (or remain the same).
RA7–Q8: Activity A will be critical only when activity B has a duration of 10 (50 %) and activity C has a duration of 10 (50 %). The probability is equal to 50 % * 50 % = 25 %.
RA8–Q1: Since the CRI is measured as a correlation, any value can be possible.
RA8–Q3: See MS Excel solution file, tab "RA8".
RA8–Q4: See MS Excel solution file, tab "RA8".
RA8–Q5: See MS Excel solution file, tab "RA8".
RA8–Q6: Since there is no variability in the duration, the correlation has no value. See MS Excel solution file, tab 'RA8".
RA8–Q7: See MS Excel solution file, tab "RA8".
RA9–Q3: See MS Excel solution file, tab "RA9".
RA9–Q4: See MS Excel solution file, tab "RA9".
RA9–Q5: See MS Excel solution file, tab "RA9".
RA9–Q6: See MS Excel solution file, tab "RA9".
RA10–Q3: See MS Excel solution file, tab "RA10".
RA10–Q4: See MS Excel solution file, tab "RA10".
RA10–Q5: See MS Excel solution file, tab "RA10".
RA10–Q6: See MS Excel solution file, tab "RA10".
RA10–Q7: See MS Excel solution file, tab "RA10".
RA11–Q1: In a serial network, the value for the CI is equal to 1 for all activities.
RA11–Q3: See MS Excel solution file, tab "RA11".
RA11–Q4: See MS Excel solution file, tab "RA11".
RA11–Q5: See MS Excel solution file, tab "RA11".
RA11–Q6: See MS Excel solution file, tab "RA11".
RA11–Q7: See MS Excel solution file, tab "RA11".
RA12–Q1: For a serial project, SSL = 0 for all activities. The formula for the SI can hence be written as follows: SI $= E\left\{\frac{SAD}{SAD} * \frac{SPD}{E(SPD)}\right\} = E\left\{\frac{SPD}{E(SPD)}\right\} = 1.0$
RA12–Q3: See MS Excel solution file, tab "RA12".
RA12–Q4: See MS Excel solution file, tab "RA12".
RA12–Q5: See MS Excel solution file, tab "RA12".
RA12–Q6: See MS Excel solution file, tab "RA12".
RA13–Q5: The CC/BM technique allows the use of buffers to monitor the performance of the project, rather than monitoring individual activities, and hence, it can be considered as a top-down approach, similar to Earned Value Management.
RA15–Q1: This is exactly the principle of the net present value, as discussed in article BS25.
RA15–Q3: See MS Excel solution file, tab "RA15".
RA15–Q4: See MS Excel solution file, tab "RA15".
RA15–Q5: See MS Excel solution file, tab "RA15".
RA16–Q1: There is no such concept as cost buffer in the CC/BM method.

RA16–Q2: Given that the aggressive time estimates are used it is normal that a substantial fraction of the buffer will be consumed, only when the buffer consumption is more than could reasonably be expected the project manager should take action.
RA16–Q3: Ideally, buffers should be sized according to the variability in the chain (answer b), but maybe the total duration of the chain could also be taken into account (so answer c should also be considered as correct).
RA16–Q5: Resource buffers are not time buffers, but only warning signals to assure the resource is ready when the activity starts.
RA17–Q2: Critical Chain = Critical Path = 1-3-5-6-7 based on the aggressive time durations (total duration = 16 time units). See MS Excel solution file, tab "RA17".
RA17–Q3: Variance of critical chain = $1^2 + 2^2 + 0.5^2 + 0.5^2 + 1^2 = 6.5$.
RA17–Q4: Variance of feeding chain (2-4) = $1^2 + 1^2 = 2$.
RA17–Q5: $\sqrt{1^2 + 1^2} = 1.41$.
RA18–Q1: Length = 50 % of 16 = 8 time units. See MS Excel solution file, tab "RA18".
RA18–Q2: Length = 50 % of 6 = 3 time units.
RA18–Q3: The cut and paste method ignores the long tail and takes 50 % of the short durations, so it will probably underestimate the real duration.
RA18–Q4: A high value for the coefficient of variation means that standard deviation is likely to be high compared to the average durations. The cut and paste method ignores the standard deviation.
RA18–Q5: This activity has no risk and should therefore not be taken into consideration when buffering the critical chain.
RA19–Q1: $\sqrt{(6-4)^2 + (10-6)^2 + (2-1)^2 + (2-1)^2 + (6-4)^2} = 5.1$. See MS Excel solution file, tab "RA19".
RA19–Q2: $\sqrt{(5-3)^2 + (5-3)^2} = 2.8$.
RA19–Q3: Long chains will overestimate the size of the buffers since the variability will be summed up. However, in reality, the variability will cancel out at some places in the chain.
RA19–Q4: If it is assumed that the length of the critical chain is equal to its expected duration, then it can be found on the normal distribution that adding two times sigma to the average is equal to the 97.5th percentile, and the probability to exceed is then equal to 2.5 %.
RA20–Q1: This is the only arc that enters the critical chain.
RA20–Q2: A resource buffer should be added every time a new type of resource is used in the CC. In this scenario the CC changes resources twice, hence two resource buffers should be added: one for resource B and one for resource C. One could argue to add 3 resource buffers, i.e. also a resource buffer for resource A for activity 5.
RA20–Q3: The CNC is equal to the number of arcs over de number of nodes.
RA20–Q4: One could argue that some of the precedence relations are superfluous (arcs 2-5 and 2-6) resulting in a CNC = 3/4 but these precedence relation can not always be removed (e.g. if these precedences are different from the default

FS $= 0$ type, removing these precedences might lead to a violation of some of the intended precedence relations).
RA20–Q5: The total standard deviation of the activities in the feeding chain is equal to $\sqrt{0.5^2 + 0.5^2 + 1.5^2 + 0.5^2} = 1.73$, and hence, the size is equal to $1.73 * (1 + 5/4) = 3.90$.
RA20–Q6: The total standard deviation of the activities in the critical chain is equal to $\sqrt{2.5^2 + 2.5^2 + 3.5^2} = 4.97$, and hence, the size is equal to $4.97 * (1 + 2/3) = 8.29$.
RA21–Q1: One project buffer and one feeding buffer should be used. Assuming a buffer is used for each resource, the total number of buffers is $2 * 3 = 6$.
RA21–Q2: The critical chain is equal to 1-2-3-5-6-7 with length 19, and hence, the available work content is equal to 2 (= availability of resource A) * 19 = 38. See MS Excel solution file, tab "RA21".
RA21–Q3: Activity 4 has a duration of 3 and uses 1 unit of resource C, so the work content used is 3 * 1 = 3.
RA21–Q4: The tightest resource is B so K $= \frac{15}{19}$. The size of the critical chain is then equal to $2.739 * (1 + K)$ (the standard deviation of the CC $= \sqrt{1^2 + 1^2 + 2^2 + 0.5^2 + 0.5^2 + 1^2} = 2.739$). See MS Excel solution file, tab "RA21".
RA21–Q5: The tightest resource is C (only resource used by activity 4) (so K $= \frac{3}{6} =$ K. The size of the feeding chain is then equal to $1 * (1 + K)$ (the standard deviation of the FC $= \sqrt{1^2} = 1$). See MS Excel solution file, tab "RA21".
RA22–Q1: Buffers can shift activities in time, resulting in new resource conflicts.
RA22–Q4: After inserting buffers, the longest chain (critical chain) can be shorter than another chain. There is currently no good solution to this problem, only pragmatic solutions, as discussed in article "RA23".
RA23–Q5: If feeding buffers are randomly split, it is likely that the second buffer is too small compared to the high variability of the activities entering that buffer.

## Project Control

The solutions for the *project control* questions are displayed in Table 12.3. For the answers with an asterisk (*), a more detailed explanation is given in this section with reference to the article abbreviation and question number.
The list of comments is given along the following lines:

PC2–Q1: EVM is used both prior to the start (PV curve) and during progress (AC and EV).
PC2–Q3: PV can also be called budgeted cost of work scheduled (BCWS).
PC2–Q4: ES is expressed in time units.
PC2–Q7: Similar to EV = BAC at the end of the project, ES = PD.
PC3–Q4: The schedule adherence is expressed by the p-factor.
PC3–Q6: SPI(t) = ES/AT.

**Table 12.3** Solutions for the project control questions

| | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 |
|---|---|---|---|---|---|---|---|
| PC1 | – | – | – | – | – | – | – |
| PC2 | b* | a | b* | b* | a | a | b* |
| PC3 | a | a | a | b* | b | d* | – |
| PC4 | a | b* | b* | a | d* | c* | – |
| PC5 | b* | a | a | c* | a* | c* | – |
| PC6 | a | a | a | b* | a* | a* | – |
| PC7 | a | a | a* | b | d* | c* | – |
| PC8 | a | a | b* | a* | a* | a* | b* |
| PC9 | a | c* | b | c* | a* | b* | – |
| PC10 | b | b* | a | b* | b | a* | – |
| PC11 | b* | a | a* | b* | d* | c* | – |
| PC12 | b* | b* | b* | b* | b | b | – |
| PC13 | b* | a | a* | c* | a* | – | – |
| PC14 | b* | a | a | d* | a* | – | – |
| PC15 | a | a* | a* | b* | d | b* | – |
| PC16 | b* | a | b* | b* | c* | – | – |
| PC17 | a | b* | a | b* | a | b | – |
| PC18 | b* | b* | a | a* | a | d* | – |
| PC19 | a | b* | a* | b* | a | b* | a |
| PC20 | b* | a* | a | b* | a | b* | – |
| PC21 | a | b* | a | a* | a | – | – |

PC4–Q2: A project can have multiple critical paths.
PC4–Q3: B; The planned duration is calculated by summing the durations of the activities on (one of) the critical path(s).
PC4–Q5: All paths are critical except one (1-3-5-8).
PC4–Q6: A change in the activity cost has no impact on the duration of the project nor on the critical path, but on the total cost (budget at completion) of the project.
PC5–Q1: There is no information given on the expected delay of the project, thus the earned value can be situated both above or below the planned value.
PC5–Q4: EV = Percentage completion * BAC.
PC5–Q5: The actual cost lies above the earned value, so a cost overrun is to be expected. The earned value lies below the planned value, so a delay is to be expected.
PC5–Q6: The actual finish time is given in Fig. 10.5. The actual estimated cost = cost incurred until evaluation point (30 + 60 + 30 + 30 + 20) + planned cost for remaining activities (20 + 20 + 20) = 230.
PC6–Q4: When the project is finished, ES = PD.
PC6–Q5: Linear interpolation is used when the EV at an evaluation point is not exactly equal to any of the recorded planned values. This is then used to calculate the ES.

PC6–Q6: Both earned values will be projected on the same point on the PV-curve.
PC7–Q3: The SV(t) metric does not have a quirky behaviour at the end of the project, while the SV metric does.
PC7–Q5: AC > EV (cost overrun) and EV > PV (ahead of schedule).
PC7–Q6: AC > EV (cost overrun) and EV < PV (late).
PC8–Q3: SV is also expressed in monetary units (SV(t) is expressed in time units).
PC8–Q4: The SPI in period 8 = 120/140 = 0.86 > 0.69 (SPI in period 7), so improved schedule performance.
PC8–Q5: The CPI in period 8 = 120/200 = 0.60 > 0.53 (CPI in period 7), so improved cost performance.
PC8–Q6: CV = 70-120 = -50.
PC8–Q7: SPI = EV/PV and CPI = EV/AC. Since the EV line lies below the PV line, which lies below the AC line, CPI < SPI throughout the entire project.
PC9–Q2: Be careful since SV(t) = 0 means on time. Do not confuse SV(t) with SPI(t) (= 1 means on time).
PC9–Q4: The projection of the EV at week 5 on the PV curve shows that this occurs somewhere between weeks 4 and 5 on the horizontal axis.
PC9–Q5: At the end of the project, EV = PV and hence, SV = EV−PV = 0.
PC9–Q6: Is SPI(t) = 1 at the project end, it means the project finishes on time. SPI is always equal to 1 at the end of the project.
PC10–Q2: EAC(€) is a forecast to measure the expected deviation from the BAC, and can be equal to BAC, but also larger or smaller.
PC10–Q4: If AC > PV > EV, the project is expected to exceed budget and finish late. In this case, both CPI and SPI/SPI(t) are smaller than 1.
PC10–Q6: SCI = SPI * CPI.
PC11–Q1: $TV = \frac{SV}{PV_{rate}} = \frac{SV*PD}{BAC}$ which is not equal to SV = EV−PV.
PC11–Q3: Negative values are prohibited by the use of max(PD, AT) in the formula.
PC11–Q4: The performance factor is equal to 1, SPI(t) or SPI(t) * CPI.
PC11–Q5: None of them are always correct. In (a) CPI should be SPI, in (b) PD should be max(PD, AT), in (c) SCI should be SCI(t).
PC11–Q6: (a) $EAC(t)_{PV1} = PD\text{-}TV = PD - \frac{SV*PD}{BAC} = PD*(1-\frac{SV}{BAC})$, (b) $AT + \frac{AT-ED}{SPI} = \frac{AT*SPI+AT-AT*SPI}{SPI} = \frac{AT}{SPI}$, (c) EV should be ES.
PC12–Q1: The performance factor PF determines the future expected performance.
PC12–Q2: SPI = EV/PV and SPI(t) = ES/AT.
PC12–Q3: Both are dimensionless metrics, but the metrics used to calculated the indices, EV, AC and PV, are expressed in monetary units, while AT is expressed in time units.
PC12–Q4: The $PV_{rate}$ is used for time predictions when the planned value method is applied.
PC13–Q1: Interval estimates are used to incorporates uncertainty inherent to each project.

PC13–Q3: Summing positive and negative values can result in a lower absolute value for MPE than MAPE.

PC13–Q4: MAPE $= \frac{|9-10|+|10-10|+|9-10|}{3*10} = 0.0667$, MPE $= \frac{(9-10)+(10-10)+(9-10)}{3*10} = -0.0667$.

PC13–Q5: MAPE $= \frac{|140-150|+|150-150|+|160-150|}{3*150} = 0.0444$, MPE $= \frac{(140-150)+(150-150)+(160-150)}{3*150} = 0$

PC14–Q1: Higher values for the p-factor means better schedule adherence.

PC14–Q4: p = (min(10,10) + min(25,15) + min(5,5) + min(30,40))/(10 + 25 + 5 + 30) = 0.86. See MS Excel solution file, tab "PC14".

PC14–Q5: Since ES does not change, the vertical ES line in Fig. 10.16 remains at the same place. As activity 4 is more expensive than activity 7, a smaller portion of the activity is performed under risk (i.e. the portion of the activity progress to the right of the ES line), so the p-factor increases. More specifically, the new p-factor is (10 + 30 + 5 + 2.8 + 0 + 40 + 0) / (10 + 30 + 5 + 2.8 + 1.4 + 40 + 0.7) = 0.977 instead of (10 + 30 + 5 + 0 + 0 + 40 + 0.7)/(10 + 30 + 5 + 2.8 + 1.4 + 40 + 0.7) = 0.953. See MS Excel solution file, tab "PC14".

PC15–Q2: Plugging p = 1 into the formula results in EV(e) = EV.

PC15–Q3: Since p has a maximum value of 1, EV(e) must be smaller than or equal to EV.

PC15–Q4: R% represents the estimated portion of EV(r) that requires NO rework, so R% should be lowered when the risk of rework increases in a project.

PC15–Q6: Only (a portion of) activity 7 lies to the right of the ES line.

PC16–Q1: EV(e) is the sum of the risk-free earned value and a FRACTION of the risky earned value (i.e. the estimated portion R% that requires no rework).

PC16–Q3: These activities can still show PV and EV accrue deviations which decrease the schedule adherence.

PC16–Q4: It is possible that all activities show a uniform delay, meaning that no portion of the work performed lies to the right of the ES line. In this case, there is indeed a delay, but the delay is perfectly in accordance with the schedule, resulting in a p-factor of 1.

PC16–Q5: The schedule adherence is not related to AC.

PC17–Q2: Risk measures are defined on the activity level.

PC17–Q4: A higher degree of attention might be required, but this does not necessarily result immediately in corrective actions.

PC18–Q1: They provide information about the overall performance of the project.

PC18–Q2: One can easily set a dynamic action threshold, and it is advised to do so since the attention (thresholds) should change along the project progress.

PC18–Q4: More point will likely lie below the line, resulting in an increased intensity of control.

PC18–Q6: Normally, the EVM top-down project control approach provides one single metric at the top of the WBS, while the bottom-up control approach provides several activities to monitor at the bottom of the WBS, and hence, answer (a) would be the best answer. But since the question ignores the number of times problems are detected, it is quite likely that the top-down approach warns for many problems, and then the control approach is very intensive. Therefore, the best possible answer is (d).

PC19–Q2: If an action threshold is reached, the project manager will have to drill down to the activity level to identify the poor performing activities and take actions on them.

PC19–Q3: When action thresholds are lowered, it means they shift to the right, and then more activities have sensitivity values (e.g. for the SSI) that exceed the threshold, leading to a more intensive control approach.

PC19–Q4: When action thresholds are lowered, it means that the EVM metrics (e.g. the SPI(t)) will likely fall less below the threshold value, leading to a less intensive control approach. Note that this is only true when the thresholds lie below 1. A threshold above 1 (to detect opportunities instead of problems) will results in a more intensive control approach when the thresholds are lowered.

PC19–Q6: It is known that EVM works better for serial networks, while SRA works better for parallel networks.

PC20–Q1: An overridden logic violates precedence relations, but not always. Think e.g. of two activities without overlaps (no fast tracking). There is no reason to override the logic of the precedence relations.can always be activities for which the precedence relations are not violated when applying an overridden logic, e.g. when no fasttracking has occurred.

PC20–Q2: In a completely parallel network there are no precedence relations to be violated.

PC20–Q4: It is exactly the opposite.

PC20–Q6: $20 + 18 - 0.25 * 12 = 35$. The last term $(0.25 * 12)$ represents the reduction in project duration resulting from the 25 % overlap which is allowed when the percentage overridden/retained is 75 %.

PC21–Q2: The fixed cost of an activity is fully incurred at the start of that activity.

PC21–Q4: Since the fixed costs have been paid, only the remaining variable costs should be taken into account, and hence, PRC = 12 * € 75 + 18 * € 100 = € 2,700.