Fundamentals of Scheduling & Resource Leveling
Tammo Wilkens, P.E., PMP

Introduction

Developing a good schedule for managing the execution of a project is paramount to good project management practice. The Critical Path Method of scheduling is currently the most popular technique employed by scheduling practitioners. Understanding how a schedule is created, as well as the impacts of progress and disturbances, help the project manager and other stakeholders appreciate what they are looking at when presented with a schedule report. At the core of the CPM process are the forward and backward passes. These depend on good logic and they provide the earliest and latest dates of each schedule activity. This presentation is structured to remove the mystery of this process and will provide the reader with the skill to develop and analyze schedules. Covered issues include the use of various relationships, lags and constraints and their impact on the scheduling process.

Closely related to CPM scheduling, resource leveling appears to be a mysterious process that is not well understood. This presentation will take the schedule development process a step further by providing a clear picture of the process. Readers will gain a better appreciation of the process as well as the results. Covered issues include the priority setting for resource allocation and the trade-off between duration of the project and allocation of limited resources.

Scheduling

This is a presentation of the fundamentals of CPM scheduling to help the reader fully understand the mechanics of scheduling. The intent is to provide the concepts that will allow project team members and stakeholders to understand the published output of the scheduling process. The basic process involves the following steps:

1. Select a scheduling methodology
2. Select a tool that provides the computing horsepower and presentation facilities; this is known as the scheduling model
3. Apply the relevant project parameters to the scheduling model to create the specific project schedule model
4. Analyze the various project schedules produced by the project scheduling model to assess the resulting project execution plan
5. Adjust the project parameters to achieve a satisfactory plan
6. Publish reports for the project team and stakeholders
7. Periodically update the activity progress to reflect the actual execution of the project plan and determine whether the remaining work and execution plan are still valid
8. Adjust the plan of the remaining work as required.

Use of CPM networks is currently the most popular of the available methodologies. Other methods include manually drawing bar charts, critical chain, horse blanket, and others.

While manual methods of scheduling are feasible for small projects with just a few activities, the vast majority of scheduling is done with the aid of computer programs designed for the specific methodology. These tools also have a repertoire of screen layouts and output reports.

The relevant project parameters include the list of activities, their duration, and the interconnected logic of the sequence of execution. By applying these to the scheduling model, the project specific model is created. This, in turn, provides the means to adjust the parameters to develop various execution scenarios for the project.

Periodic updating of the schedule progress is crucial to the successful execution of the project. Events and progress simply does not proceed exactly as originally planned. Just as the original, total project execution has to be planned, so does the remaining work at every update cycle.
The Critical Path Method

The Critical Path Method (CPM) is the backbone upon which the project execution plan is built. It provides the dynamic analysis of activity sequencing and progress reporting, the foundation for advanced progress and performance analysis, such as earned value management, as well as a tool for resource deployment and management.

The CPM is a relatively simple concept. It consists of linking the activities that make up the project execution in a logical, sequential manner such that the execution is both practical and generally as fast as possible. For example, “Pour Foundation” activity must follow after the “Excavate Foundation” activity. Barring some revolutionary new construction method, anything otherwise is simply not practical. However, if the foundation is of a long physical nature, it may be possible to overlap the two activities by delaying the pouring some finite time behind the start of the excavation, rather than waiting for the completion of the total excavation before starting any pouring. The point of this is that the practicality of activity sequencing is limited only by the feasibility of the actual work.

The speed of execution is limited by another factor: the availability of the required resources. By resources we mean the personnel, materials and equipment that is necessary to perform a particular activity. Resource assignment and leveling is the subject of the second part of this paper.

By tying all the activities into a logical, sequential network, the CPM calculation can then proceed to determine the earliest possible time activities can be accomplished and the latest possible time activities need to be accomplished in order to meet the project completion date. The primary restriction of any CPM is that the total project execution must fit between the start and finish dates of the project. As we will see later, the CPM also provides the means to fit a certain set of activities between some intermediate dates between the project start and finish.

The CPM calculation consists of two parts, generally done one after the other, in the same process: The forward pass and the backward pass.

The Forward Pass

Generally, the first part of the CPM calculation is the forward pass. It consists of determining the earliest date for each activity, starting with the first activity in the CPM network. Essentially, it sets the early start date (ES) of each activity by selecting the latest of the following dates: The project start date (PS), the early finish date of all of its predecessors plus 1 (EF), the date of all “not earlier than” constraints placed on that activity (NET) and the current Data Date (DD). Graphically, this can be viewed as all activities sliding to the left until they “bump into” one of these four items. (Exhibit 1) For example, Activities A and D bump into the project start on the left; Activity C bumps into the finish of Activity B. The calculation for the early start date is:

ES = Greatest of:  
PS/EF+1/NET/DD

The early finish dates for each activity are of course determined by the duration of each activity. The calculation for the finish date is:

Finish = Start + Dur –1

For Activity B this is: 6+10-1=15.

The reader must keep in mind that the calculation can vary with the specific calendar used. The above formula holds for calendars that start the day on or after midnight and end on or before midnight.
Calculating the finish of each activity is important as that determines the earliest start for any successor activities. The forward pass applies these algorithms from left to right through the entire schedule network until all activities have an early start and an early finish date assigned.

**The Backward Pass**

The second part of the CPM calculation is the backward pass. It consists of determining the latest date for each activity, starting with the last activity in the CPM network. Essentially, it sets the late finish date (LF) of each activity by selecting the earliest of the following dates: The project finish date (PF), the late start date of all of its successors minus 1 (LS) and the date of all “not later than” constraints placed on that activity (NLT). Notice that the Data Date does not factor into the backward pass. Graphically, this can be viewed as all activities sliding to the right until they “bump into” one of these three items. (Exhibit 2) For example, Activity C bumps into the project finish on the right and Activity E bumps into the start of Activity C. The calculation for the late finish date is:

\[ LF = \text{Least of: } PF/LS-1/NLT \]

The backward pass starts with day 22 as the default project finish date. This date came from the forward pass calculation and is the last early finish date of all activities in the project.

The combined picture of the forward- and backward passes looks as shown in Exhibit 3. The dates in this illustration are ordinal dates to facilitate the review of the formulas. The early dates, which resulted from the forward pass, are shown on top of each activity bar and the late dates, which resulted from the backward pass, are shown below each bar.

Since no project required finish date was specified, the backward pass started the calculations from day 22 as the default project finish date. This date came from the forward pass calculation and is the last early finish date of all activities in the project.
Total Float

Total float is a value that is used in analyzing CPM schedules. It is calculated for each activity by subtracting the early finish date from the late finish date. Each activity has its own total float value. Depending on the logic and the calendars assigned to each activity, activities that are in a sequential string of logic, will have the same amount of total float. See Activities A, B & C in Exhibit 4.

The benefit of total float is that it indicates the number of days that an activity can slip without delaying the completion of the project or delaying the constrained milestone at the end of its critical path. If the total float is negative, it means that the project cannot be finished at the desired time without changing the logic or durations. Please note that negative float can only exist if the finish date of some activity is constrained to finish no later than a certain date. In the network displayed in Exhibit 3, there cannot be any negative float since the project finish is allowed to be day 22, the end of the forward pass calculation. Suppose that a constraint is applied to Activity C to finish no later than day 20. That would make the late finish date for Activity C 20, not 22. In that case, with the present logic and durations, the project would have a negative float of 2 days and all activities on the sequential string of logic (A, B & C in the example) would have a total float of -2. All other logic paths in the network would be affected similarly. Activities D and E would have their total float reduced from 4 to 2. If this were the case, the practitioner could use a different type of relationship between Activities B and C. For example, he or she can use a “start-to-start” relationship with a lag of 8. This assumes that eight days of work on Activity B is sufficient to permit the start of Activity C.

Intermediate constraints also affect the total float of some activities. Suppose that it was necessary to finish Activity E by Day 10. In that case, a finish-no-later-than constraint of Day 10 should be applied to Activity E. This would result in a total float of -1 for Activities D and E. The solution to this dilemma might be to allow Activity E to start before the full completion of Activity D. If the work can be done physically, the practitioner can apply a negative lag of 1 day to the relationship between Activities D and E.

Free Float

Free float is an often overlooked key to keeping projects on time. The free float is the number of days that an activity can be delayed without impacting the start date of any of its successor activities. In Exhibit 4, the free float of Activity A is 0 as any delay to finish A would impact the start of B directly. Similarly, Activity D has a free float of 0 as any delay in its finish would impact the start of Activity E. Activity E is the only activity with any free float. In this example, the free float of E is 4 days, coincidently the same as the total float.

The significance of free float in managing a project cannot be overstated. Suppose that Activity E represents trial testing of a new software application by a group of testers. The testing is to be done on the premises under controlled conditions and with immediate feedback. The testers are to be flown into town from various parts of the country. Now suppose that the program development group is thinking that they have 4 days of total float, so a delay of two days won’t do any harm. However, the carefully planned trip of the testers is now in jeopardy. Similar situations abound in any kind of project. Subsequent resources are usually scheduled and any upset to that schedule can wreak havoc. However, as long as you manage the free float and don’t exceed it, your project is guaranteed to finish on time!
Schedule Analysis

Schedule analysis often takes the shape of looking at a schedule printout in either tabular or graphic format and seeing how the project finish is impacted. Proper schedule analysis includes comparing actual progress against planned progress and determining the reasons for any slip in completion dates. That must be followed by a review of the remaining work and perhaps adjustments to the logic or durations to bring the project back into the planned timeframe for completion. This is no different than the original project planning; it needs to be done with every update cycle.

Part of the schedule analysis process is the publication of the schedule information. Popular practice is the use of bar charts to depict the status of the project plan. Exhibit 5 shows a typical layout of a bar chart to display the previous and current status of the schedule. The yellow bars represent the progress from the previous update cycle. Here it is quickly evident how the schedule is either improving or slipping. The critical path activities have all slipped a day from the previous update while Activity E has finished a day early. Although Activity B has also finished a day early, Activities H and G have slipped because no progress has been reported on them.

Resource Leveling

A project requires resources to execute the activities. These resources include the labor, equipment and materials required to get the work done. Labor is the people, such as craft, engineers, programmers, systems analysts, etc. Equipment includes such things as cranes, test rigs, process simulators, etc. Materials includes such things as the concrete to be poured, the wire to be installed, etc. In the ideal world, resources are unlimited and available as required. Yet, resources are generally not unlimited and the project team needs to “level” out the use and consumption of resources.

The Challenge

The basic challenge of the planner is to develop a project execution plan that fits the allotted time and can be achieved with the available resources. As there are many activities vying for resources at various times, it is a complex juggling job to try to schedule the activities so that they will not be held back by the lack of resources.

Leveling Criteria

When the computer is asked to do the leveling, the program needs criteria for decision making. The initial criterion is to complete each activity by its baseline date. The baseline date may be determined in several methods. One can simply adopt the early schedule as the baseline, but that is generally not achievable as things happen and the early dates are missed. One can simply adopt the late dates as the baseline, but that is dangerous as it tends to hide schedule problems until it’s too late. A better approach is to adopt a schedule that lies in between the early and late dates. Using a resource leveled schedule to define the baseline dates is generally a good practice. It takes into account when resources are available and therefore provides a plan of execution that has a better chance of achievement.

A second criterion is that the “most important” activities are scheduled first when there is a limit of required resources. The total float value of each activity is most often used to determine the “importance” of an activity.
However, other indices of importance can be used as well. Some programs allow the assignment of a priority code to each activity that is then used to set the order in which activities are scheduled.

**Preparation**

When a schedule is first developed, it must be done without regard to the availability of resources. Failure to do that will result in logic and constraints that inhibit the optimization of the CPM algorithm. The availability of resources is much too complicated to analyze manually as the schedule durations and logic is developed. Therefore, if the resource availability is considered when determining how long each activity should take and what the logic should be, the CPM network will have durations and logic that may be less than optimum. This should not be confused with taking the amount of a resource required for an activity into consideration to determine the duration of an activity. For example, physical work is often geared to the amount of a resource and the rate of application or consumption of that resource. Take the activity of pouring concrete. One needs to know at what rate the concrete can be poured and how much concrete is to be poured. That will determine the duration of the activity. However, the availability of the number of concrete pumps and the availability of the craft people who will pour the concrete, must not be considered when initially establishing the activity duration.

Once the CPM network of activities has been established without regard to the availability of the required resources, the practitioner can turn to the task of analyzing the amount of resources required at each time interval. Generally, when the resource library is established, the maximum amount of each resource is entered for use in the leveling process. More advanced programs allow the setting of additional levels of availability. This is useful in the case where the normal maximum level reflects the current staffing for the project (in the case of labor) and the higher limit reflects the additional staffing that could be hired on an emergency basis.

The next step is to determine what to use as the prioritizing index for ordering the scheduling of activities competing for the same limited resource. As mentioned above, the most common index used is total float. This paper will be limited to using that index. Other indices will function similarly.

At this point, the resource leveling algorithm can proceed to establish new start and finish dates for each activity. When there are many different resources assigned to a project, the resource leveled solution tends to be more difficult to achieve in the desired time frame. It is generally accepted practice to select those resources for leveling that are most “significant” on the project or the most difficult to augment or reschedule. To illustrate, let’s assume that a project requires millwrights, laborers and a crane. The crane is limited to a certain time frame of availability, millwrights are a skilled craft with limited availability, laborers are readily available to add to the project, but the project only has a limited number on staff. In this case, it is better to omit the laborers from the leveling process. This is important as the project completion will be pushed out further the more resources are leveled.

**The Process**

The process of resource leveling will result in a new schedule with “resource leveled” dates. We will call these the “resource start” (RS) date and the “resource finish” (RF) date. These dates cannot be any earlier than the early start and finish dates as the program is restrained by the network logic. Depending on the goal of the resource leveling process, the dates may or may not be later than the late start and finish dates. If the goal is to keep the leveled schedule within the project finish constraint, then the resource leveling solution may indicate that the project requires more resources than made available. If the goal is to execute the project with the available resources, then the resource leveling solution may indicate that the project requires more time to finish than originally planned.

The program will analyze each time period of the schedule calendar and determine how many resources are required by all activities that were originally scheduled to be active in that time period. Within each time period, it will rank the currently planned activities by the leveling priority index. We will use total float in this presentation. Starting on Day 1, if two or more activities require millwrights, the activity with the lowest total float will be scheduled first. That is, it will be assigned a RS and RF. If additional resources are to be leveled, each of them is analyzed to determine if an activity can start because all of the resources are available. If any on of the resources is not available, then the activity is delayed for consideration in the next periodic time slot. Note that once an activity is scheduled, it will continue uninterrupted until it is finished and its resource assignments will be scheduled. Hence, both RS and RF are assigned in the same periodic iteration of the leveling process. If there are enough millwrights...
available for the other activities, they will also get scheduled. If there are not enough millwrights available, then the activities with the higher total float will not be scheduled in that time period.

The program moves on to Day 2 and analyzes that time period in similar manner. Keeping track of the activities that are already scheduled, it will determine if there are enough resources available to start another activity. That would be the case if either an activity using millwrights finished and “released” those millwrights to work on another activity, or if more millwrights became available. For example, a second shift was mobilized or more millwrights were hired. The program will schedule the activities with the lowest total float, and delay the start of those activities with the higher total float.

This process continues until all activities have been scheduled; that is, received a RS and RF date. The scheduling of activities will take into account which of the two goals (limit the resources or constrain the project finish) has been established. Often practitioners will run the calculation both ways to see the difference in the results. If in the time period under consideration there is not enough of a resource available, the program will do one of two things. If the goal is to limit the resources, it will delay the activity even beyond its late dates until the resource becomes available. If the goal is to constrain the project finish, it will schedule the activities even though it would require more resources than are currently available. The report of the leveling calculation would then show how the resource limits are exceeded and by how much. The practitioner will then have to determine how to deal with the dilemma of insufficient resources: hire more or stretch the project duration. Another alternative is to review the project execution logic and try a different plan. Then the resource leveling process can be run all over again to determine if that produces a more acceptable solution.

The Baseline

The end result of the resource leveling process is to assign a RS and RF to each activity. These dates must then be captured and stored as the “resource leveled schedule.” After the resource leveled dates are determined, recalculating the schedule will wipe out the resource leveled dates unless they have been stored. The mechanism will depend on the actual software being used. The ultimate desired outcome of a resource leveling exercise is to generate a baseline schedule that is both achievable and makes economic use of the available resources. Ideally, this will fall somewhere between the early start schedule and the late start schedule. But it is up to the practitioner to select the most desirable solution that is available.

Conclusion

Using a scheduling model to create the project schedule model provides the foundation for developing good project schedules. The project schedule model indicates the various options for executing the project. This includes following the early start schedule or a baseline schedule. The baseline schedule can be generated from a resource leveled schedule. Resource limits often force a trade-off between stretching the project completion out or finding some way to obtain more resources. A full-featured software application will have enough facilities to let the practitioner establish resource priorities, secondary resource limits and to set the goal of limiting the resources or limiting the project completion. It will also generate a comprehensive report to let the practitioner analyze the trade-offs made in each time slot of the leveling calculation.