Problems, Methods and Tools of Advanced Constrained Scheduling

Victoria Shavyrina, Spider Project Team
Shane Archibald, Archibald Associates
Vladimir Liberzon, Spider Project Team

1. Introduction

In this paper we will review project constraints that must be taken into account when a project schedule is created. We will also review problems and challenges of constrained scheduling.

The traditional Critical Path Method was developed in the middle of the last century for calculating project schedules and takes into account only network dependencies and known estimates of activity durations. But logical constraints are only a small part of project constraints that must be considered when planning and scheduling a project. With increased power of modern computers we can go beyond CPM and create realistic and feasible schedules.

2. Project Information and Constraints

2.1. Cost Components and Centers

Project cost consists of cost components like salary, cost of materials, cost of machines, indirect cost, etc. It may also be necessary to group cost components into cost centers.

Cost components may use different currencies so it is necessary to define the currency for which the overall project cost and cost centers will be calculated.

Some cost components may represent project financing or revenues.

2.2. Activity Information

For some activities duration is estimated information and does not change when adding, removing, or changing resources. For other activities, duration depends on the volume (amount) of work to be done and overall productivity of assigned resources. When assigned resources are known and their productivity is known, activity duration can be calculated by the following formula:

\[
\text{Activity Duration} = \frac{\text{Volume}}{\text{Total Productivity}},
\]

\[
\text{Total Productivity} = (R_1^N \times R_1^P) + (R_2^N \times R_2^P) + \ldots + (R_n^N \times R_n^P)
\]

-where \( R_i \) = Resource \( i \), \( N \) = number of resources, \( P \) – productivity of resources

But in many cases activities need certain resource skills that can be satisfied by many different resources. For example: an activity requires an excavator and there are several excavators (excavator 1, excavator 2, excavator 3, etc.) with different capacities that may be assigned. Assigning a skill (“excavator”) instead of a physical resource (excavator 1) and instructing the software to assign physical resources on project activities when optimizing a project schedule is called “skill scheduling.” In this case, the activity duration is not known before the specific resources are selected by the software and the schedule is created.

An activity may be interruptible or not interruptible. If an activity is interruptible then resources assigned to perform that activity may be moved to another activity with higher priority and then returned to finish the work.
An activity may be continuous. This means the activity is not permitted to start the work at the end of one work day and finish at the beginning of the next day. If a continuous activity cannot be finished before the end of a work day, its start will be delayed until the beginning of the next work day. An example of such an activity is concrete work.

Activities may be of ASAP or ALAP types. ALAP type means the activity will use all Free Float – the activity will be performed as late as possible but without delaying other activities.

Activity costs (by components) may be assigned as fixed, per volume unit and per work hour.

Activity calendar defines when an activity may be executed.

2.3. Renewable Resource Information
Renewable are those resources that can be used again after they finish executing some activity (people, machines).

Renewable Resource information may include their quantities by periods, resource calendars (defines regular work hours and possible overtime hours), costs per regular work hour and overtime costs that may be different for different periods, material requirements per work hour (gas consumption, for example), and skills (what types of work may be performed by these resources).

Renewable resources may be produced (mobilized) or consumed (moved to another site) on project activities.

Renewable resource quantities may be constrained for some resources. For others, project management software will determine required quantities at any moment of project execution.

2.4. Consumable Resource information
Consumable resources are spent on project activities (materials or installed equipment, for example). Consumable resource data includes costs per unit that may be different in different periods.

Consumable resources may also be produced on project activities (supplies).

If consumable resource supplies are limited these limitations become schedule constraints.

2.5. Activity dependencies
Standard dependencies (FS, SS, FF and SF) are of No Earlier Than type. Succeeding activities may start after dependency conditions are met. But in some cases it is necessary to apply strict dependencies when the next activity starts immediately after the dependency conditions are met.

Dependencies may include time and volume lags.

Time lag means that a certain amount of time must pass after a dependency condition is met before the succeeding event (activity start or finish) may happen. Time lag may have its own calendar that is not the same as calendars of dependent activities.

Volume lag defines an amount of work on a preceding activity that must be done before a succeeding event may happen (for links from activity start) or an amount of work on a succeeding activity that must remain incomplete (for links from activity finish).
In some cases it is useful to apply *Double dependencies* – when a certain amount of work on a preceding activity must be done before a certain amount of work on a succeeding activity can be completed. This means double dependencies can include two lags – one on the preceding activity and one on the succeeding activity.

Dependencies may be *hard* (mostly technological or physical limitations; aka PDM logic) or *soft* (implied by people decisions; aka “preferential” logic). Soft dependencies can easily change and it is a “nice to have” option in the planning and scheduling process.

Dependencies may be created by resource limitations. If one activity is delayed because required resources are performing another activity, then these two activities are linked with a *Resource dependency*. Resource dependencies are not defined by people – they are the result of calculating resource constrained schedules and depend on applied resource leveling algorithms. These dependencies are considered when a project schedule is calculated.

### 2.6. Renewable Resource Assignments

When assigning renewable resources, it is necessary to set their quantity required for activity execution as well as their planned workload (percent of work time that will be used for activity execution). For example: an activity has a duration of 6 hours and 2 units of a certain resource will be involved in activity execution for 3 hours each – it means that 2 resource units are assigned with 50% workload.

But this data may be uncertain. It is not rare when an activity may start with some minimal amount of resources and when additional resources become available, they join and accelerate activity execution. This means it is necessary to assign not a certain quantity but rather the range of resource quantity and workload necessary for activity execution. We call this variable resource assignment.

When initial activity information is “volume of work to be done” in physical units (meters, tons, etc.), it is necessary to set the productivity rates of assigned resources (like meters per hour).

Another option is to assign resource skills to activities and set the number of required resources that have necessary skills. When assigning skills to activities with Volume of work, we may also want to set the total productivity required of assigned resources rather than the number of resources. In this case, the number of resources assigned will depend on the productivity of resources that are selected during optimization and the volume of work to be completed.

### 2.7. Consumable Resource Assignments

Consumable resources may be assigned directly to activities or they can be assigned to activity resource assignments if we want to specify what renewable resources consume certain materials and in what quantities (excavators and fuel, for example).

Consumable resources may be assigned as a fixed amount per activity or assignment, as a certain amount per volume unit, or an amount per hour.

They may also be consumed by resources as defined in their properties.

Consumable resources may also be produced on project activities (supply activities). This production can be modeled as fixed, per activity volume unit, per calendar hour.
2.8. Conditions
Sometimes the activity network is conditional. This means that a set of activities to be performed and their dependencies may depend on some results or condition (For example, if some event is scheduled during winter then the following activities will not be the same as they would in Summer, or if some milestone will be met with a delay, certain corrective actions will be applied). In this case the network shall include options that will be selected during project scheduling.

3. Constrained Scheduling

3.1. Scheduling tasks and parameters
Project schedules need to take into account all schedule constraints including:
- A quantity of limited renewable resources available at any given time,
- Material and equipment availability at any given time,
- Sufficient financing at any given time.
The schedule should have minimal possible durations and should take into account all other restrictions that were described earlier.

Critical Path Method scheduling is well defined and has a static mathematical solution. Using CPM scheduling software calculates the dates of activity execution and different floats and other schedule parameters with the assumption that project resources are unlimited.

Activity floats include:
- **Total Float** – time period of potential activity delay that does not cause a delay of the project finish or any target date set in the project schedule
- **Free Float** – time period of potential activity delay that does not cause a delay of any other activity in the project schedule

Both total and free floats may be applied to activity start and activity finish.

In addition to Total and Free Float, there is **Super Float** – the time period of potential activity finish delay without delaying the activity’s start and that does not delay the project finish date. Super Float shows what maximum deceleration of activity execution is acceptable.

Flexibility of activity start and finish dates are also characterized by start and finish Activity Flexes:
- **Finish Flex** shows the potential delay of the activity finish without the moving activity start date and that does not delay any other schedule activity
- **Start Flex** shows if Activity start may happen earlier with the same activity finish without changing any other dates in the current schedule

Knowing activity flexes and super floats allows managers to optimize resource assignments on specific activities and to understand if activity performance can be interrupted and for what duration.

Another parameter that is called **DRAG** shows an impact of the project activity on project duration. For example, Activity A is critical and has 10 days duration, but its impact is only 5 days because if activity A duration becomes less than 5 days another path will become critical. So activity A DRAG is 5 days.

All these parameters are useful for project schedule analysis and decision making.
3.2. Project constrained scheduling

3.2.1. Resource Leveling Heuristics

Constrained scheduling is much more complex and different packages produce different results with the same initial data. Since resource constrained schedule optimization is NP-hard, it is a mathematical problem—its solution can only reasonably be found manually for very small projects. That is why project management software packages use leveling heuristics.

These heuristics define certain rules that prioritize activities for limited resource assignments. Some packages use certain predefined priority rules (like MS Project), others ask users to select certain priorities from a set of predefined rules (like P6). These approaches do not even try to find the optimal solution. But with the possibility of trying different heuristics, the chances of finding a better schedule are significantly improved.

3.2.2. Resource leveling problems

Project “Test” consists of 4 activities and Finish milestone, and uses two resources (A and B). Below is the schedule created for this project by MS Project 2013:

![Gantt Chart](image)

The schedule created by MS Project 2013 lasts 60 days and total slacks are obviously wrong: activity 3 is shown as critical though it has 19 days free float, activities 1 and 2 are critical (their delay will delay the project finish date) but MS Project shows that both activities have 20 days total slack.
Let's look at what happens if the last activity of this schedule is one day longer:

Now MS Project selects the optimal sequence of activity execution but still shows wrong total slacks for activities 3 and 4.

This illustrates one more problem that needs to be considered when resource leveling is applied: resource constrained schedules created by project management software are not stable. Small changes in activity durations may have huge impacts on the sequence of activity execution and overall project duration.

A default resource constrained schedule created for this project by P6 is as poor as the schedule created by MS Project:

Free Float of activity 2 in this schedule is amazing.
But P6 permits us to use other leveling priorities and if we select Original Duration as the main priority for resource leveling then the schedule will be better, but the floats will remain wrong:

But the best heuristics for one part of the project may be the worst for another. So this approach does not guarantee obtaining resource constrained schedules that are close to optimal.

We know only two project management packages that attempt to optimize a project schedule. These packages are Spider Project and Aurora. We did not test Aurora capabilities but the Spider Project leveling engine includes the Optimization option that automatically produces resource constrained schedules that are at least close to optimal.

Let’s look at the schedule created for the same project Test by Spider Project:

It shows that all activities are critical in the optimal schedule that was automatically created. It means that activity floats in this resource constrained schedule are zero. Spider Project also shows resource
dependency as the dotted line between activities 3 and 1. Resource dependency shows that activity 1 was delayed because a required resource was busy on activity 3.

Activity floats in resource constrained schedules are called Resource constrained floats. If resource constrained floats are known then it is easy to find the Resource Critical Path (RCP) the same way as traditional Critical Path is determined. RCP is also known as Critical Chain.

3.2.3. Manual Leveling

Some people state that they level their projects manually, creating so called soft dependencies or preferential logic that define the order of activity execution. There are several problems with this approach:

1) It is not likely that near optimal solutions will be found manually. Like in our Test project, selecting the wrong resource assignment on any step may lead to poor results in the future. To find the optimal solution, it is necessary for us to consider and compare many options that can be done by the software but is too hard to be done manually

2) When the project is executed, some deviation from the initial plan will certainly happen. It means that the project will be rescheduled based on current estimates of remaining work. In this case, existing soft dependencies would need to be removed and rebuilt. This is an incredibly time consuming job that people will not do after entering actual information for each update cycle. It also means that between schedule updates the sequence of activity execution will not be optimized

3) Creating soft dependencies is difficult, as can be seen in the following example

Project “Soft” consists of three independent activities of the same duration that require the same resource (A). Let’s assume that only 2 units of resource A are available. How do we create soft dependencies in this case?

It is not clear if activity 3 should follow activity 1 or activity 2. After the finish of either of them, activity 3 may be executed. If we will link activities 2 and 3, this link may delay the
project finish date if activity 1 is completed faster than activity 2 like in the following screenshot. In the schedule below you can see that activity 3 could start on June 21 instead of June 25 and the project could finish earlier.

Manual leveling and soft dependencies created for manual resource leveling cannot replace a good software tool that evaluates many options and selects the best order of activity execution without creating artificial links.

3.3. Other scheduling options

3.3.1. Cost and Material constrained scheduling

Other schedule constraints may include cost and material supply restrictions that must also be considered.

This means the scheduling software should model not only expenses but also incomes; not only material consumption, but also material production and supply.

Space constraints are usually modeled as material constraints. Space is consumed when resources start to work at a certain location and is produced when they are moved to another location.

Project management software needs to delay project activities that require materials that are not available yet and if project financing at some moment does not cover activity costs.

3.3.2. Advanced methods of constrained scheduling: shift work simulation, skill scheduling and variable resource assignments

Another challenge to proper resource constrained scheduling is the simulation of shift work. It is not known before project scheduling which shift will be able to start certain activities and when they will be finished. The software will assign those resources that will be available at the period of activity execution.

Skill scheduling is applied when there is a choice of resources to use for certain activity executions. The software selects those resources that are available at the scheduled time and will do the job with maximum efficiency. Efficiency criterion may be selected by the user. In Spider Project, the default criteria for resource selection is minimal cost of activity
execution but users may select other criteria like maximum productivity (minimal activity duration).

An additional option of resource constrained schedule optimization is the application of variable resource assignments. Assigning a range of resource requirements (for example, “three to five resources”) will allow maximum resource utilization.

3.3.3. Schedule stability

Earlier in this paper we showed that a resource constrained schedule created using some activity leveling priority is not stable. Small changes in durations of some activities may create large changes in the project schedule, including changes in the sequence of activity execution. These changes may be unacceptable for many reasons including contract conditions. Even if the recalculated schedule results in faster project execution, changes in the sequence of activity execution may be unacceptable. So it is necessary to be able to reschedule the project while maintaining the order of activity execution that was previously accepted.

Schedule stabilization is no less important than schedule optimization. Spider Project allows users to maintain activity sequencing when rescheduling an optimized schedule.

4. Conclusions

Project schedules must consider all existing constraints including resource, supply and financing restrictions. It should model real life work, taking into account working in shifts, activities where execution cannot be split, resources that may replace one another, and everything else that is taken into consideration by people that create resource schedules manually. If the software cannot consider real life restrictions, the schedules created by this software will not be practical.

Resource constrained schedules created by different packages for the same project may be different. If the allows the use of different leveling priorities it can be useful to try different options and select the best solution for current project. However, a priority rule that is the best for one project may be the worst for another. This statement even applies to sections of the same project! It is obviously better if the software tries to optimize the resource constrained schedule and may guarantee that the resulting schedule is close to optimal. Unfortunately few packages do this.

Manual leveling is not practical. For large projects it is very time consuming, does not guarantee good results, and cannot be done with each schedule update.

It is necessary not only be able to find shortest resource constrained schedule for the current project but also to create resource constrained schedules that maintain a certain order of activity execution.

A Project Schedule should consider all existing constraints including resource, supply, space, and financing constraints and should maximize resource utilization resulting in minimal project duration.