MISTA 2013

Scheduling with MS Project: an add-in

André Filipe Macedo Guedes • Anabela Pereira Tereso

1 Introduction

The main objective of this research was to integrate scheduling techniques, documented in the literature, with Microsoft Project 2010, in order to help project managers' deal with scheduling tasks in an easier and controlled way, and at the same time achieve better results.

As far as we know, there is no information about the techniques or algorithms used by Microsoft Project to do the scheduling tasks so the possibility to select the algorithms to use gives a higher degree of control to the project manager.

The initial hypothesis was that it is possible to get better results, concerning the project's duration, using the studied techniques rather than the default scheduling technique used by Microsoft Project 2010, with an increased control.

2 Project Scheduling and Scheduling Techniques

The most supported motto, by a vast majority of authors, is that project scheduling is the most important task for the success of a project. The more detailed and clear the scheduling is, the easier will be the progress of the project. There will be greater interconnection between the players involved, with consequent maximization of the efficiency of resources utilization, minimization of costs, and higher profits.

But what is, exactly, project scheduling? Scheduling is to set the guidelines that the project team has to follow to achieve success. These guidelines indicate which tasks need to be done and which resources are needed to implement them. They help to meet the customer's expectations concerning delivery dates, cost and work accomplished. The ultimate goal of a project manager is to fulfill these expectations in an optimal way, if possible.

Project scheduling, despite being detailed and objective, can be reduced to three questions: what, how and when do things need to be done?

The most important aspect, regarding the scheduling phase, is the actual tasks' schedule. The scheduling is a way to indicate the sequence in which the activities of a project will be implemented. From this sequence of events, the project manager will be able to make estimations regarding the time of beginning and ending of each activity. These values will be crucial, so that the goal for project completion, established with the client, can be met.

But building this sequence of activities is not a trivial task. To be able to proceed with the scheduling of the project's activities it is necessary to take into account a range of details and constraints that will directly impact on this task. Each activity is characterized by an amount of time needed for its execution (the duration), an amount of resources needed to support the implementation and a set of precedence relations with other activities. All these parameters are important when allocating an activity. It can't be scheduled to run when there are insufficient resources or when one of their predecessor activities, if any, is not yet completed. This problem is known in the literature as the Resource Constrained Project Scheduling Problem (*RCPSP*).

A simple solution for the scheduling problem, without considering resource constraints, would be to make use of the slack of an activity to move its execution time to a moment that does not interfere with another conflicting activity, nor the total project time, whenever possible.

André Filipe Macedo Guedes
Anabela Pereira Tereso
CGIT Research Center
University of Minho, 4800-058 Guimarães, Portugal
afm.guedes@gmail.com; anabelat@dps.uminho.pt

The *RCPSP* aims to find the time instants in which the activities of a given project should begin, subject to precedence and resource constraints, in order to minimize the project duration.

Due to the precedence relationships, it is common for an activity a_j to have more than one predecessor. If these predecessors do not finish at the same time, activity a_j will have to wait until the last one finishes, before it can start its execution. This makes the activities that have ended earlier as predecessors to have slack time.

The scheduling techniques used in the developed add-in were: Early Start Schedule, Late Start Schedule, Constructive Heuristics (with different Priority Rules, Scheduling Schemes and Scheduling Directions) and Branch-and-bound (see references [1-6] for more details on these techniques).

3 MS Project Add-in

The MS Project Add-in was developed using the Microsoft Visual C Sharp (C#) programming language in the Integrated Development Environment Microsoft Visual Studio 2010 (VS2010) on the Windows 7 operating system.

Figure 1 shows the tab of the software application developed that appears within MS Project, with four buttons that allow the end user to choose the algorithm and perform the scheduling.

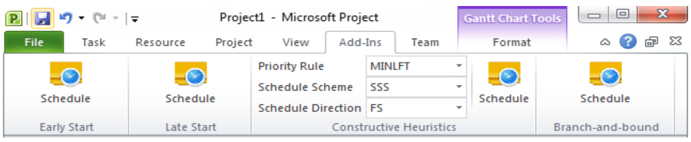

Figure 1 – Tab with the new MS Project Add-in

After the user presses the OK button, the algorithm will compute the start and end times of all activities, and will adjust the Gantt chart to represent these changes.

In addition to the treatment of the data from MS Project files, entered directly by the user, it is also possible to upload projects in the test environment from VS2010.

4 Results

To allow performing tests to the developed add-in, some project examples were created. The different methods studied were tested with all these projects. Due to the extension of the results obtained and to limitations of space, only one illustrative project will be presented in this paper.

The results presented are the Gantt charts obtained and the total project duration, for each scheduling technique implemented.

The Early Start Schedule was tested using some example projects and the schedules obtained were identical the ones generated by MS Project without resource constraints, so they are not presented in this paper.

The Late Start Schedule also gave the same results as MS Project, if the user selects to schedule activities as late as possible.

Figure 2 represents the example project network with ten activities and three resource types, which will serve to illustrate the application of the rest of the scheduling techniques. The numbers above the activities represent the activity duration and the numbers below the activities represent the number of resources needed.

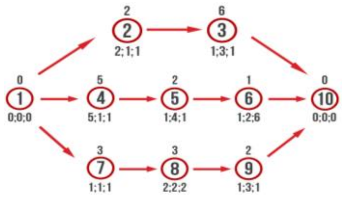

Figure 2 – AoN network for example project

The following resources availabilities were considered:

$$a_1 = 6, a_2 = 4 , a_3 = 8.$$

For the thirty two possible combinations to apply the *Constructive Heuristics*, a great variety of results were observed. Some shown a reduction on the total duration of the project, some gave similar results to those of MS Project (using manual resource leveling), and some shown worst results.

Figure 3 shows a typical result obtained using one of these scheduling techniques. Figure 4 shows the result obtained by MS Project, so a more objective comparison can be made.

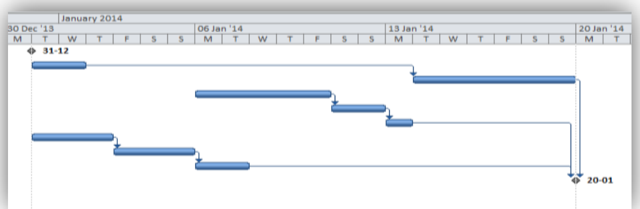

Figure 3 – Scheduling obtained using *Constructive Heuristics* (SPT – SSS – FS)

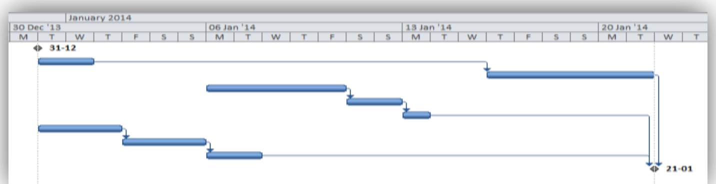

Figure 4 – Scheduling obtained using MS-Project with manual resource leveling

The result of the application of the *Branch-and-Bound* technique is presented in Figure 5. It can be seen that it represents a good alternative to the other schedules.

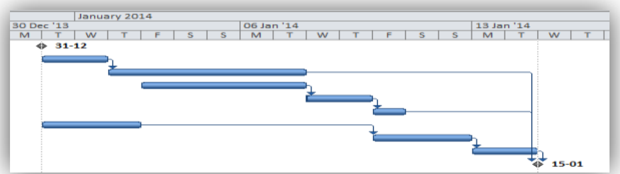

Figure 5 – Scheduling obtained using *Branch-and-Bound*

5 Conclusions

After performing all the tests, with various project examples, the conclusion was that *Branch-and-Bound* is the technique that achieves better results. For the example project, *Branch-and-Bound* obtained a schedule with duration equal to 15 days, while MS Project reached the minimum of 21 days.

In projects with one or two different types of resources, and few critical paths, the results do not show a great variation between methods. This is due to the low complexity of the project networks, which results in schedules that do not differ much. Results show significant differences for projects that have a large number of activities. For these projects, the schedule computed by MS Project is dominated by almost all the methods used.

The best results belong to the *Branch-and-Bound* method and *Constructive Heuristics*, composed by *Parallel Scheduling Scheme*, the priority rule *MINLFT* and the *Backward Direction*. The worst outcome is for the heuristic using the *Serial Scheduling Scheme*, the priority rule *GRD* and the *Forward Direction*.

As the *Branch-and-Bound* method is an implicit enumeration technique, all possible solutions to the problem were analyzed, and it was chosen the one having a shorter duration, since this is the variable to be minimized.

Thus, the conclusion was that the resource leveling method used by MS Project to schedule a project under resource constraints can be used for simple situations, where there are a small number of activities and few different types of resources. Even in these cases, its performance is poor and it is not automatic.

When it comes to larger and more complex networks, with regard to the variety of resources available, the technique used by MS Project is overtaken by the majority of the methods studied, mainly by *Branch-and-Bound* and the *Constructive Heuristics* using the *PSS*, *MINLFT* and *Backward Direction*. Another important outcome is the possibility made available to the user to select the scheduling method he wishes to use, allowing more control on the results obtained.

There are a number of other approaches to the RCPSP that should be interesting to address in the future, namely the use of meta-heuristics, like the pseudo particle swarm optimization, recently proposed to solve this problem [7].

Acknowledgements This work was supported by Portuguese Foundation for Science and Technology, under Project Pest-OE/EME/UI0252/2011.

References

1. Demeulemeester E, Herroelen W. Project Schedule - A Research Handbook: Kluwer Academic Publishers; 2002
2. Brucker P, Drexl A, Mohring R et al. Resource-constrained project scheduling: Notation, classification, models, and methods. European Journal of Operational Research 1999; 112: 3-41
3. Ulusoy G, Ozdamar L. Heuristic Performance and Network Resource Characteristics in Resource-Constrained Project Scheluling. Journal of the Operational Research Society 1989; 40: 1145-1152
4. Davis EW, Patterson JH. A Comparison of Heuristic and Optimum Solutions in Resource-Constrained Project Scheduling. Management Science 1975; 21: 944-955
5. Boctor FF. Heuristics for scheduling projects with resource restrictions and several resource-duration modes. International Journal of Production Research 1993; 31: 2547-2558
6. Brucker P, Drexl A, Möhring R et al. Resource-constrained project scheduling: Notation, classification, models, and methods. European Journal of Operational Research 1999; 112: 3-41
7. Nasiri MM. A pseudo particle swarm optimization for the RCPSP. International Journal of Advanced Manufacturing Technology 2013; 65: 909-918