Project resources leveling using software agents

Nivelarea resurselor în proiecte utilizând agenţi software

Professor Constanta-Nicoleta BODEA, Ph.D.
The Bucharest Academy of Economic Studies, Romania
e-mail: bodea@ase.ro
Mirela SABĂU, Ph.D. Candidate
The Bucharest Academy of Economic Studies, Romania

Abstract

Different approaches to project planning and scheduling have been developed. The Operational Research (OR) approach provides two major planning techniques: CPM and PERT. Artificial Intelligence (AI) initially promoted the automatic planner concept. In order to plan a project, the automatic application of predefined operators is required. However, most domains are not so easily formalized in the form of predefined planning operators. The paper focus is on the agent-based approach to project planning and scheduling, especially in Resource Leveling issues. The authors have developed and implemented the ResourceLeveler system, an agent-based model for leveling project resources. The objective of Resource Leveler is to find a scheduling of resources similar to the optimal theoretical solution which takes into consideration all constraints stemming from the relationships between projects, activity calendars, resource calendars, resource allotment to the activities and resource availability. ResourceLeveler was developed in C# as a plug-in for Microsoft Project.

Keywords: *project management, agent-based models, artificial intelligence, project resource leveling.*

Rezumat

Diferite abordări in planificarea proiectelor au fost definite in de domeniul cercetarilor operaţionale (OR), precum: CPM si PERT. Inteligenta Artificiala (AI) a promovat initial conceptual de palnificare automata. În aceasta abordare, pentru a planifica un proiect, este necesara definirea prealabila a unor operatori.In cele mai multe domenii, definirea acestor operatori ridica o serie de probleme. Noul model promovat de AI are la baza utilizarea agentilor software. Lucrarea se concentrează pe abordarea agent in problema de nivelare a resurselor. Autorii au dezvoltat şi implementat un sistem bazat pe agenţi. Obiectivul sistemului este de a identifica o solutie de programare a resurselor, care ia în considerare toate constrângerile rezultate din conditionarile dintre proiecte,

activitati calendare, calendare de resurse, alocarea de resurse pe activităţile şi disponibilitatea resurselor. Sistemul a fost realizat in C # ca un plug-in pentru Microsoft Project.

Cuvinte-cheie: *managementul proiectelor, agent software, inteligenta artificiala, nivelare resurse*

JEL Classification: O22, D44, D00

Introduction

Different approaches to project planning and project scheduling have been developed and presented in (Goldmann, 1996*),* (Henoch and Ulrich, 2001), (Petrie, Goldmann , and Raquet, 1999), (Russell and Norvig, 2003) . The Operational Research (OR) approach provides two major planning techniques: CPM and PERT. Artificial Intelligence (AI) initially promoted the *automatic planner concept* (Tate , 1977) and (Vere, 1983). In order to plan a project, the automatic application of predefined operators is required. However, most domains are not easily formalized in the form of predefined planning operators. The new AI approaches promote *model-based planning and scheduling*. An important class is that of agent-based models.

An agent is an entity that can perceive its environment through sensors and act upon that environment through effectors. The goal of AI is to design the agent program: a function that implements the agent mapping percepts to actions (Russell and Norvig, 2003). This program runs on some sort of computing device, called agent architecture. The Procura model was developed by S. Goldmann in cooperation with Stanford University (Goldmann, 1996*)*. Procura is an agent-based model which supports the planning, scheduling and execution of complex projects in an incremental and hierarchical approach. Procura uses and extends the Redux model (Petrie, Goldmann , and Raquet, 1999).

Definition of the resource leveling problem. IT toolsused in resource leveling

Starting from a well-defined resource collection allotted to a project, one can define Resource Leveling as the planning of the project's activities in a manner that respects all constraints resulting from activity dependencies and resource availability. It also minimizes the project duration. Resource Leveling implies finding the minimal solutions for the activity plan with consideration to the above mentioned constraints. We will see that there is no standard procedure in finding an optimal solution in the case of Resource Leveling. Even the recognition of a solution similar to the optimal one is problematic when dealing with complex projects that have complicated dependencies and allotments of multiple resources for their activities. A number of IT instruments have been developed to assist project managers. The best known tools are Microsoft Project and Primavera Project Planner. Considering the market percentage, Microsoft Project is the most popular project management software. It is useful and powerful in almost every aspect of project management.

The approach of ResourceLeveler

An solution of solving the problem of resource leveling is the ResourceLeveler model. The objective of the Resource Leveler plug-in is to find a resource scheduling similar to the optimal theoretical solution which takes into consideration all constraints stemming from the relationships between projects, activity calendars, resource calendars, resource allotment to the activities and resource availability and has the flexibility required by the agile environment. ResourceLeveler is based on a multi-agent system and an auction market. During the pre-leveling stage the statistic data of the project is computed (including the analysis of the critical path). Data collected in this stage will be used during leveling to compute the priority of each task. The leveling is realized by analyzing the work periods with a certain precision (hour or day) from the beginning of the project to its end. For each of these periods the program runs a negotiation round between the agents which represent the tasks in the frame of a virtual market that simulates a resource auction.

The market has the objective of deciding the winning offers and implicitly the activities which will be planned for the specified time span. Every offer received from the agents contains the desired resources and the required quantity as well as a price which characterizes the estimate value of the resources at the moment of auction for the agent.

The agents who represent the actions decide the leveling strategy because the price generated by the offers determines the task's importance in the present context. In order to set up a price, the agent uses a database that contains all considered elements. Some characteristics are common to all agents and represent proprieties of the project (for example the dependence graph between tasks), while other characteristics are specific to the represented activity. In the following we will present the main components of ResourceLeveler.

The Auction Market

On this market resources are exchanged. The resources are sold by the auction judge (in this case the market) and bought by agents who represent the activities of the project. In case of an over-allotment these auctions are held with a deficit of resources. In this case the winning offers are the ones which have offered the best price. These winning offers are bound to an activity which will be planned for implementation in the current day of the project execution. An important characteristic of this market is the way in which the auction is held. The implementation of a first-price auction with sealed offers has been chosen because the goal of the bid is not to encourage a competition between the participating agents but to create a hierarchy of the theoretic values of the represented activities. An important factor was the fact that such an auction takes place rapidly because it consists of only one bidding round and no negotiations. The bidding market is responsible for the coordination of the auction with sealed offers. The bidding market plays the role of the auction judge, deciding the winning offers.

The difference between the implementation used by ResourceLeveler and the classical implementation of bidding with sealed offers is the way in which goods are sold. Classically, the goods are sold one by one, every agent wishing to participate having to make an offer for the auctioned resource. Despite this, the particularities of allotting a resource for the tasks have led to an extended version of this type of auction. All auctioned goods are presented before the bidding begins and the involved agents make a single offer

for all goods the whish to obtain. In this way, one has realized a natural and efficient model for the allotment of multiple resources in the same activity. By providing a single price for all auctioned goods, the agent's offers raise further problems regarding the choice of a winner because one has to find the combination of offers that maximize the market's profit.

Figure 1 presents the market structure used by ResourceLeveler. The main steps of the auction are:

1. The first agent reads the total of available resources.
2. The first agent generates a proposal to the market.
3. The second agent reads the total of available resources.
4. The second agent generates a proposal to the market.
5. After all offers have been received, they are ordered in according to the price offered.
6. In decreasing order of the price, the necessary resources are verified and compared to the available resources. If all resources are available, the offer is accepted and the resources consumed. The next offers will be verified according to the new resource availability. The process continues until all offers are analyzed.

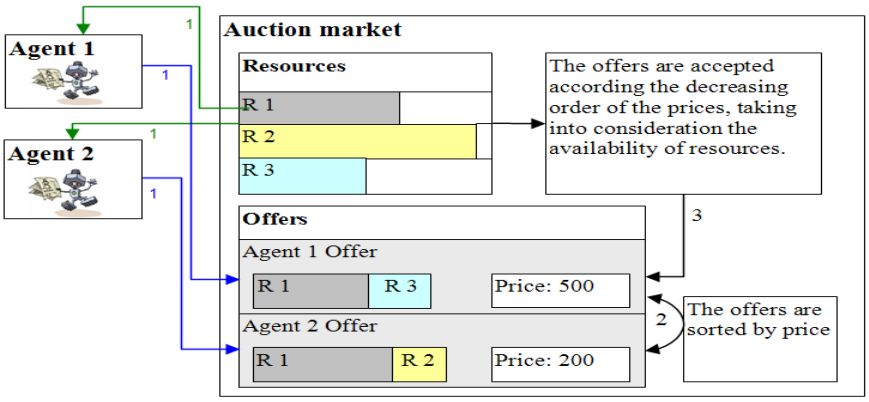

Figure 1 The structure of the auction market with sealed offers

Bidding Agents

These are the main entities of the resource leveler and have a strong impact on its behavior. By changing the types of agents used one can completely change the program's behavior. This is why it is important that these entities be carefully designed. The agents represent the component activities of the project and their interest is to gain the necessary resources for the execution of the represented activities. If an agent makes an offer and wins the resource bid, the represented activity can be executed on the same day. From case to case the starting date of the task will be modified or a new section for the planning of the activity (split) will be created. Because a system of sealed offers is used, the bidding agents use the estimated value of the resources as price. The value estimation of the necessary resources represents the agent's logic and determines his behavior. The described model supports any implementation of the agents and even a number of different implementations of the agents. The implemented agent's complexity varies from ordinary agents of level 0,

who have no own models, to level 2 agents who model both the system as well as the other competitor agents. For the implementation ordinary level 0 agents were chosen, who found their reasoning on small heuristic algorithms analyzing the data of different tasks, the data extracted from the critical path analyze and the data referring to resource allotments. Three types of ordinary agents have been implemented, each of them being the representation of a specific resource leveling strategy:

a) Agent Based on the Duration of the Activities Following the Represented Task;

b) Agent Based on the Time Float of the Represented Activity;

c) Agent Based on the Number of Allotted Resources and on the Time Float of the Represented Activity.

The ResourceLeveler system: structure and implementation

ResourceLeveler system was developed in C#, considering the Microsoft Project plug-in support which is dedicated to the programmers using .NET technologies. The system has the following functional modules, which communicate through the interfaces (figure 2):

- *Interface module*. This module is responsible for the insertion of the ResourceLeveler button into the Microsoft menu and the communication with the user;
- *Wrapper module.* This module extracts the project data offered by Microsoft Project;
- *Leveling module.* This is an intermediary module which adapts the negotiation algorithm based on bid to the leveling process
- *The auction market simulation module*.
-

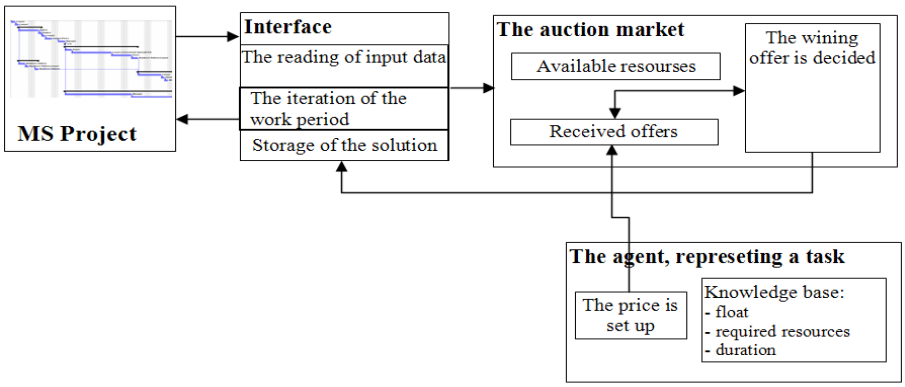

Figure 2 The structure of ResourceLeveler

Conclusions. Future research

From the point of view of managers, a good resource leveling tool ensures the minimal duration of a project taking the available resources into consideration. This is because a project finished early saves costs. In spite of this, managers hesitate to over-allot

resources in order to speed up a project. This reasoning is based on two factors: the human factor and the financial factor. The latter takes into account the rise in costs because of the over-allotment taxes, and the human factor deals with the unwanted collateral effects of using a human resource over its normal work capacity. In future research we intend to extend the types of agents acting on the auction market in order to increase the system's flexibility. We intend to develop agents that use an *iterative estimation* of the activities' duration and time float. We will compare current results with the ones obtained through their implementation in ResourceLeveler. Through this comparative analysis we will develop an agility indicator for software agents.

References

Carayannis E. G., Kwak Y. H. (2002). *The story of managing projects: a global, cross-disciplinary collection of perspectives*, Greenwood Press, Quorum Books, 2002.

Goldmann S (1996*). Procura: A project management model of concurrent planning and design*. Proceedings of WETICE-96, Stanford, CA, 1996, http://www-cdr.stanford.edu/ ProcessLink/procura/papers/procura.html

Henoch J., Ulrich H. (2001). *Agent-based simulation platform for evaluating management concept*, Proceedings of EUROSIM Spanning Future with Simulation, Delft, The Netherlands, ISBN: 90-806441-1-0.

Petrie C., Goldmann S., Raquet A. (1999). *Agent-based project management*. Lecture Notes in AI – 1600, Springer-Verlag, http://www-cdr.stanford.edu/ProcessLink/papers/ DPM/dpm.html

Russell S.J., Norvig P.(2003). *Artificial intelligence, A modern approach*, Prentice Hall, Inc. 2003, 31-52.

Tate A.(1977). *Generating project networks*, IJCAI, 1977, Boston, MA, USA.

Vere S.A. (1983). *Planning in time: windows and durations for activities and goals*, IEEE Trans.