project, project scheduling, resource-constrained project scheduling, project-driven manufacturing, multi-mode, heuristic, branch and bound scheme, make-to-order

Iwona PISZ*
Zbigniew BANASZAK**

APPLIED A NEW METHOD FOR MULTI-MODE PROJECT SCHEDULING

Abstract

The aim of this paper is to present a modelling heuristic framework that enables one to cope with a problem of a project-driven manufacturing. The objective is to find computationally effective method aimed at scheduling of a new project subject to constraints imposed by a multi-project environment. The application of a heuristic method of scheduling is demonstrated on one example of a makespan-feasible schedule that follows the constraints imposed by the precedence relation and by the time-constrained resources availability. This heuristic method is based on concept of critical path and branch and bound scheme.

1. INTRODUCTION

Most companies, particularly small and medium size enterprises have to manage various projects, which share a pool of constrained resources, taking into account various objectives at the same time. Not only one but several, even dozens or hundreds of projects are typically going on at the same time within an enterprise [7]. A project is defined as a temporary endeavor undertaken to create a unique product or service [9]. In order to decide whether a new project can be executed in a given production system the producer capabilities and the customer needs have to be taken into account. The issue considered in the paper belongs to resource-constrained project scheduling problem. The problem is important for make-to-order companies where the products are manufactured based on make-to-order principle.

Enterprises have to manage various projects at the same time. According to the surveys conducted about 84% of firms have to deal with multiple projects, which share a pool of constrained resources, taking into account various objectives simultaneosly. Other results indicate predominance of projects with less than 50 activities (84%), while about 95% of projects have

* Dr inż. Iwona Pisz, Technical University of Opole, Faculty of Management and Engineering, Ozimska 75, 45-370 Opole, Poland, e-mail: i.pisz@po.opole.pl

** Prof. Zbigniew Banaszak, Technical University of Koszalin, Department of Computer Science and Management, Śniadeckich 2, 75-453 Koszalin, Poland, e-mail: banaszak@tu.koszalin.pl

less than 100 activities [6]. The availability of the resources assigned to a project is limited and often not efficient to execute the activities of the project.

The resource-constrained project scheduling problem has been considered through two approaches: the scheduling of a single project and the scheduling of multiple simultaneous projects. Most of the publications on project scheduling have been dedicated to single project. In recent years there is a growing interest in problems related to project scheduling in multi-project environments. In the single project case, the time objectives are one of the most dominant problem, for example minimising project duration (see, e.g. [1, 4, 8]). In contrast, scheduling of several projects with common and constrained resources takes into account other criteria such as: idle resources, resource levelling, in-proces inventory, project splitting [6]. Many articles focused on the development of scheduling for static environments [3, 5. 12]. There are other studies devoted to problem of scheduling [10, 11, 13].

This paper addresses an issue of decision-making support for small and medium size enterprises. The objective is to find a computationally effective method of scheduling in an enterprise. It aims to find a feasible schedule that follows the constraints imposed by duration order and price given by customer and by the time-constrained resources availability. In other words it is looking for the answer whether a given work order can be accepted for processing in a given enterprise. The problem belongs to a class of multi-mode case problems of a project scheduling, where finding of a feasible solution is NP-complete [2, 11]. Because of real-life constraints such as requirements imposing on-line decision making one may consider usaging of a branch and bound scheme in the course either of exhaustive or selected (heuristic dependent) search. Only small-sized problem instances with up to 60 activities can be solved exactly in a satisfactory manner. Therefore, heuristic solution procedures remain as the only feasible method of handling practical scheduling problem [11].

The rest of the paper is organized as follows: in the next section the main problem is formulated. A concept standing behind the heuristic method a feasible project schedule is presented in Section 3. In Section 4 one illustrative example of the method usage is provided. Results and some concluding remarks are presented in Section 5.

2. PROBLEM STATEMENT

The following problem can be now formulated. Consider a manufacturing system providing a given production capability while processing some other work orders. That means that only a part of the production capability (specified by in the time-restricted resources availability) is available for use in the system. A given work order is represented by an activity-on-node (AON) network where the nodes and the arcs represent the activities and precedence relations, respectively.

The project is specified by project duration deadline, which is equivalent to a presumed completion time (the work order cycle) as well as a total project cost constraint (selling price). Each activity may be executed in one out of the set of modes (system resources). Also, each activity may not be pre-empted and the mode once selected may not be changed. Considering a time horizon, that is an upper bound on the project's makespan, there is only a certain number of units of available renewable resource in the considered period. The cost of using the unit of the resource is specified in the map of accessible of renewable resources.

The problem considered here is related to finding a makespan – a feasible schedule that satisfies the constraints imposed by the precedence relations and by the time-constrained resources availability. The objective is to find an answer for the following question: whether the company production capacity is sufficient for the execution of a project in accordance with customer's requirements, and especially, given the planned project execution deadline and cost whether there exists a feasible schedule.

3. THE HEURISTIC METHOD

In order to cope with this problem, let us consider a heuristic method, which is based on the branch and bound scheme. The searching procedure is driven by an upper bound evaluation policy.

3. 1. The searching strategies

In order to avoid costly exhaustive enumeration of possible schedules the cases explored are limited first of all to the ones possessing the lowest margins of cost and time. In other words, those cases could lead to an unfeasible schedule. Of course, the proposed way the cases are explored can be treated as searching with an assumption that a feasible schedule does not exist.

A difference between assumed project duration deadline and a project makespan obtained in the case of absence of resource time-constraints is applied as an upper bound evaluation. It means that at the beginning a difference between assumed project duration deadline and a project makespan obtained (i.e., corresponding to a critical path) in the case of absence of resource time-constraints is calculated. The same regards cost evaluation (i.e., cost of resources occurring along the critical path). In the case when a cost margin (i.e., the difference of costs) or time margin is less than zero, a feasible schedule does not exist, or else the makespan taking into account availability of time-restricted resources is calculated. For such a newly obtained critical path, the cost and time margins are once more calculated. In the case when one of margins is less than zero the feasible schedule does not exist, or else the searching process is continued.

In order to continue the searching process, a modified project network and a modified resource availability constraints have to be considered as new data. Removing activities from the project network assigned to the critical path one may consider a set of subnetworks. Each subnetwork, in turn, has its own duration time deadline following from the former makespan (see the moments corresponding to the fork and/or joint type nodes of the critical path in the project network). In turn, after removing resources assigned to the critical path the modified resources constraints have to be considered as well.

Following the above considerations an ordered set of subnetworks $\{AN_1, AN_2, AN3,\ldots,AN_n\}$ can be observed and the following conditions can be supposed to be the sufficient ones for a feasible project schedule.

$$AN_i \prec AN_j \;\Leftrightarrow\; UB_i \le UB_j, \quad UB_i = TL_i - CP_i \quad \text{and} \quad UB \ge 0 \tag{1}$$

$$\forall i \in \{2,\ldots,n-1\}\ CP_i(RC_{i-1}) \le M_j(CP_{i-1}), \text{ and } CP_1(RC_0) \le M1(CP_0) = T_h \text{ and } M_j(CP_{i-1}) \ge 0 \tag{2}$$

$$\forall i \in \{1,2,\ldots,n-1\}\ M_j = M_{j-1} - CP_j \le 0 \ \text{ and } M_0 = AC \quad \text{and} \quad M_j \ge 0 \tag{3}$$

where: AN_i – the i-th activity network,
UB_i – the upper bound of the AN_i,
AC – the assumed project cost limit,

TL_i – the time period limiting duration of the AN_i subnetwork,
CP_i – the makespan of the AN_i subnetwork (the time constraints imposed on resources are not taken into account),
$RC_i = f(RC_{i-1}, CP_{i-1})$ – the i-th actualisation of the resource time-availability constrains, i.e., the actualisation of the (i-1-th resource constraints following exclusion of resources associated to the (i-1)-th critical path,
$CP_i(RC_{i-1})$ – the makespan of the AN_i subnetwork taking into account the time constraints RC_{i-1} imposed on resources,
$M_i(CP_{i-1}) = TL_i – CP_i(RC_{i-1})$ – the time margin of the AN_i taking into account the actualisation of the (i-1)-th resource constraints following exclusion of resources associated to the (i-1)-th critical path,
M_i – the cost margin after taking into account the cost CP_j, i.e., the cost of utilization of the resources associated to the critical path of the AN_i,
T_h – the assumed project duration (and/or joint type nodes of the critical path in the project network).
In turn, removing resources assigned to the critical path the modified resources constraints have to be considered as well. Therefore, for each subnetwork the corresponding upper bound can be calculated. Finding the subnetwork with the lowest value of the upper bound allows to repeat the main procedure, i.e. to calculate the cost and time margins, and then to consider the new subnetworks. It means that from the extended set of subnetworks one has to find the element distinguished by a smallest value of the upper bound. Then calculate the margins, and so on. The procedure ends either in the case when one of margins is less than zero or the set of subnetworks is exhausted.

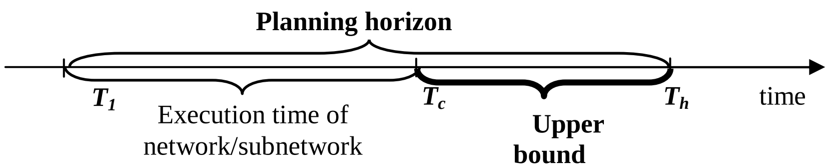


Fig. 1. Estimation value of the upper bound

The heuristic applied focuses on the searching process on the upper bound value and results in an order along to which the resource constraints are then modified. Of course, the resource constraints modification can influence already calculated value of upper bounds.

The heuristic rule applied can be treated as a set of sufficient conditions. In the case if they hold for the given project and manufacturing system specifications, then there exists a feasible project schedule. However a feasible solution may exists also in the cases the sufficient conditions are not satisfied. This obvious disadvantage, diminishes the computational efficiency of the procedure provided. The negative result simply means that a feasible schedule cannot be considered due to the conditions build-in the searching procedure. In other words it means there is no guarantee the feasible schedule does not exist. Such an evaluation could be enhanced in the case when the set of sufficient conditions is extended. In order to illustrate a way the new heuristics could be considered (i.e. the new conditions could be added), let us assume some modification of the previously introduced conditions (1) – (3). Let us replace the former condition (1) by the following one:

$$AN_i \prec AN_j \;\Leftrightarrow\; UB_i \geq UB_j;\ UB_i = TL_i - CP_i \;\text{ and }\; UB \geq 0 \tag{4}$$

The conditions encompassing the cost and time constraints together with resources cost and availability constraints provide a natural framework for implementation of the constraint logic programming methods.

3. 2. The resources allocation heuristics

The proposed method includes four resources allocation heuristics. The heuristics are based on estimation of the value of resource time availability and the average of the resource cost usage in the given period. The heuristics proposed are following:

- the smallest resource time availability,
- the greatest resource time availability,
- the lowest average resource cost,
- the highest average resource cost.

The two first heuristics base on estimating the value of the resource availability in the given time horizon (the subnetwork duration). From the set of resource alternatives the one characterized by the smallest/greatest time availability is selected and then is assigned to the given activity. The value of the resource time availability is computed as follows:

$$d_i = \sum_{i=g}^{DD} y(r_i) \quad , d_i \in \{0,1\} \qquad (5)$$

where:
d_i – the i-th the resource capability,
g – the beginning of planning horizon,
DD – the end of planning horizon, $y(r_i)$ – binary variable which determine the availability the r-th resource in the i-th unit time. d_i = 0 means that the cost of resource utilization equals to zero (the given resources is busy). d_i = 0 means that the resource cost is bigger than zero.

According to third and fourth heuristics the priority is given to the resource characterized by the lowest/highest cost in the given planning horizon. The cost value is estimated as follows:

$$\overline{k_i} = \frac{1}{d} \sum_{i=g}^{DD} k_i \qquad (6)$$

where:
$\overline{k_i}$ – the average of the i-th resource cost of usage,
k_i – the resource cost of utilization, d – time of the resource availability in i-th unit time in a given time period. This variable denotes the sum of resource capability (the cost of resource utilization is bigger than zero) in duration time deadline for the given subnetwork.

4. ILLUSTRATIVE EXAMPLES

For illustration purposes let us consider the project specified by the activity network shown in Fig. 2.

Let the project duration deadline equals T_h = 31 units of time and the project total cost is equal K_h = 200 cost units.

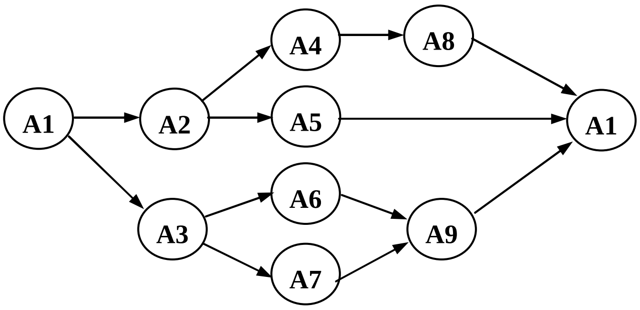


Fig. 2. The network of the project

Assume the variant of activities allocation as shown in Table 1. The amounts in the table specify the time of activity processed with help of the assigned resource. There are no altternative resources. It means that only one resource is assigned with one given activitie. In this situation the resources allocation heuristics are not used.

Tab. 1. Avariant of project network activities allocation

	A1	**A2**	**A3**	**A4**	**A5**	**A6**	**A7**	**A8**	**A9**	**A10**
R1								2		
R2	4									
R3			3							
R4		1							1	
R5					6		3			
R6				7		7				4

The Figure 3 specifies the resources time-constrained availability. Not blanked cells of the table specify the cost assigned with resources usage in the units of cost. The cost equals zero means that the given resource is busy in the moment. Due to the Critical Path Method the critical path consists of the following activities: A1 – A3 – A6 – A9 – A10 and corresponds to the following production routing: R2 – R3 – R6 –R4 – R6. The minimum completion time of the project is equal to 19 units of time.

R1	0	0	3	3	3	3	3	3	3	3	0	0	0	0	0	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2	7	7	7	7	7	7	6	6	6	6	6	6	6	6	8	8	8	8	8	0	0	0	0	0	0	0	0	0	0	0	0
R3	0	0	0	0	0	0	0	0	0	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4	0	5	5	5	5	5	5	5	5	0	0	0	0	0	0	0	0	5	5	5	5	5	5	5	5	5	0	0	0	0	0
R5	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0
R6	0	0	0	0	3	3	3	4	4	4	4	4	4	4	4	0	0	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**	**31**
	Planning horizon																														

Fig. 3. Resources avaiability constraints

Taking into account the resources availability constraints the project makespan equals to 29 units of time (see Fig. 4). The cost associated is equal to 115 units of cost. So, the both: time (i.e. 31- 29 ≥ 0) and cost (i.e., 200 – 115 ≥ 0) margins allow one to continue the searching process.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R1	0	0	3	3	3	3	3	3	3	3	0	0	0	0	0	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2	7	7	7	7	7	7	6	6	6	6	6	6	6	6	8	8	8	8	8	0	0	0	0	0	0	0	0	0	0	0	0
R3	0	0	0	0	0	0	0	0	0	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4	0	5	5	5	5	5	5	5	5	0	0	0	0	0	0	0	5	5	5	5	5	5	5	5	5	5	0	0	0	0	0
R5	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0
R6	0	0	0	0	3	3	3	4	4	4	4	4	4	4	4	0	0	5	5	5	5	5	5	5	5	5	5	5	5	5	5

Planning horizon

Fig. 4. The Gantt chart of critical path taking

Let us remove from the project network the activities belonging to the critical path. As result consider a set of subnetworks (see Fig. 5) and the new resources availability constraints (see Fig. 6). The upper bounds of distinguished subnetworks are as follows: 11 units of time for the subnetwork AN_1, and 9 units of time for the subnetwork AN_2. Note that due to the critical path from the Fig. 4 the duration deadlines for subnetworks AN_1, AN_2 are 21 (since 5 till 25), and 12 (since 13 till 24) units of time, respectively.

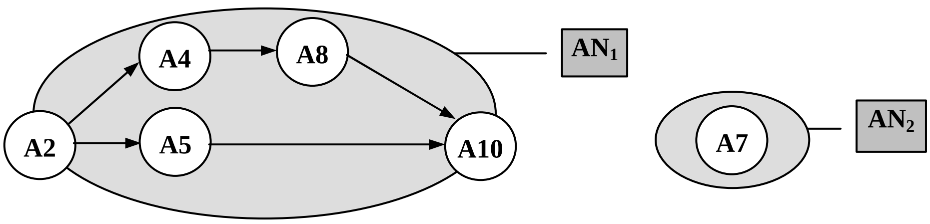


Fig. 5. The subnetworks of the project

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R1	0	0	3	3	3	3	3	3	3	3	0	0	0	0	0	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2	**0**	**0**	**0**	**0**	7	7	6	6	6	6	6	6	6	6	8	8	8	8	8	0	0	0	0	0	0	0	0	0	0	0	0
R3	0	0	0	0	0	0	0	0	0	**0**	**0**	**0**	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4	0	5	5	5	5	5	5	5	5	0	0	0	0	0	0	0	5	5	5	5	5	5	5	5	**0**	5	0	0	0	0	0
R5	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0
R6	0	0	0	0	3	3	3	4	4	4	4	4	4	4	4	0	0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	5	**0**	**0**	**0**	**0**	5	5

Planning horizon

Fig. 6. The Gantt chart of subnetwork AN_2

Since the lowest value of upper bound corresponds to the subnetwork AN_2, its critical path has to be determined as first (see Fig. 6). The relevant time and cost margins are 9 units of time and 73 = 200 – 127 units of cost, respectively. The subnetwork AN_2 contains of a single path.

Therefore, the searching process moves on the subnetwork AN_1. The critical path of the subnetwork AN_1 consists of the following activities A2 – A4 – A8 and corresponds to the following production routing R4 – R6 – R1. The corresponding production flow is shown in Gantt's chart in Fig. 7. Because the time and cost margins holds (i.e. the time and cost margins are equal to 8 units of time and 36 units of cost, respectively) the activities of the critical path have to be removed from the subnetwork AN_1.

R1	0	0	3	3	3	3	3	3	3	3	0	0	0	0	0	3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2	0	0	0	0	7	7	6	6	6	6	6	6	6	6	8	8	8	8	0	0	0	0	0	0	0	0	0	0	0	0	0
R3	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4	0	5	5	5	5	5	5	5	5	0	0	0	0	0	0	0	5	5	5	5	5	5	5	5	0	5	0	0	0	0	0
R5	0	0	0	0	0	0	4	4	4	4	4	4	0	0	0	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0
R6	0	0	0	0	3	3	3	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0	0	0	5	0	0	0	0	5	5
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Planning horizon																														

Fig. 7. The Gantt chart of a critical path of the subnetwork AN_2

Removing the activities of the above considered critical path from the subnetwork AN_1 results in the single path containing the activity A5 which has to be executed in the period [6; 25], i.e. within 20 units of time. The margins of time and cost corresponding to the relevant critical path (see Fig. 8) are equal to 13 units of time and 12 = 200 – 188 units of cost, respectively. The obtained feasible schedule of the project is shown in Fig. 9.

R1	0	0	3	3	3	3	3	3	3	3	0	0	0	0	0	0	0	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2	0	0	0	0	7	7	6	6	6	6	6	6	6	6	8	8	8	8	0	0	0	0	0	0	0	0	0	0	0	0	0
R3	0	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4	0	5	5	5	0	5	5	5	5	0	0	0	0	0	0	0	5	5	5	5	5	5	5	5	0	5	0	0	0	0	0
R5	0	0	0	0	0	0	4	4	4	4	4	4	0	0	0	4	4	4	4	4	4	4	4	4	0	0	0	0	0	0	0
R6	0	0	0	0	3	0	0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	0	0	0	5	0	0	0	0	5	5
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	Planning horizon																														

Fig. 8. The Gantt chart of a critical path of the subnetwork AN1 consisiting of activity A5

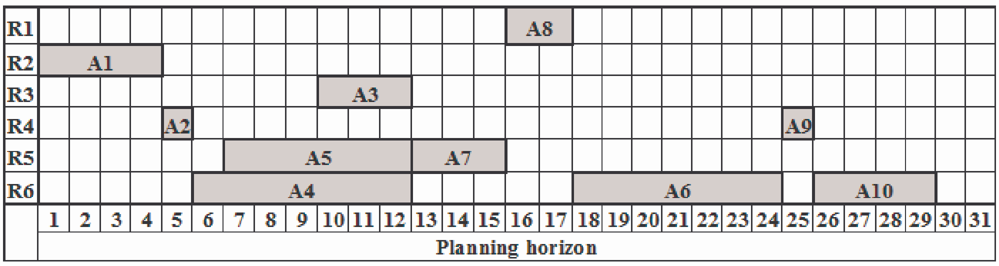


Fig. 9. The feasible schedule of the project

The searching procedure may be thought as an upper bound driven one, i.e., focused on a smallest difference between the DD_i and a makespan of $AN_{j,k,..,I}$ (a makespan determined under assumption the all resources are available at the start of the project) among currently available subnetworks. In the case considered, the searching order has been determined by the following sequence of vertices: 1 – 1,1 – 1,2 – 1,2,1 that corresponds to the following sequence of differences: 12 = DD – 19, 9 = DD_1 – 3, 11 = DD_2 – 9, 19 = DD_3 – 6, where DD = 31, DD_1 = 12 , DD_2 = 21, DD_3 = 25 (Fig. 10).

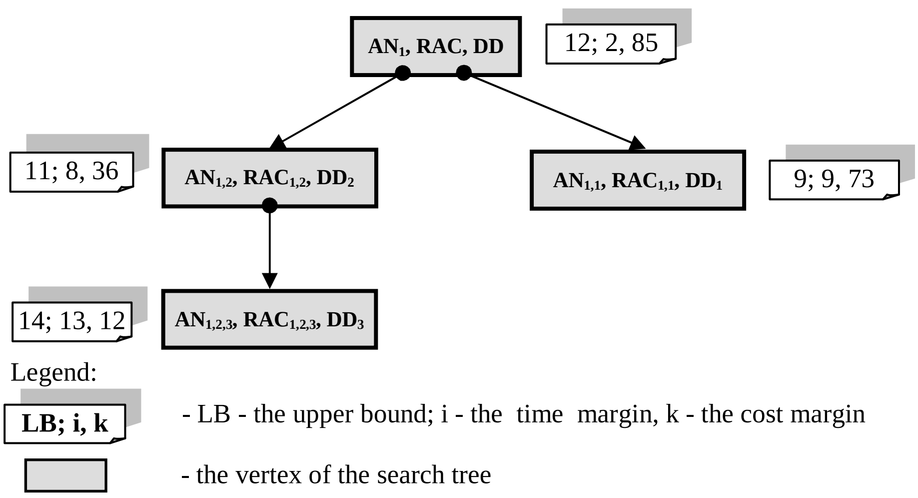


Fig. 10. The tree of the feasible schedule search

5. CONCLUSION

Project scheduling is important for make-to-order enterprises where the capacities have been cut down in order to cope with lean management concepts. This paper addresses the issue of decision-making support for small and medium size enterprises. The objective is to find a computationally effective method of project scheduling in an enterprise. The considered problem concerns finding a feasible schedule work order that follows the constraints imposed by order duration and a price given by a customer and by the time-constrained resources availability. The problem belongs to a class of multi-mode case problems of project scheduling. This paper shows the method of balancing the projects tasks and manufacturing system capability. A new heuristic method has been proposed based on the critical path method and the branch and bound scheme.

A modeling framework supporting decision making systems design, which in turn are aimed at finding the answer whether a given project can be accepted for processing in an enterprise assumed is considered. It provides a good platform for consistency checking between the work order completion requirements and a workshop capability offered.

The approach proposed seems to be useful for the project-driven production flow management applied in a kind of make-to-order companies, especially in small and medium size companies.

References

[1] BOCKOR F.F.: *Some efficient multi-heuristic procedures for resource constrained project scheduling*, European Journal of Operational Research nr 49/1990, s. 3-13

[2] BRUCKER O., DREXL A., MŐHRING R., NEUMANN K., PESCH E.: *Resource-constrained project scheduling: Notation, classification, models, and methods*, European Journal of Operational Research nr 112/1999, s. 3–41

[3] DUMOND J., MABERT V. A.: *Evaluation project scheduling and due date assigment procedures: An experimental anaysis*, Management Science nr 34/1988, s. 101-118

[4] KOLISCH R., HARTMANN S.: *Experimental investidation of heuristics for resource-constrainted project scheduling: An update*, European Journal of Operational Research nr 174/2006, s. 23-37

[5] KURTULUS I. S., DAVIS E. W.: *Multi-project scheduling: Categorization of heuristic rules performance*, Management Science nr 28/1982, s. 161-172

[6] LOVA A., MAROTO C., TORMOS P.: *A multicriteria heuristic method to improve resource allocation in multiproject scheduling*, European Journal of Operational Research, nr 127/2000, s. 408–424

[7] MARTINSUO M., LEHTONEN P.: *Role of single-project management in achieving porttfolio management efficiency*, International Juornal of Project Management, nr 25/2007, s. 56-65

[8] ŐZDAMAR C., ULUSOY G.: *A note on an iterative forward/backward scheduling technique with reference to a procedure by Li and Wilis*, European Journal of Operational Research, nr 89/1996, s. 400-407

[9] PMI STANDARDS COMMITTEE: *A Guide to the Project Management Body of Knowledge*, PROJECT MANAGEMENT INSTITUTE, Newtown Square, PA, 2004

[10] SPERANZA M. G., VERCELLIS C.: *Hierarchical models for multi project planning and scheduling*, European Journal of Operational Research nr 64/1997, s. 312-325

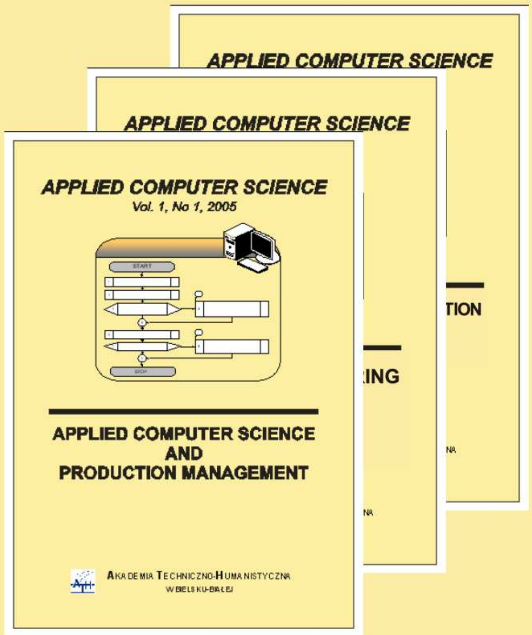
[11] VALLS V., BALLESTIN F., QUINTANILLA S.: *A hybrid genetic algoritm for the resource-constrained project scheduling problem*, European Journal of Operational Research, nr 185/2008, s. 495-508.

[12] WILEY V. D., DECKRO R. F., JACKSON J. A. JR.: *Optimization analysis for design and planning of muti-project programs*, European Journal of Operational Research nr 107/1998, s. 492-506

[13] YANG K. K. SUM C.C.: *An evaluation od due date, resource aloacation, project release, and activity scheduling rules in a multi-project environment*, European Journal of Operational Research, nr 103/1997, s. 139-154

APPLIED COMPUTER SCIENCE

- Management
- Designing
- Manufacturing
- Organization
- Innovation
- Competitiveness
- Quality and Costs



This issue is published by:

Lublin University of Technology, Institute of Technological Systems of Information, 20-618 Lublin, Nadbystrzycka 36, Poland

Slovak Productivity Center, Univerzitná 1, 010 08 Žilina, Slovakia

University of Bielsko-Biała, Department of Industrial Engineering, 43-309 Bielsko Biała, Willowa 2, Poland

University of Economics, 130 67 Prague, Winston Churchill Sq.4, Czech Republic

Technical University of Koszalin, Department of Computer Science and Management, 75-453 Koszalin, Śniadeckich 2, Poland

University of Žilina, Mechanical Faculty, Department of Industrial Engineering, Univerzitná 1, 010 26 Žilina, Slovakia

Editorial Office:
INSTITUTE OF TECHNOLOGICAL SYSTEMS OF INFORMATION
Mechanical Faculty, Lublin University of Technology,