Prof. Zbigniew Banaszak
University of Zielona Góra
Institute of Computer Science and Production Management
M. Sc. Iwona Pisz
Technical University of Opole
Department of Management and Production Engineering

PROJECT-DRIVEN PRODUCTION FLOW MANAGEMENT

The aim of this paper is to present a modelling 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. Concluding results are summarized on example of a makespan-feasible schedule that follows the constraints imposed by the precedence relation and by the time-constrained resources availability.

1. INTRODUCTION

Searching for the optimal solutions, regarding for example resources allocation, time lags, makespan, costs, and so on, has to be preceded by formulation of a feasibility problem or equivalently speaking by a constraint satisfaction problem whereby the objective function is included into the set of constraints. Solution to this problem permits a user to investigate the effect of a new work order impact on a performance of a manufacturing system. In other words, enables finding an answer to the most important question whether a given work order (i.e. a new project) can be accepted to be processed in a manufacturing system on hand (e.g. multi-project environment), i.e., whether its completion time, batch size, and its delivery period satisfy the customer requirements while satisfying constraints imposed by the enterprise configuration and the process of manufacturing of other products [1, 2, 3].

A model considered consists both a specification of a new project (including project network, and project duration and project cost constraints) as well as a manufacturing system specification (including cost of resources and time-restricted resources availability). The modelling framework provides a platform for a feasibility problem formulation as well as for a branch and bound-like procedure applied in order to solve this problem.

The rest of the paper is organized as follows: Section 2 describes the modelling framework enabling to state a problem. A concept standing behind of the method aimed at searching of a feasible project schedule is then presented in Section 3. In Section 4 an illustrative example of the method usage is provided. Some conclusions are presented in Section 5.

2. STATEMENT OF THE PROBLEM

The approach considered can be applied to the production flows observed in virtual enterprises [4, 8] as well as to the project-driven small and medium size enterprises. In the last case we have to consider, however, the following problem statement.

Consider a manufacturing system providing a given production capability while processing some other work orders. So, only a part of the production capability (specified by in the time-restricted resources availability) is available for use in the system. Given works order is represented by an activity-on-node network, and 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. Each activity may be executed in one out of the set of $M_{(i,j)}$ modes (system resources). Also, each activity may not be preempted and the mode once selected may not be changed.

The problem considered regards of finding of a makespan-feasible schedule that follows the constraints imposed by the precedence relations and by the time-constrained resources availability.

3. PROJECT SCHEDULING IN CONCURRENT MULTIPROJECT ENVIRONMENT

The problem considered belongs to a class of multi-mode case problems of a project scheduling, where finding of a feasible solution is NP-complete [6, 7]. In order to cope with the problem one may consider usage of a branch and bound scheme.

Given a project specified by an activity network as well as by its duration deadline and total cost constraints. Consider manufacturing system specified by resources cost and time-restricted resources availability. The question we are facing with regards of a feasibility of the project schedule, i.e. the question: Whether there exists a project schedule following project's duration deadline and cost limits or not?

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 first of all the cases those could lead to an unfeasible schedule are considered. Of course, the proposed way the cases are explored can be treated as searching aimed at proving 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-constrains is applied as a lower bound evaluation. It means 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-constrains is calculated. The same calculation regards of cost evaluation (i.e., cost of resources occurring along the critical path). In the case a time margin (i.e., the difference of costs or time margin is less than zero a feasible schedule does not exists, else the makespan taking into account availability of time-constrained resources is calculated. For such a newly obtained critical path the cost and time margins are once more calculated. In the case one of margins is less than zero the feasible schedule does not exist, 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 a new data. Removing from the project network activities 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, removing resources assigned to the critical path the modified resources constraints have to be considered as well.

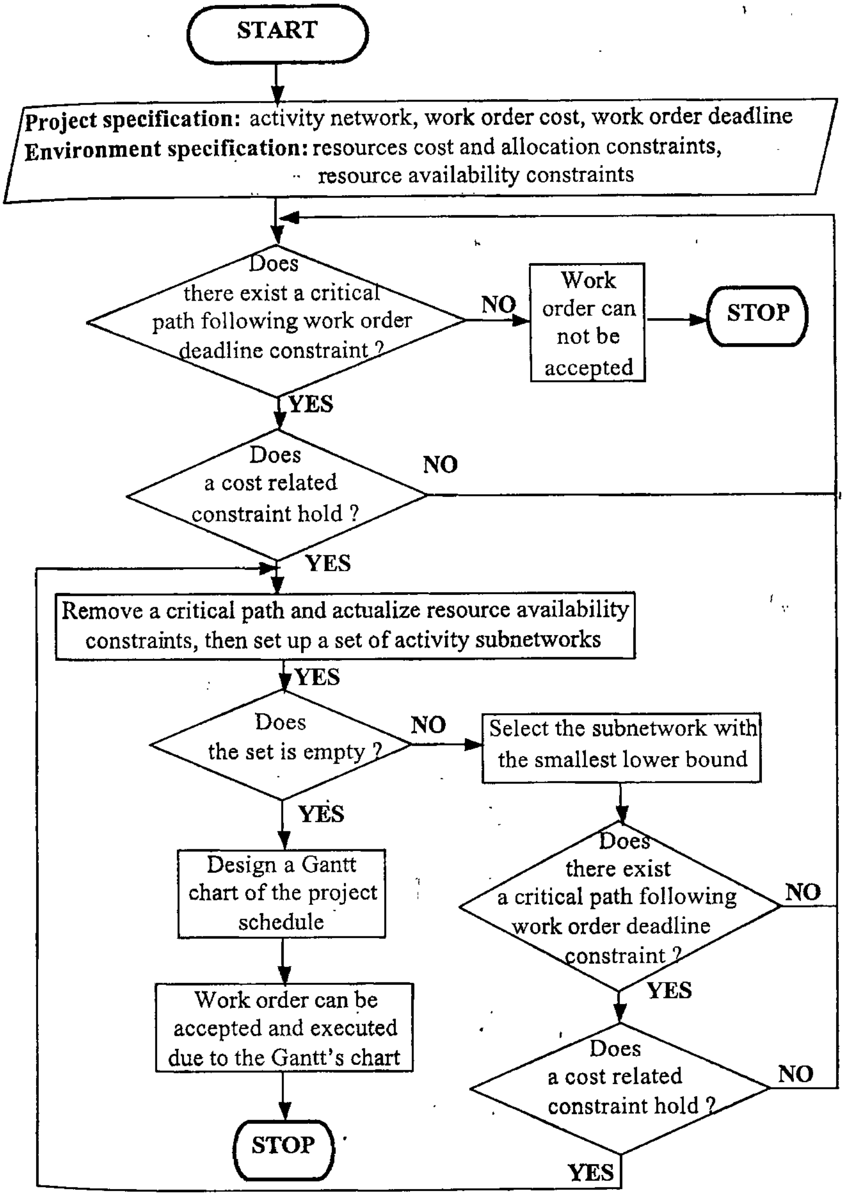

Fig. 1. Flowchart of the searching procedure

Therefore, for each subnetwork the corresponding lower bound can be calculated. Finding the subnetwork with the lowest value of the lower bound allows to repeat the main procedure, i.e. to calculate the cost and time margins, and than to consider the new subnetworks. It means from the extended set of subnetworks one has to find the element distinguished by a smallest value of the lower bound. Then calculate the margins, and so on. Procedure ends either in the case one of margins is less than zero or the set of subnetworks is exhausted.

The flowchart shown in Fig.1 provides the graphical representation of the procedure considered.

A heuristic applied focuses the searching process on the lower 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 lower bounds. This fact, in turn may lead to neglecting of some feasible schedules.

In other words, the heuristic rule applied can be treated as a set of sufficient conditions. It means 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 causes the computational efficiency of the procedure provided.

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 [ ]. That observation explains why the above observed constraints propagation based technique leads to drastically reduced size of the search tree and accelerates searching process.

4. ILLUSTRATIVE EXAMPLE

For the 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 equals $K_h = 200$ cost units.

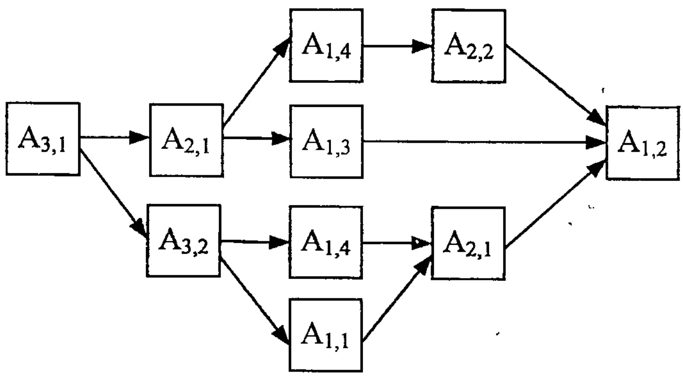

Fig. 2. A project network

Assume the variant of activities allocation as shown in Table 1. Not blanked cells of the table specify the time of activity processed with help of the assigned resource. The

Table 2 specifies the resources time-constrained availability. Not blanked cells of the table specify the cost assigned with resources usage in the unit of costs. The cost equals zero means that the given resource is not available in the moment. Due to the CPM, the critical path consists of the following activities: $A_{3,1} - A_{3,2} - A_{1,4} - A_{2,1} - A_{1,2}$ and corresponds to the following production routing: $R_2 - R_3 - R_6 - R_4 - R_6$ (see Fig. 3 – the shadow nodes depict the critical path). The minimum completion time of the project is equal to 19 units of time (Table 3).

	$A_{1,1}$	$A_{1,2}$	$A_{1,3}$	$A_{1,4}$	$A_{2,1}$	$A_{2,2}$	$A_{3,1}$	$A_{3,2}$
R_1						**2**		
R_2							**4**	
R_3								**3**
R_4					**1**			
R_5	**3**		**6**					
R_6		**4**		**7**				

Table 1. A variant of project network activities allocation

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
	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																														

Table 2. Resources availability constraints

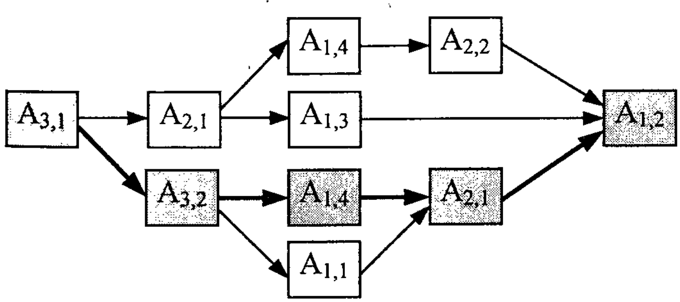

Fig.3 Critical path $A_{3,1} - A_{3,2} - A_{1,4} - A_{2,1} - A_{1,2}$

Taking into account the resources availability constraints the project makespan equals to 29 units of time (see Table 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.

Let us remove from the project network the activities belonging to the critical path. As result consider a set of subnetworks (see Fig. 4) and the new resources availability constraints (see Table 5). The lower bounds of distinguished subnetworks are as follows: 11 units of time for the subnetwork S_1, and 9 units of time for the subnetwork S_2. Note that due to the critical path from the Table 4 the duration deadlines for subnetworks S_1, S_2 are 21 (since 5 till 25), and 12 (since 13 till 24) units of time, respectively.

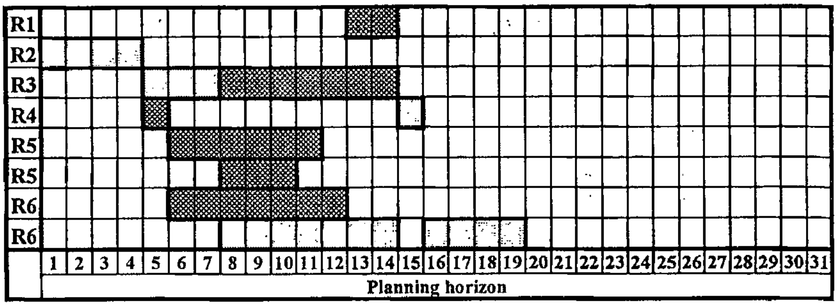

Table 3 Gantt's chart of a critical path determined under assumption the all required resources are available at the start of project

R1			3	3	3	3	3	3	3	3						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												
R3										9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4		5	5	5	5	5	5	5	5								5	5	5	5	5	5	5	5	5	5					
R5							4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4							
R6					3	3	3	4	4	4	4	4	4	4	4			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																														

Table 4 Gantt's chart of a critical path taking into account resources availability constraint from Table 2

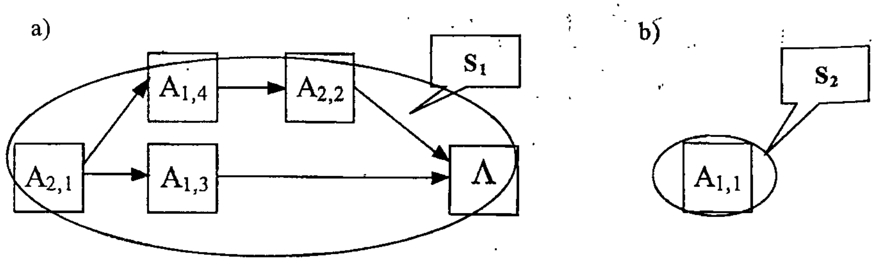

Fig.3 Subnetworks, a) subnetwork S_1, b) subnetwork S_2

Since the lowest bound corresponds to the subnetwork S_2, hence its critical path has to be determined as first (see Table 5). The relevant time and cost margins are 9 units of time and 73 = 200 – 127 units of cost, respectively. The subnetwork S_2 contains the unique path. Therefore, the searching process regards of the subnetwork S_1, as next.

R1			3	3	3	3	3	3	3	3						3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2					7	7	6	6	6	6	6	6	6	6	8	8	8	8	8												
R3													9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4		5	5	5	5	5	5	5	5								5	5	5	5	5	5	5	5		5					
R5							4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4							
R6					3	3	3	4	4	4	4	4	4	4	4										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																														

Table 5 Gantt's chart of a critical path of the subnetwork S_2

The critical path of the subnetwork S_1 consists of the following activities $A_{2,1} - A_{1,4} - A_{2,2}$ and corresponds to the following production routing $R_4 - R_6 - R_1$. The corresponding production flow is shown as the Gantt'a chart in the Table 6. Because the time and cost margins holds (i.e. the time and cost margines equal to 12 units of time and 40 units of cost, respectively) the activities of the critical path have to removed form the subnetwork S_1.

R1			3	3	3	3	3	3	3	3						3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2					7	7	6	6	6	6	6	6	6	6	8	8	8	8	8												
R3													9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4		5	5	5	5	5	5	5	5								5	5	5	5	5	5	5	5		5					
R5							4	4	4	4	4	4				4	4	4	4	4	4	4	4	4							
R6					3	3	3	4	4	4	4	4	4	4	4										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																														

Table 6 Gantt's chart of a critical path of the subnetwork S_1

Removing the activities of the above considered critical path from the subnetwork S_1 results in the unique path containing the activity $A_{1,3}$ which has to be executed in the period [6;25], i.e. within 19 units of time. The margins of time and cost corresponding to the relevant critical path (see Table 7) are equal to 9 units of time and 16 = 200 – 184 units of cost, respectively. Finally obtained feasible schedule of the project is shown in Table 8.

R1			3	3	3	3	3	3	3	3								3	3	2	2	2	2	2	2	2	2	2	2	2	2
R2					7	7	6	6	6	6	6	6	6	6	8	8	8	8	8												
R3													9	9	9	9	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10
R4		5	5	5		5	5	5	5								5	5	5	5	5	5	5	5		5					
R5							4	4	4	4	4	4				4	4	4	4	4	4	4	4	4							
R6					3							4	4	4	4										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																														

Table 7 Gantt's chart of a critical path of the subnetwork consisting of unique activity $A_{1,3}$

	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																■	■														
R2	■	■	■	■																											
R3										■	■	■																			
R4					■																				■						
R5							■	■	■	■	■	■	■	■	■																
R6						■	■	■	■	■	■							■	■	■	■	■	■	■		■	■	■	■		
	Planning horizon																														

Table 7 Gantt's chart of the feasible schedule obtained

The searching process has a tree structure as illustrated on the Fig. 5. The following notation is applied:
AN_i – the i-th activity network, $AN_{i,j}$ – the j-th subnetwork of the i-th activity network,
RAC – resources availability constraint, $RAC_{i,j}$ - the j-th actualisation of the i-th resources availability constraint,
DD – the assumed project duration deadline, DD_i – the duration deadline determined for the $AN_{k,l,...,i}$ subnetwork.

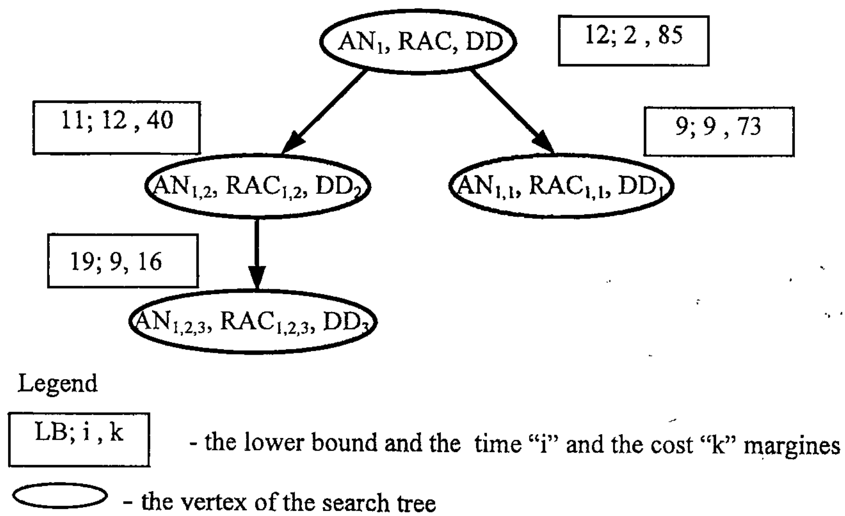

Fig. 5 Tree of feasible schedule search

The searching procedure may be though as a lower 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 $DD = 31$, $DD_1 = 12$, $DD_2 = 21$, $DD_3 = 25$.

6. CONCLUSIONS

A modeling framework supporting decision making systems design, which in turn are aimed at finding the answer whether a given work order 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.

Also, the approach proposed seems to be useful for the project-driven production flow management applied in a kind of make-to-order companies as well as for rough prototyping of the virtual organization structures.

REFERENCES

[1] Banaszak Z.A., Decision support for production flow management. *Methods of artificial intelligence*. T. Burczyński, W. Cholewa,W.Moczulski (Eds), ISBN 83-914632-3-0, Gliwice, 2002, pp81-84.

[2] Banaszak Z., Skołud B., Computer supported production flow management. In: New challenges and old problems in enterprise management. T. Krupa (Ed.), *WNT*, Warszawa, 2002, pp.111-128.

[3] Banaszak Z., Skołud B., Zaremba M., Computer-aided prototyping of production flows for a virtual enterprise. *Journal of Intelligent Manufacturing*, Vol.14, No.1, (2003), pp.83-106.

[4] Banaszak Z. A., X.Q. Tang, S.C Wang and M.B. Zaremba., Logistics models in flexible manufacturing. *Computers in Industry*, Vol.43, issue 3 (2000), 237-248.

[5] Borowiecki T., and Z.A.Banaszak., A constraint programming approach for multiassortiment workflow managing. *2nd Workshop on Constraint Programming for Decision and Control*, Ed. J. Figwer, CPDC'2000, 27 June, Gliwice, Poland,2000, 21-26.

[6] Brucker.O., et al., Resource-constrained project scheduling: Notation, classification, models, and methods. *European Journal of Operational Research*, 112, 1999, pp.3-41.

[7 Herroelen W., F. Demeulemeester, and B. De Reyck, A classification scheme for project scheduling problems. *Technical Report*, Department of Applied Economics, Katholieke Universiteit Leuven, 1997.

[8] Sihn W., R. von Briel, and T. Rost, Interactive Factory Planning. - *Preprints of the IFAC Symposium on Manufacturing, Modeling, Management and Control*, July 12-14, Rio Patras, Greece, 2000, pp.411-417.