Computer Assisted Mechanics and Engineering Sciences, **12**: 247–257, 2005.

Constraints based scheduling in the multi-project environment

Bożena Skołud, Aleksandra Zientek
Silesian University of Technology,
Department of Engineering Processes Automation and Integrated Manufacturing Systems,
Konarskiego 18a, 44-100 Gliwice, Poland.

(Received March 31, 2005)

The paper deals with the problem of new projects acceptance into the multi project environment, where constraints are limiting the number of projects that a company is able to carry out concurrently. The objective of this paper is to answer the question: Is it possible to execute new project on time in the multi-project environment? For answering the question combination of Theory of Constraints and conditions guaranteeing project due dates with constraint-based scheduling are proposed. As a result the decision of the project implementation and the schedule of project activities, which the company is able to implement concurrently are obtained.

Keywords: Theory of Constraints, multi-project scheduling, constraint programming

1. Why a new approach to multi-project scheduling is needed?

Recently, the diversification of customer demands has significantly increased the industrial competition. The decision-making for a system producing a variety of products with changeable characteristics is one of the most important and difficult tasks of management. Such problem is observed mainly in companies characterised by multi-tasking, production planning based on the receipt of customer orders and the fact that both product lead time and product price vital for customer satisfaction are subjects of negotiation with the customer (Hendry, Kingsman [6]). It is also observed that prise is not a subject of negotiation unlike products quality and delivery data. Consequently, allocation of resources and theirs capacities in order to keep the time regime is one of the most difficult tasks. In other words, the main task is to set up the activities schedule in a given system determined by the resources availability in time. Being competitive involves the organisational method of production flow, and time at which the method is being chosen and applied.

In the competitive market enterprises should react to the potential client expectations as soon as it is possible. Under such circumstances, enterprises are characterised by activities connected with unique batch production of small production series, which are never repeated once they are finished. This situation causes that the small batch production in small and medium enterprises (SME) is managed by adopting rules similar to those involved in the project management called "project driven management."

A project can be considered as an achievement of a specific objective, which involves a series of activities and tasks consuming resources. It has to be completed with defined time. The function of project management includes defining requirements of work, allocating work to resources, planning the executing (Munns, Bjermi, [9]). In practice, especially in the multi-project environment, resources have often limited availability (Demeuleemester, Horroelen, [4]). In (Banaszak, Pisz, [1]) a modelling framework that enables to cope with problems of the project-driven manufacturing by finding the computationally effective method of a feasible new project scheduling due to constraints imposed by a multi-project environment is presented.

Planning of large projects is a difficult, common and important problem of modern enterprises. Well known network planning techniques such as PERT (Program Evaluation and Review Technique) and CPM (Critical Path Method) make it possible to find minimum duration of projects assuming that the various resources required for project completion are not the constraining factor. In practice, however, project completion requires using various resources, which limited availability influences time estimations and scheduling problems. The important resource allocation problem is the resource-constrained project-scheduling problem, which involves scheduling of a project to minimise its total duration.

The inability to deal with real life project scheduling problems such as late completion, over spending, cutting specifications of a project calls for permanent analysis and new methods application. One of them is Theory of Constraints (TOC). TOC assumes that Critical Path (CP) should be extended to Critical Chain (CC) of activities using scarce resources. The aim is to focus on critical areas, by identifying CC, and to insert time buffers at appropriate places (moments) in the project network. These time buffers ensure project realisation on time. In the multi-project situation CC scheduling should be extended of drum resource scheduling, which as the constraint of the multi-project environment limits a greater number of projects. Having realised both the goal of the system and the fact of constraints presence, five focusing steps to get the most out of a system were proposed (see Fig. 1) (Leach, [7]).

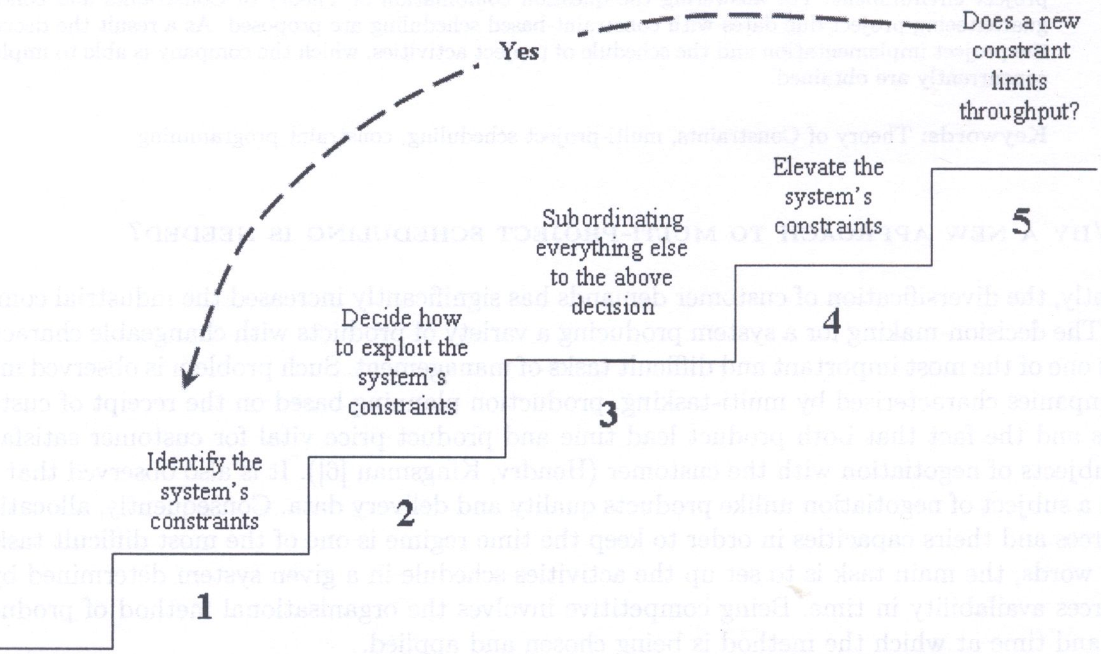


Fig. 1. Five focusing steps representing the TOC approach to ongoing improvement

In the project management system the weakest link can be anywhere: in the project management process, in company management policy, in any of the supply chains, in work producers, in the measurement system or in communication. To eliminate undesirable effects, constraints should be first identified (step 1). In step 2 it should be decided what to change. Exploiting the system constraint is getting the most out of the weakest link. Step 3 is the first part of deciding how to cause the change. It is possible that some of existing measures used to manage the project make it more difficult to meet the goal. Step 4 is an attempt to answer a question: how to cause the change.

If a constraint is broken in step 4, following 5-th step is the return to step 1. Continuous elevating the current constraint causes that eventually another constraint appears. A strong warning follows the five focusing steps: do not let management's inertia become the system constraint!

Only 5% of project plans are resource-levelled that means that conflicts in which the demand for resource(s) exceeds its supply are removed (Leach, [7]). Some companies after verifying the resource availability increase limited resources, which according to TOC means passing to Elevate Stage omitting others, preceding stages such as Identify, Exploit and Subordinate Stage (Fig.1). The constraint in a company most often is a person, but it may be also a physical, market or policy constraint. The company constraint resource becomes the drum for multi-project scheduling and it establishes the rhythm for all projects. The drum schedule determines sequencing of all projects utilising the drum resource. If the drum completes a project activity earlier subsequent projects are started earlier otherwise they are delayed. For that reason, projects in the multi-project environment require time buffers to protect the drum resource(s).

For a new project it is more practical to make use of existing resources like personnel, or machines rather than hiring or subcontracting others. In this situation there are big advantages of considering resource dependencies between projects. Sometimes resources are dedicated to specified projects, but usually occur that projects overlap (when for example one worker is required to work on two or more projects in the same time or one machine is required for several projects). By planning projects together, decisions might affect time estimations of other projects, events are better coordinated and less protection from buffers is needed. The more globally scheduling is done, the smaller (in total) buffers need to be, the shorter lead-time, and the more competitive is the company.

The constraint-based scheduling may be an approach for solving multi-project scheduling problems because it achieves a very good efficiency in solving real-life combinatorial problems like multi-project scheduling. However before solving the scheduling problem it is very important to identify all constraints. The goal of this paper is to formulate some constraints limiting throughput of a company in multi-project environment, which can be further applied to constraint-based scheduling.

Problem formulation: A system of resources determined by theirs capacities is given. Resources that are overloaded limit the number of projects that can be executed. The objective is to maximise the number of projects, which the company is able to implement concurrently. In that context, each one project consisting of many sub-projects can be treated as a multi-project environment. The following question appears: Is it possible to complete all coming projects on time in a multi-project environment?

2. Applying TOC into the multi-project environment

Single project scheduling can be successful even in the multi-project environment, in which individual projects are independent. This can happen, for example, when most of resources are contracted. In that situation CCs determine lead times and are constraints for each single project. However a great number of companies deal with numerous of interdependent projects running at the same time. In that situation the strategic resource used by a great number of projects can determine the overall throughput of the company. Planning in a multi-project environment requires pointing out some considerations (Leach, [7]):

- The market is a leverage point (constraint). The biggest inter-project impact is on the capacity of the strategic resource. Projects must be subordinated to market requirements and therefore the CC scheduling in order to obtain earlier and more reliable completion times is required.

- A resource may be considered as a company-wide strategic leverage point. If there is such a resource, attention has to be focused on it, because the overall throughput of the company is constrained by this resource.

- A non-strategic resource can become the constraint. Projects in the multi-project environment often need to be planned together. Both strategic resources and CC are important.

PERT method deals with uncertainty in the same way for all activities, whether or not they are on CP. The aim of the TOC approach is to relocate safety time buffers to strategic positions. The application of TOC assumes that almost all activities (time estimates) in a project can be reduced by up to 50%, but the safety time buffer called Project Buffer (PB) has to be added at the end of the whole project (Rand, [11]). PB should be equal to 50% of the project realisation time. In the Fig. 2 given is the effects of reducing the duration of the project; also CP and CC for the project are shown. According to (Goldratt, Cox, [5]) reduced time of all activities is transferred into PB.

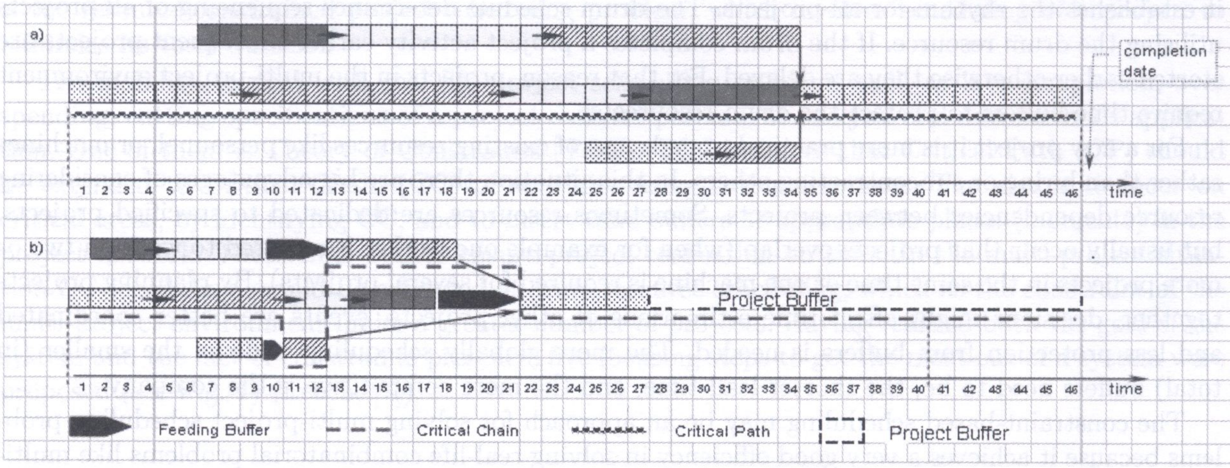


Fig. 2. The sequence of project activities. The difference between a) CP and b) CC

In the figure different colours and textures illustrate different resources. Feeding Buffers (FBs) are added at the end of non-critical sub-paths. PB is added at the end of the whole project.

In Fig. 3 an example of extending project of Project Buffer (PB) and FBs is shown. Activities executed on the constraint are marked with a dashed circle. The dashed, bold line shows project CC. If several activities of the project have to be executed on frequently limited resources, it is necessary to consider dependencies that might arise between activities requiring utilisation of the

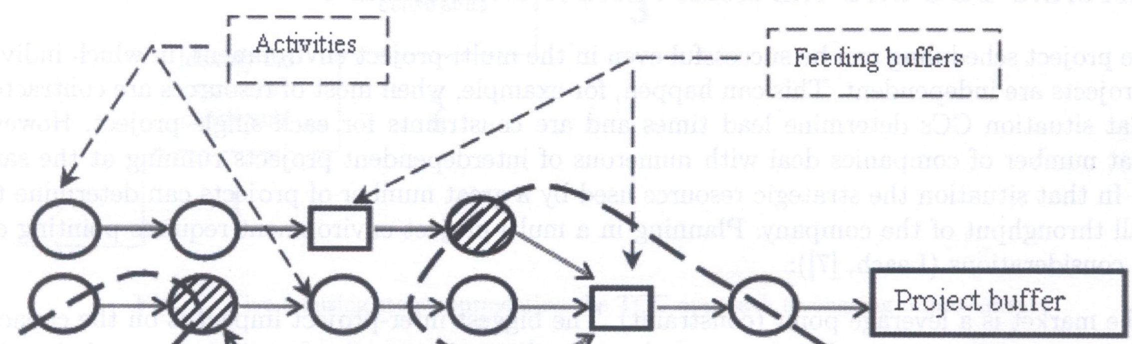


Fig. 3. A diagram showing CC

same resource(s). If that is the case, these activities must be carried out sequentially, rather than in parallel. CC is defined as the longest chain of dependent activities of a single project.

3. Introduction of a new project into the multi-project environment

3.1. New project of lower priority

According to (Goldratt, Cox, [5]) the only way to schedule a new project is through the drum schedule, which determines the system capacity. It means that activities of the new project should be allocated to company resources across all running projects. All ongoing projects have to be prioritised before creating the drum schedule. If a company prefers first-in, first-out priority rule the new project calling for realisation gains the lowest priority. If the new project is important for a company it may gain a higher priority than other ongoing projects. CC of the new project must be prepared to realisation when the drum resource(s) is necessary for its activities. Individual CC schedules determine the duration, earliest time, and relative times for each project executed on the drum. Next new project activities have to be scheduled considering ongoing projects in the proper sequence. On the basis of the new drum schedule the start time of the new project is determined by backward scheduling. The rest of the project should be scheduled forward (Leach, [7]).

In Fig. 4 drum resources are shown (R_1, R_2, R_3). It is established that the capacity of the drum cannot be exceeded. Three ongoing projects are running on three available resources. Activities of the new project are waiting for an idle resource. They should be carried out on one of resources belonging to the drum schedule. The method is to push projects activities of the lower priority later in time until there is an idle resource for their execution (dashed arrows in the Fig. 4).

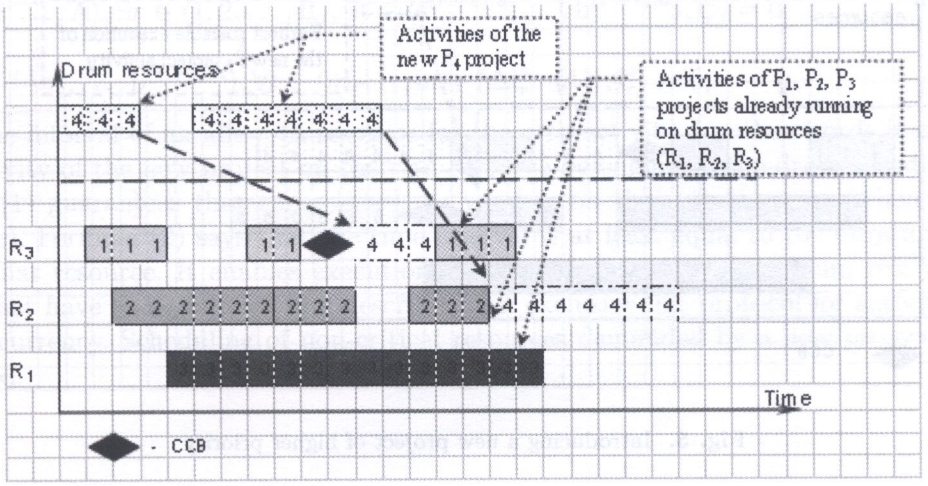


Fig. 4. A drum schedule accommodates all projects (P_1, P_2, P_3, P_4)

Critical Chain Buffers (CCBs) ensure that the constraint resources belonging to the drum are available for the new project when they are needed. CCB is placed "*between*" the prior ongoing project and the first its use in the first activity of the new project on constraint resources. It defines the start date for the resource-using activities. Drum Buffer (DB) ensures that the drum resources has projects to work on when they are required.

3.2. New project of higher priority

If the new project is of a higher priority than any already running projects, the schedule of ongoing projects of lower priority might be changed. It may lead to an interruption of work, however, nearly completed projects or those that do not have immediate resource demand from other projects should not be interrupted. The new project should be considered into the schedule assuming that it starts as early as possible, but it should be placed above the drum schedule.

In Fig. 5a the drum schedule, where three resources are available, is illustrated. The difference between this example and the one presented before is that the new, 5-th project is of a higher priority than one ongoing, scheduled before (dashed contour in Fig. 5a). Projects of a lower priority should be put above the new one (dashed arrows in Fig. 5a). Then once again both projects should be scheduled in the proper sequence in the drum schedule starting from the project of the higher priority rule.

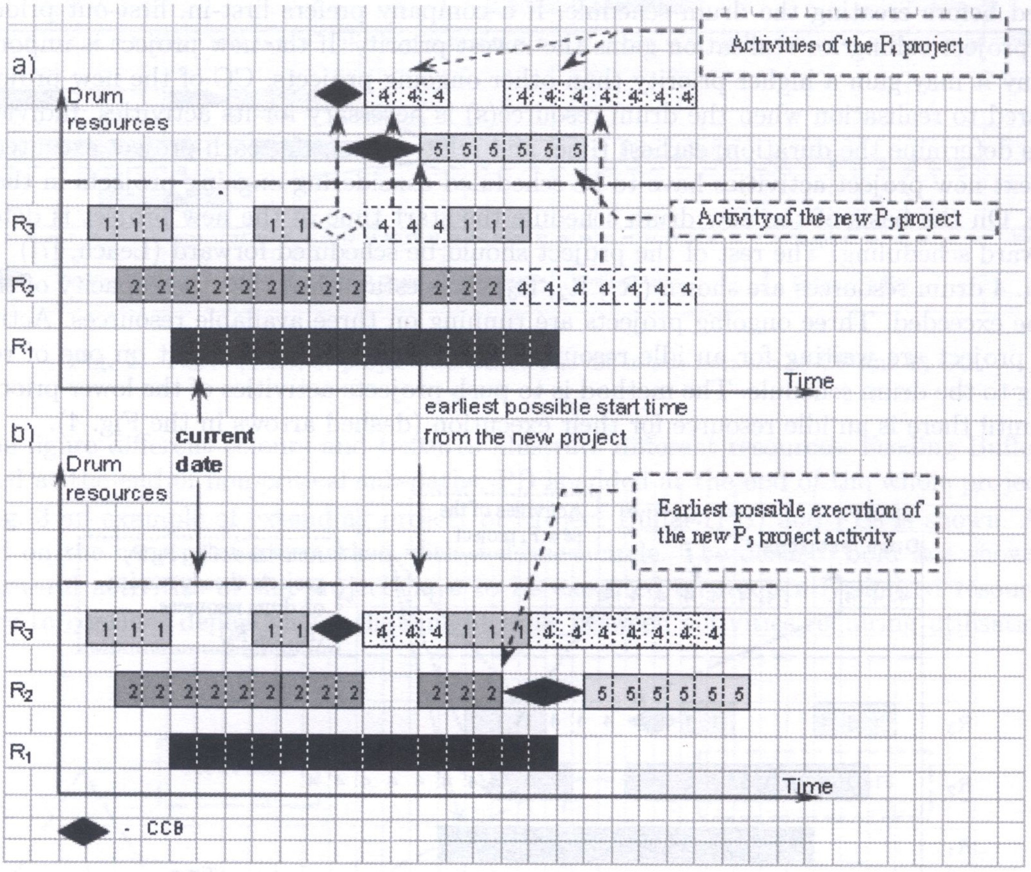


Fig. 5. Introducing a new project of higher priority

In Fig. 5b the new drum schedule of five projects is shown. The new project influences the drum resource schedule and causes that the second activity of 4-th project (P_4) is scheduled later.

3.3. Methodology

It is established that there is interdependence among activities of the new, (P+1)-th project. They ought to be performed during time specified between forward scheduling start time and backward scheduling end time of the new project. Activity of the new (P+1)-th project can be realised on any resource belonging to the drum schedule. Duration of the new project activities is constant and it does not depend on the resource, on which it is curried out. Notations are as follows:

$s_r^{p,a}$ – forward scheduling start time for the a-th activity of the p-th project on the r-th resource,

$e_r^{p,a}$ – forward scheduling end time for the a-th activity of the p-th project on the r-th resource,

$s^{(P+1),a}$ – forward scheduling start time for the a-th activity of the new (P+1)-th project,

$b^{(P+1),a}$ – backward scheduling end time for the a-th activity of the new (P+1)-th project,

R – the number of resource,

P – the number of already scheduled project,

A – the number of activities of the new (P+1)-th project executed on drum resources,

$V_r^a = [l_{r,1}^a, l_{r,2}^a, \ldots, l_{r,t}^a, l_{r,T}^a]$,

V_r^a – vector of the r-th resource occupancy by the a-th activity,

$l_{r,t}^{\,a}$ – V_r^a vector element referred to the t-th time unit, equal:

$$l_{r,t}^{\,a} = \begin{cases} 0, \text{the } r\text{-th resource is idle,} \\ 1, \text{the } r\text{-th resource is occupied by an already running project.} \end{cases}$$

Dimension of the vector V_r^a for each resource belonging to the drum schedule:

$$T^a = b^{(P+1),a} - s^{(P+1),a}.$$

Two conditions that assure fulfilment of a new (P+1)-th project on time using available resources are presented below.

$$\forall a \in \langle 1, 2, \ldots, A\rangle, \exists r \in \langle 1, 2, \ldots, R\rangle, \exists t \in \langle s^{(P+1),a}, \ldots, b^{(P+1),a}\rangle, l_{r,t}^a = 0, \tag{1}$$

$$\exists m \in \langle s^{(P+1),a}, \ldots, b^{(P+1),a}\rangle, \forall t \in \langle m, \ldots, m + (\Delta l_r^a - 1)\rangle, l_{r,t}^a = 0, \tag{2}$$

Δl_r^a – time interval of an idle r-th resource for the a-th activity. In other words it is duration of the a-th activity of the new project on the r-th resource belonging to the drum schedule.

Formula (1) guarantees that there is a time window on the r-th resource (when the r-th resource is idle). Formula (2) says that the time interval is at least equal to the duration of the new project on that resource. It enables execution of the new project on the drum resource(s). Next conditions that have to be fulfilled and described are: exclusive-like protocol for activities of a new project, concurrency. Scheduling of non-critical resources demanded by a new project also should be considered.

4. Constraints programming

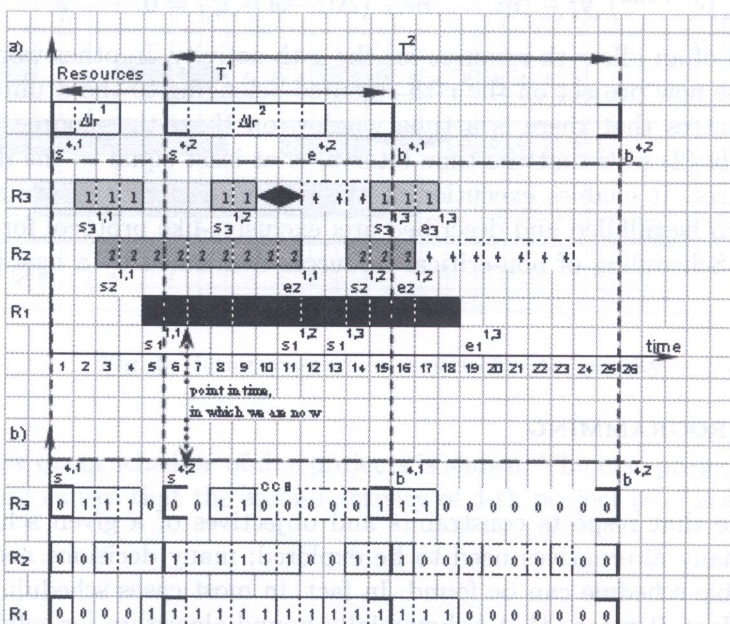
Computing a schedule that respects constraints and objectives of a given scheduling problem is *combinatorial task*: many alternatives need to be explored; many decisions need to be made and undone before a feasible schedule can be found. In fact, in most cases scheduling problems belong to the NP-complete class. It means that polynomially bound algorithm for solving them optimally does not exist yet. As a result, scheduling problems are often solved by means of *heuristics*: solution procedures that focus on finding a feasible schedule of "good" (as opposed to optimal) quality within an acceptable amount of time.

Declarative character of constraint programming (CP) supports the natural modelling of real-life combinatorial problems via specialised constraints. In the traditional operation research approach constraints binding all variables and solving methods are used to global reasoning about the problem. In the constraint programming the problem is modelled by using a set of constrains binding small sets of variables, which are much smaller than the set of all variables in the problem. The aim of constraint propagation is to achieve some level of consistency in the network of constraints and variables by removing inconsistent values form the variables domain. Constraint-based scheduling is an approach for solving scheduling problems using constraints satisfaction technique and can be modelled as constraint the satisfaction problem using a set of variables, their domains, and constraints describing feasible combination of variables values.

Decision-making, based on scheduling problems, belongs to the area of combinatorial optimisation problems and can be naturally described as constraint satisfaction problem. To model the problem as CSP, problem objects should be mapped into variables and constraints. One of traditional modelling approaches uses three variables identifying the position of the activity in time: start time, end time, and processing time (duration). The domains of activities of the new project variables are discrete (e.g. natural numbers), where release time and the deadline of the activity make natural bounds for them (and time windows make the domains even more restricted).

4.1. Illustrative example

Let us consider an organisation of three resources available in the drum schedule (DR); denoted DB = {R_1, R_2, R_3}. Three projects are already running on the drum. Activities of a new project P_4 are waiting for theirs acceptance on these resources. On the Gantt' chart (in Fig. 6a) activities of the new project are placed above the dashed horizontal line. Under the line occupancy of projects already running in the system is shown. Dashed vertical lines represent time windows for the new activity, which shows the possible delay of each activity according to forward and backward pass that ensure completion of this project on time. The lower chart (in Fig. 6b) shows the presentation of an alternative method of the idle resources notation.



for a = 2 $s^{4,2} = 5$, $b^{4,2} = 25$, $\Delta l^1_r = 3{-}0 = 3$, $\Delta l^2_r = 12{-}5 = 7$,

Fig. 6. A method of resource occupancy notation

The main purpose is to check the possibility of a new project implementation without comparing the chart representing system resource occupancy and the new project chart. Each resource belonging to the drum should be only analysed in the range of the vector V_r^a, from the current moment until the end of this vector.

$s^{4,1} = 1 \qquad b^{4,1} = 15 \qquad \Delta l_r^1 = 5$

$s^{4,2} = 6 \qquad b^{4,2} = 25 \qquad \Delta l_r^2 = 7$

Resources domains values are as follow:

$V_3^1 = [0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1] \qquad V_3^2 = [0, 0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0]$

$V_2^1 = [0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1] \qquad V_2^2 = [1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

$V_1^1 = [0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1] \qquad V_1^2 = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0]$

$b^{(P+1),a} < s^{(P+1),2}$

Additionally to simplify the example an assumption that the system is already running from some time is made. It means that verifying the resources availability should start from 7-th unit (see Fig. 6). This assumption constraint all vectors elements domains:

$V_3^1 = [l_{3,7}^1, l_{3,8}^1, ..., l_{3,15}^1] \qquad V_3^2 = [l_{3,7}^2, l_{3,8}^2, ..., l_{3,15}^2]$

$V_2^1 = [l_{2,7}^1, l_{2,8}^1, ..., l_{2,15}^1] \qquad V_2^2 = [l_{2,7}^2, l_{2,8}^2, ..., l_{2,15}^2]$

$V_1^1 = [l_{1,7}^1, l_{1,8}^1, ..., l_{1,15}^1] \qquad V_1^2 = [l_{1,7}^2, l_{1,8}^2, ..., l_{1,15}^2]$

$V_3^1 = [0, 1, 1, 0, 0, 0, 0, 0, 1] \qquad V_3^2 = [0, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0]$

$V_2^1 = [1, 1, 1, 1, 1, 0, 0, 1, 1] \qquad V_2^2 = [1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

$V_1^1 = [1, 1, 1, 1, 1, 1, 1, 1, 1] \qquad V_1^2 = [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0]$

Searching for an admissible solution, that fulfills all constraints constraining domains of R_1, R_2 and R_3 is presented in the tree form (Figs. 7, 8), in which to find the possible solution of the problem the backtracking method based on the systematically inquisition of the possible solutions is applied. The examination and produce of the solutions follows a model of non-cyclic graph is a *searching tree* (Bzdyra, Sitek, Banaszak, [3]).

Having found the possibility of the first activity implementation next step is to find an idle resource for the second activity, which cannot begin before the first one is finished.

The root of the tree represents the set of all solutions. Nodes in lower levels represent smaller sets of isolated solutions. The *tree* is created during the examination of the solutions so that no rejected solutions are created. When a node is rejected (no), the whole sub-tree is rejected, and backtrack to the ancestor of the node should be done. The procedure is based on depth-first solution strategy in which nodes in the solution tree represent resource and precedence of activities of feasible partial solution.The solution for the first activity of the new project was achieved on the 3-rd resource and of the second activity was found on the 2-nd resource (Fig. 6).

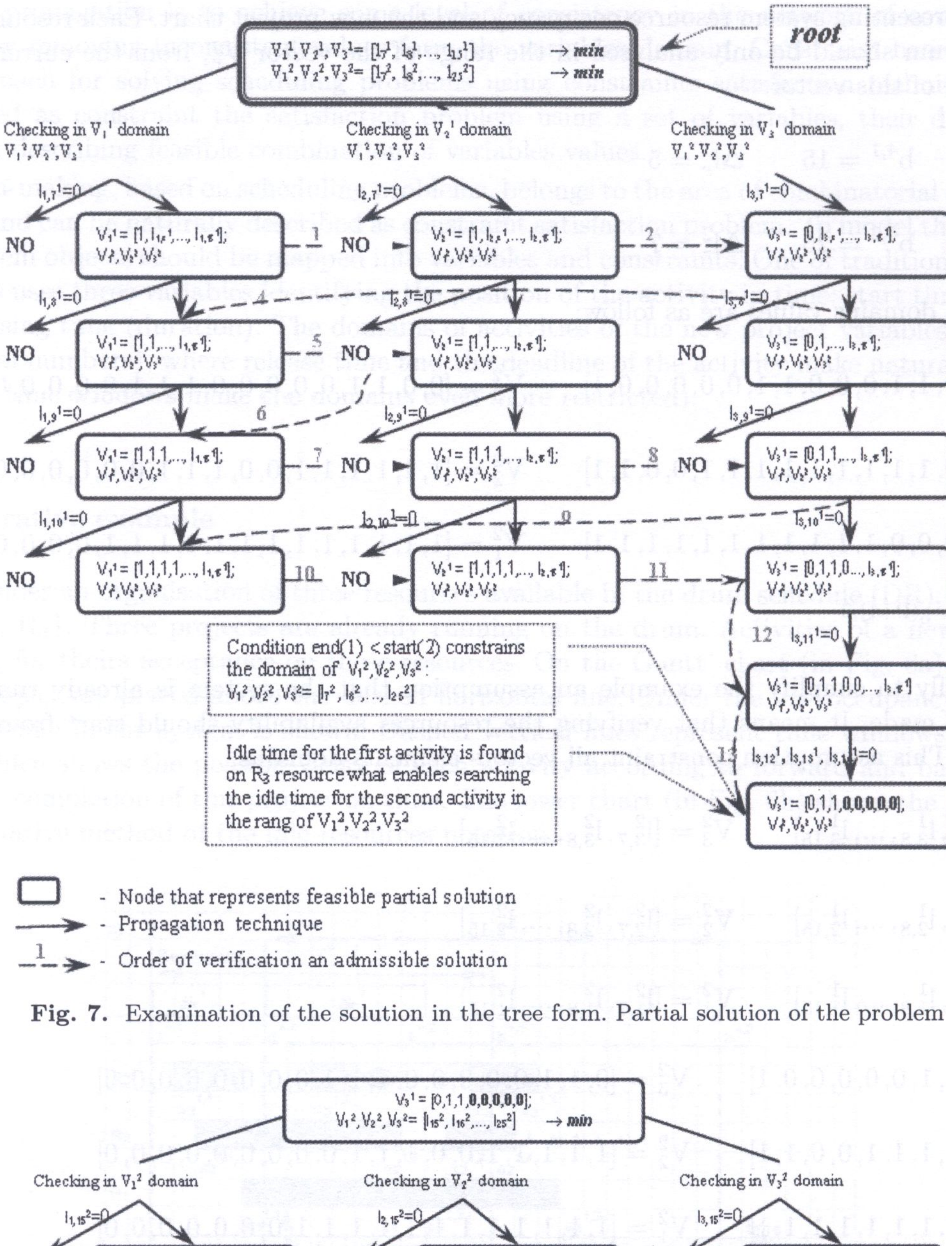


Fig. 7. Examination of the solution in the tree form. Partial solution of the problem

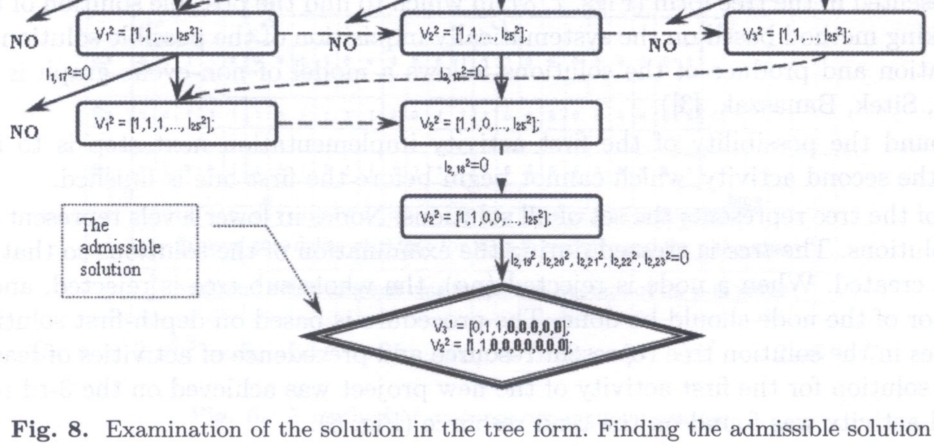


Fig. 8. Examination of the solution in the tree form. Finding the admissible solution

5. Conclusions

CC for a single project is usually not the only constraint for a company performing multi-projects. It is necessary to identify constraints in the multi-project environment and use it for projects scheduling. The drum resource(s) almost always is(are) the constraint in the multi-project environment and must be selected. All projects accessing the drum must be prioritised, which enables proper sequencing of projects in the drum schedule. CC schedules of individual project are subordinated to start times developed from the drum schedule. If the new project should be executed on the resource(s) belonging to the drum it may be introduced to the system only through the drum schedule. The existence of many disparities in the requirements of scheduling problems implies that a general-purpose software package efficiently solving all scheduling problems does not exist. Moreover, currently available scheduling software packages tend to be designed and implemented for a particular type of scheduling problem, and cannot be easily used for others: they are based on a pre-configured and inflexible scheduling model that does not extend to other scheduling situations. That is why attempts of constraint-based scheduling for solving scheduling problems are taken. Major advantages of that kind of scheduling over the existing approaches are: clarity and generality of the models. Moreover, it provides generic solution techniques of constraint satisfaction that can be further tuned for scheduling problems by using special filtering algorithms and scheduling strategies.

References

[1] Z. Banaszak, I. Pisz. *Project-driven production flow management, Project Driven Manufacturing,* WNT, 53-71 Warszawa, 2003.

[2] R. Barták. *Constraint-Based Scheduling: An introduction for newcomers. Intelligent Manufacturing Systems 2003.* IFAC Publications, Elsevier Science, 69-74, 2003.

[3] K. Bzdyra, P. Sitek, Z. Banaszak. *CLP-based approach to integrated planning of production flow,* Automation, 344-352, Warszawa, 2004.

[4] E.L. Demeuleemester, W.S. Horroelen. An effective optimal solution procedure for the pre-emptive resource-constrained project scheduling problem. *European Journal of Operational Research.* **90:** 334-348, 1996.

[5] E.M. Goldratt, J. Cox. *Critical chain* (in Polish), Werbel, Warszawa, 2000.

[6] L.C. Hendry, B.G. Kingsman. Production planning systems and their applicability to make to order companies. *European Journal of Operational Research.* **40:** 1-15, 1989.

[7] L.P. Leach. *Critical Chain Project Management,* Artech House, Norwood, 2000.

[8] A. Lowa, C. Maroto, P. Tormos. A multicriteria heuristic method to improve resource allocation in multi-project scheduling. *European Journal of Operational Research.* **127:** 408-424, 2000.

[9] A.K. Munns, B.F. Bjermi. The role of project management in achieving project success. *International Journal of Project Management* **14** (2): 81-87, 1996.

[10] T.B. NcMullen, R.C. Newbold. *Project Management in the Fast Line. Applying the Theory of Constraints,* St.Luice Press, 2000.

[11] G.K. Rand. Critical Chain: the theory of constraints applied to project management, *International Journal of Project Management.* **18**, 2000.

[12] H. Simonis. *Constraint Logic Programming,* Cosytec S.A. France 1997.

[13] B. Skołud. Planning of the multiassortment repetitive production (in Polish). Zeszyty Naukowe Politechniki Śląskiej, Mechanika, No.136, Gliwice, 2000.

[14] B. Skołud, A. Zientek. *Project Management - TOC Approach,* Automation, 177-184, 2003.

[15] B. Skołud, A. Zientek. *Constraint-based multi-project scheduling, Project Driven Manufacturing,* WNT 2003.

[16] H. Steyn. Project management applications of the theory of constraints beyond Critical Chain scheduling, *International Journal of Project Management.* **20**: 75–80, 2002.