project management, Theory of Constraints, constraints programming

Bożena SKOŁUD*, Aleksandra ZIENTEK*

AN APPROACH TO TIME PROJECT ORIENTED MANAGEMENT

Abstract

The objective presented below is to answer the question: How to exploit resources to complete all projects in expected time in a multi-project environment? A solution to maximise the number of projects, which the company is able to implement concurrently is proposed by combining the Theory of Constraints and conditions guaranteeing project due dates with constraint-based scheduling.

1. PROJECT MANAGEMENT

1.1. Project Basis

A project is a group of inter-related activities planned and executed in a certain sequence to create a unique output (a service or product) within a specific time frame. It is undertaken to accomplish a unique purpose. A project brings about change to create a unique product or service within a specified timeframe. Projects are often critical components of a company strategy or relate directly to policies and initiatives of the Government. Typically a project has its own funding cost accounting, and delivery schedule. A project can be also considered as an achievement of a specific objective, which involves a series of activities and tasks consuming resources.

Attributes of projects:

- unique purpose, development character,
- leading towards an essential result,
- temporary,
- require resources, often from various areas, have a considerable capacity,
- should have a primary sponsor and/or customer,
- involve uncertainty,
- complicated,
- professionally transverse,
- organisationally transverse.

* Silesian University of Technology, Department of Engineering Processes Automation and Integrated Manufacturing Systems, Konarskiego 18a, 44-100 Gliwice, Poland, e-mail: bozena.skolud@polsl.pl, aleksandra.zientek@polsl.pl

Every project is constrained in different ways by its:

- scope goals - that refers to all the work involved in creating the products of the project and the processes. Project scope management includes the processes involved in defining and controlling what is or is not included in the project.
- time goals - Project time management involves the processes required to ensure timely completion of a project. Processes include activity definition, sequencing, duration estimating, schedule development and its control.
- cost goals - Cost is connected with a resource utilisation to achieve a specific objective. Project cost management includes the processes required to ensure that the project is completed within an approved budget usually measured in monetary units.

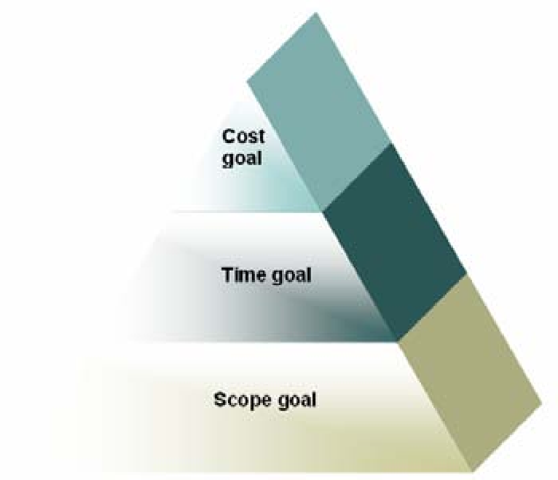

Fig.1. The Triple Constraint of Project Management

It is the project manager's duty to balance these three often competing goals. Project managers need to take a systems view of a project and understand how it is situated within the larger organisation.

In other words the structuring of the project gives the activities that should be planned according to:

- aim (what and of which quality?),
- means (how and with what efforts?),
- time (when and which dependencies?).

There is an effort connected with each activity in the form of resource consumption and time consumption (duration) and each completed activity entails greater value and degree of completion of the project.

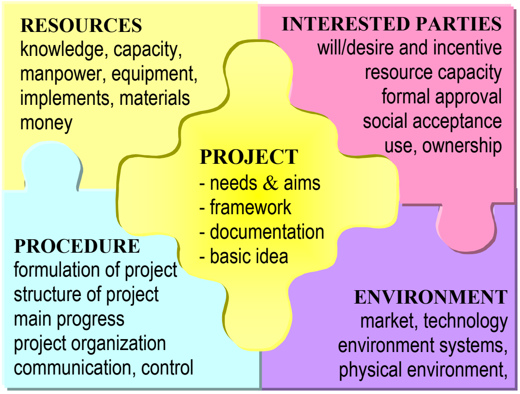

Fig.2. The five basic elements of project and its subject area

Project can have very different types starting from product development, system development, organisation/company development, technical assignments and finishing with planning/clearing assignment and arrangements.

Nowadays in the competitive market enterprises should react to the potential client expectations as soon as possible. Enterprises are characterised by activities connected with unique batch production of small production series, which are never repeated. 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. Consequently, allocation of resources and theirs capacities within the time regime is one of the most difficult tasks. In other words, the main goal is to set up the activities schedule in a given system determined by the resources availability in time. Time and capacity management may refer to the acceptance of single or small production order in the multi-project conditions. In this case decision depends on constraints of the customer (expected realisation time, delivery batch size) and producer (available machine capacity, buffers space, etc).

Project management (PM) is "the application of knowledge, skills, tools, and techniques to project activities in order to meet or exceed stakeholder needs and expectations from a project" [9] PM is a formalised and structured method of managing change in a rigorous manner, which develops specifically defined products or services (outputs) to achieve planned benefits (outcomes). It focuses on achieving outputs by a certain time, to a defined quality and with a given level of resources so that planned outcomes are achieved.

Projects vary in size or complexity. They may involve changes to existing systems, policies, legislation and/or procedures; entail organisational change; involve a single person or many people; involve a single unit of one organisation, or may cross organisational boundaries; involve engagement and management of external resources.

Good PM provides assurance and reduces risk and tools and environment to plan, monitor, track, and manage schedules, resources, costs, and quality. It also provides a history or metrics base as well as good documentation for future planning. Knowledge areas describe the key competencies that project managers must develop:

- 4 core knowledge areas lead to specific project objectives (scope, time, cost, quality),
- 4 facilitating knowledge areas are the means through which the project objectives are achieved (human resources, communication, risk, and procurement management),
- one knowledge area (project integration management) affects and is affected by all of the other knowledge areas.

1.2. Project Management Techniques

Modern PM began with the Manhattan Project, which led to develop the atomic bomb by the U.S. military. In 1917 Henry Gantt developed the Gantt chart as a tool for scheduling work in job shops. In 1958, the Navy developed PERT charts. In the 1970s, the military project management software, was developed and used by the construction industry. By the 1990s, almost every industry was using some form of project management. Professional societies like the Project Management Institute have grown extremely.

Project management tools and techniques assist project managers in various aspects of project management. Some specific ones include

- Project Charter and WBS (scope),
- Gantt charts, PERT (Program Evaluation and Review Technique) charts, CPM (Critical Path Method) analysis - (time),
- Cost estimates and Earned Value Analysis (cost).

Project management techniques helps increase the likelihood that the project will be on time, within budget and with an acceptable level of quality. Well known network planning techniques such as PERT and CPM 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 for a new project it is more practical to make use of existing resources rather than hiring or subcontracting others. In this situation there are big advantages of considering resource dependencies between projects. Sometimes resource are dedicated to specified projects, but usually occurs that projects overlap (when one resource is required for several projects in the same time). That is why the 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.

2. WHAT SHOULD BE CHANGED IN PROJECT MANAGEMENT

If several activities of a project have to be executed on the frequently limited resources, it is necessary to take into account dependencies that might arise between activities that require the utilisation of the same resource(s) and these activities must be carried out sequentially, rather than in parallel. If projects are pushed into an organisation without regard to the system's capacity and capability, work-in-process (in the form of started, but unfinished projects) quickly clogs the system. It is therefore necessary to look beyond the individual projects, or even pairs of them, to the larger system encompassed by the organisation responsible for accomplishing many projects. By planning projects together, decisions might affect time estimations of other projects. Thanks to that events are better coordinated and less protection from buffers is needed. The more globally schedule is done, the smaller (in total) buffers need to be, the shorter lead times, and the more competitive the company is.

An environment of many projects typically generates many priorities for project resources and can make that focus is difficult to achieve. Multi-project organisations very often tend to launch projects as soon as they are understood, concurrently with existing projects, simultaneously with other new efforts, and unfortunately too often, without sufficient regard to the capacity of the organisation. A common result is that the responsibility for sorting out conflicting priorities often falls to project resources and their managers. Locally set priorities may not be synchronised with each other or/and with the global priorities of the larger organisation. In addition, many project teams rely on early starts of projects to try to assure and achieve timely project completion. As a result, these practices of early starts and multitasking have been recognised as common practice in many organisations, and even in project management software tools. The common sense belief that "the sooner you start, the sooner you finish" becomes questionable. In industrial practice, however, it is only justified when the work completed by one resource allows another resource to start its work. The common practice of multitasking results in multiplying the time it takes to complete tasks, delaying true progress in projects. And if many resources in the organisation work in this manner, then most projects will take significantly longer than necessary.

One of the key challenges of multi-project or program management therefore is the avoidance of pressures on resources to multitask and the ability to assess and direct the most beneficial use of resources. The stress to multitask comes from the combination of having more than one task to do and the lack of clear priority. If there were a way of setting priorities for the maximum benefit of the organisation, it would make sense to all that we set aside some tasks to wait for the completion of the most critical. And if there were a way of reducing the queue of tasks waiting for a resource, there would be less need for assessing priorities.

Typically, projects are managed by focusing on the delivery of tasks that make up the project. It is believed that if these tasks are done on time, the project will be done on time as well. In too many cases, and for a variety of reasons, the long-established strategy of focusing on task completion does not seem to work well enough. Often, project management becomes a chaotic, resulting in inordinate pressure to meet task due dates and frequent rescheduling of the project, that is recalculating the start and finish dates of all uncompleted activities based upon progress as of a specified date. In accomplishing project there is often conflict between minimising project lead time by avoiding safety times and leaving a margin of safety time in case of any disturbance in the project. Trim a little safety in one task, causes a little lateness of the project. Add a little safety in another task, extend the project. There is never really a satisfactory compromise.

All these weaknesses mentioned before and the inability to deal with real life project scheduling problems such as late completion - lead times of projects are too long; over spending - projects over budget, not providing planned scope and cutting specifications of a project - reluctance to meet project due dates or to take on new projects calls for permanent analysis and new methods application.

3. THEORY OF CONSTRAINTS AND ITS APPLICATION TO PROJECT MANAGEMENT

3.1. Theory of Constraints

The theory of constraints (TOC) is a management philosophy developed by Eliyahu M. Goldratt. It is a systems approach based on the assumption that every organisation has at least one factor that inhibits the organisation's ability to meet its objectives. The normal objective for a business is to maximise profit. TOC emphasizes the maximisation of profit by assuring that the factor that limits production is used most efficiently. Throughput is the rate at which a system generates money through sales after reduction for material costs, commissions, and distribution cost. Under TOC, the objective is to maximise throughput while minimising operating expenses for labour, sales, and administration and simultaneously minimising investment outlays for inventory, plant, and equipment. The first step in applying TOC is to identify the constraining factor, that for manufacturing is often but not always the time available on a certain machine or process executed without any brakes. For companies that employ skilled workers, and for many service organisations, the constraint is often the time of one or a few key employees. The key to maximising profit is to concentrate on selling and producing products that provide the highest throughput per unit of constraining factor [15].

Generally speaking, knowing the goal of the system and its constraints, the following steps of TOC should be executed [14]:

1. Identify the system's constraints. This includes also prioritisation.
2. Decide how to exploit the system's constraints. It has to be decided how to manage the constraints within the system, how about the majority of the resources that are not constraints? One should manage them to provide what is needed to match the output of the constrained resources and never let them supply more output than is needed because doing so moves us no closer to the goal.
3. Subordinate everything else to the above decision in Step 2. Since the constraints are keeping from moving toward our goal one should apply all of the resources that one can to assist in breaking them. In practically all cases, in which constraints limiting impact can be reduced or eliminated.
4. Elevate the system's constraints. Continuing to work toward breaking a constraint (also called elevating a constraint) at some point will eliminate a constraint. The constraint will be broken.
5. If the constraint is broken, return to Step 1. When that happens, there will be another constraint, somewhere else in the system that is limiting progress to the goal.

The process must be repeated, perhaps many times. It is very important not to let inertia become a constraint. Most constraints in organisation are entrenched rules, policies, and procedures that have developed over time. Many times, when a constraint is finally broken, managers do not go back and review and change the rules and policies that caused the constraint initially.

Applying the management philosophy known as the Theory of Constraints (TOC) suggests that components of the system being managed subordinate their efforts to the larger system of which they are a part.

3.2. TOC in Project Management

The project management of tasks and resources that perform them must subordinate to the needs of projects, and the management of projects must subordinate to the needs of the multi-project organisation to which they belong. The TOC-based solution for managing single projects is known as *Critical Chain Scheduling and Buffer Management*. Critical Chain (CC) is formed by extending Critical Path (CP) of activities using scarce resources. The most important decision is connected with answering following questions:

- When each activity should start? (time placement decisions)
- On which resource(s) each activity should be executed? (resource allocation decisions)

A variety of constraints (like: an activity duration, release and due dates, precedence constraints, transfer and set-up times, resource availabilities, and resource sharing) may influence scheduling. These constraints are related to due dates, productivity, frequency of tool changes, inventories level, overtime, etc.

A critical chain schedule removes the pressure of artificial task due dates from the concerns of project resources and by aggregating and concentrating the safety that is typically embedded in individual tasks in a system of *buffers* positioned to protect the promise of the project.

Buffers absorb deviations from the critical chain model made up of target task durations from which significant safety has been removed. If there is sufficient unconsumed buffer related to a task waiting for attention, a resource can hold off on picking it up and multitasking, and instead, maintain focus on the current task at hand until it's complete. The deliverable of the current task can be handed off before moving to the queued task, minimising the set-down, set-up, and half-finished work that extends project lead-times when multi-tasking is the usual response.

TOC applied to multi-project systems, also provides guidance on assessing the capacity of such systems and related mechanisms for synchronised projects. TOC tends to focus on maximising flow of work through a system rather than balancing capacity. This higher-level view of system capacity rather than resource capacity leads to the conclusion that it is enough to keep only what one resource effectively utilised to manage and maximise the throughput of the system. In order to do so, it is required that other resources have sufficient protective capacity to protect that throughput.

One possible synchronising point of the system (drum of the system) might be a resource that is commonly used across projects and more heavily used relative to most other resources. In other words, in the multi-project situation the CC scheduling should be extended of drum buffer scheduling, which is the constraint of the multi-project environment and which limits a greater number of projects.

The role of the drum is to set the pace (sequence) at which projects are launched into the system. They allow overlap of project schedules, minimise peak loading on all resources and the pressure to multitask that is the usual result of these peak loads. Once a drum has been identified, a drum schedule for the multi-project program can be put together that, combined with individual critical chain project schedules, will provide the basis realistic and reliable project promises. To develop this schedule, projects are first assigned a strategic precedence. The resultant sequence of project launches its pace set by the capacity and capability of a commonly used and heavily loaded drum resource. Combined with the individual critical chain schedules systems of buffers, this drum schedule of projects allows resources and their projects to recover from delays and disruptions in a timely, rational.

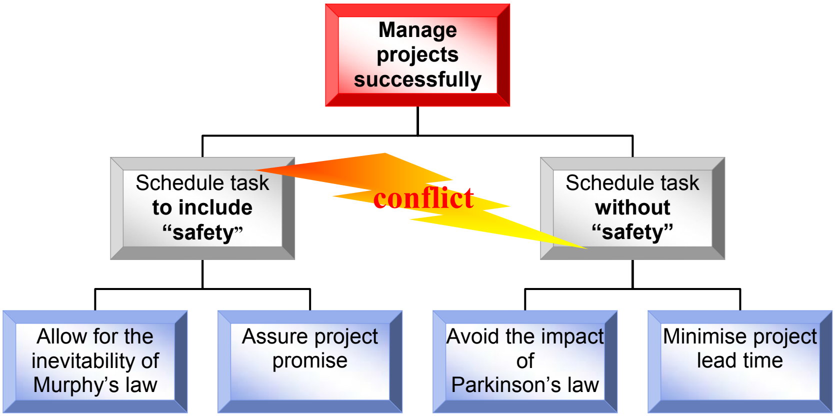

Fig.3 Between Parkinson's Law and Murphy's Law [13]

In the real world, buffers are used to absorb time deviation in the execution of tasks without influence on the resource performing the task, while at the same time protecting the promises due dates of the project. The result is the elimination of intermediate task due dates and the pressures, behaviours, and practices associated with them. These include Parkinson's Law ("Work expands to fill the time allowed for it.") and the Student Syndrome ("Delaying the start of a task due to achieve more than enough time to accomplish it.").

A project schedule can be designed to protect the project due date by taking "safety" that was spread among the tasks and concentrating it where they are most needed - as buffers at the end of the critical path and where others paths feed that critical path. Thanks to the aggregation, these buffers can be much shorter than the sum of the spread out safeties they replace, hence shortening the overall lead-time of the project. Implementations of Critical Chain (CC) scheduling and buffer management typically result in project schedules that can be 15-25% shorter than traditional schedules, but with considerably more reliability of the promised final project due date with less chaos and rescheduling. This safety must deal with the uncertainty involved in the work (Murphy's Law), the impact of distractions and interruptions they live with in their organisation, and, in many cases, the effect of dealing more than one such project at a time [4].

The drum schedule determines a sequence of all projects utilising the drum resource. If the drum completes a project earlier subsequent projects are also executed earlier otherwise they are delayed. For that reason, projects in a multi-project environment require time buffers to protect the drum resource(s) and to ensure that project(s) running on it never starve the constraint.

The aim of 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 [18]. An aggressive estimate, reflecting only the amount of work required might have a 50% level of confidence, while a longer realistic estimate might be closer to an 85-95% range of confidence. So task estimates have plenty of safety in them, above and beyond the actual expected time to do the work. Often this safety is the larger part of the estimate, doubling or tripling the amount of time the work would require if done in a vacuum.

How the promise date of an entire project from Murphy and uncertainty, which brings Parkinson and wasted safety time into the picture can be systematically protected?

Three things can help to avoid Parkinson's Law:

- Build the schedule with target durations that are too tight to allow/encourage diversion of attention.
- Get rid of task due dates.
- Charge management with the responsibility to protect project resources from interruptions rather than getting in their way with unnecessary distractions.

As previously mentioned, estimates typically include not only the amount of focused effort and time they expect the work to take, but also "safety" to deal with:

- The uncertainty involved in the work itself (Murphy's Law).
- The impact of distractions and interruptions they live with in their organisation/environment, and, in many cases.
- The effect of dealing more than one such project at a time.

The CC methodology requires that the schedule should be built with only the time to do the work without any safety. This is the time the work is expected if there are no significant problems. Usually this estimate is described in terms of having a 50% confidence level. The decision about buffers time allocation is caused by multitask approach, and the student syndrome, which means that each task starts as late as it is possible. Again one should remember that for project manager it is not important the duration of each single project stage but the total project duration.

Proficiency at managing single projects individually without dealing with the interactions between them is not sufficient to assure goals of the organisation. The system that really needs to be managed in most cases is greater than the sum of the single projects. It is a larger, complex system of projects, priorities, policies, and practices that guide the behaviours of managers and resources and requires consistent and coherent coordination for maximum effectiveness.

Project managers avoid unnecessary changes in priority by relying on buffers to absorb most of the normal, expected variability in the execution of tasks and projects. Resources have a single priority - the current task to which they are assigned. Without the distraction of pressures to multitask or to meet false priorities of task due dates, they can concentrate on the task and do it justice to assure a quality, successful projects, and maximum throughput for the organisation.

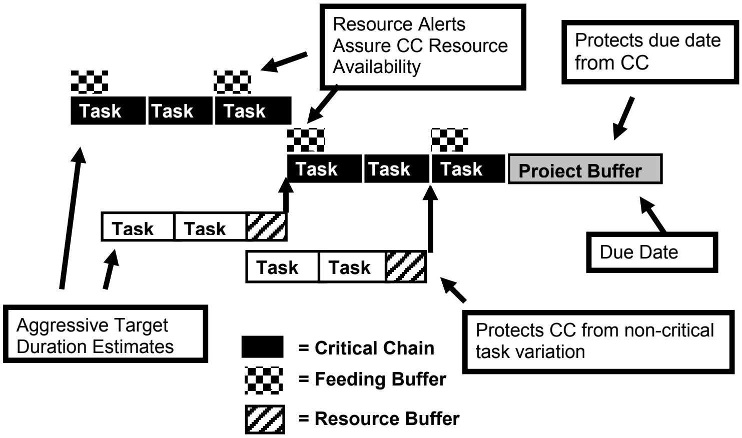

Fig.4. Time buffers: Project Buffer (PB) and Feeding Buffers (FBs) and Resource Buffers

A great number of companies deal with numerous, 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 of a multi-projects environment requires pointing out some considerations [10]:

- The market is a leverage point. The biggest inter-project impact is on the capacity of the strategic resource. Projects must be subordinated to market requirements, and therefore CC scheduling in order to obtain earlier and more reliable completion times is required.
- A resource may be considered a company-wide strategic leverage point. If there is such a resource, attention has to be focussed on it, because it contain the overall throughput of the company.
- A non-strategic resource can become a constraint. Multi projects often need to be planned together. Both strategic resources and CC are important. There are many reasons on account of which scheduling more than one project at a time can be difficult.

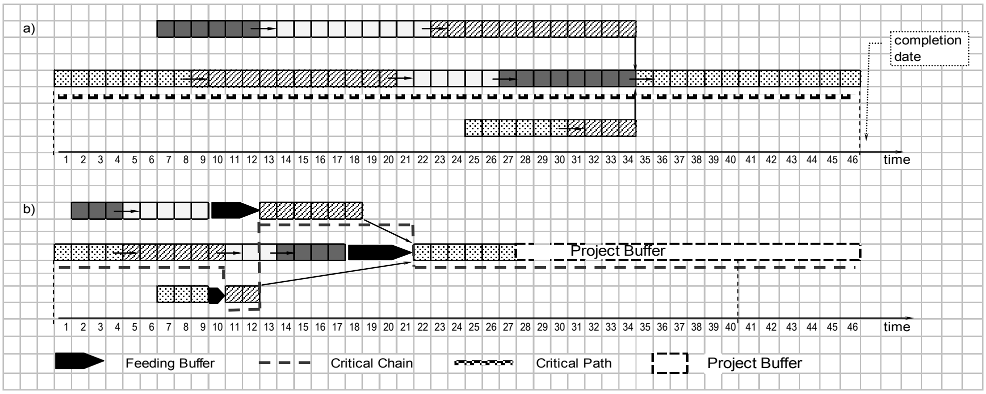

Fig. 5. The sequence of project activities. The difference between a) CP and b) CC

In Figure 5 duration of activities was reduced by 50%. Different resources are illustrated by different texture or shade. FBs are added at the end of non-critical sub-paths.

A major distinction from a schedule based on critical path methodology is the proactive approach of using feeding buffers to keep the critical chain critical up front rather than relying on reacting to a changing critical path. A resource-constrained critical path can be used as the project's critical chain. The Critical Chain Schedule avoids expansion from Parkinson's Law by eliminating due dates and allowing us to take advantage of early task finishes. This schedule is also protected against untimely availability of critical resources by the alerts of work coming from preceding tasks. The project promise is protected from variation (Murphy) in the critical chain by the project buffer and the critical chain is protected from variation in non-critical work by the feeding buffers.

A number of benefits that can be obtained by projects that make use of the Critical Chain Scheduling and Buffer Management approach include the following:

- An aggressive target duration schedule, along with elimination of task due-dates, minimizes impact of "Parkinson's Law."
- Buffers allow resources to focus on work without task due-date distraction and efficiently protect against "Murphy's Law" with shorter project lead-times through concentrated safety protecting what is crucial to project success.
- Resource alerts and effective prioritization of resource attention allow projects to take advantage of good luck and early task finishes while buffers protect against bad luck and later than scheduled finishes.
- Buffer Management provides focus for schedule management, avoids unnecessary distraction, and allows recovery planning to take place when needed, but well before the project is in trouble.

There are additional benefits of this approach when the concepts that underlie it are expanded to multi-project environments. The use of buffers to prioritise resource attention allows such organisations to allow the focus on the task to speed projects in the context of multi-project programs. The Critical Chain approach to single projects allows the multi-project environment to avoid the lead-time multiplying effect of multi-tasking.

Some of the significant shifts obtained by the implementation of Critical Chain Scheduling and Buffer Management include:

- Stop spreading safety, hidden and wasted in the tasks. Concentrate safety in strategic places that protect what is important to the project from Murphy's Law. This can only happen effectively when resources trust management and project owners to accept that their tasks-target durations are not commitments and that the buffers are sufficient to protect the project.

- Stop the behaviours that waste time in the project. Avoid task due-date focus and Parkinson's Law. Old habits are hard to break. Project managers must stop publishing date-laden project schedules.
- Avoid resource multi-tasking and the lead-time multiplication it results in. Focus on the task at hand. Management must take responsibility for protecting resources from competing priorities that drive multi-tasking.
- Account properly for resource contention. Project managers, when building project schedules must realize resource dependency is as real as task dependency when determining what is critical for the project.
- Track the consumption and replenishment of buffers. The project team must plan and act to recover when necessary, as dictated by buffer status, but only when necessary, in order to avoid unnecessary distraction of project resources who should be allowed to focus on their work.

4. APPROACHES TO THE NEW PROJECT ALLOCATION

Activities of a new project should be allocated to a company resource constraint across all running projects prioritised before creating the drum schedule. The drum schedule determines the system capacity. Individual CC schedules determine the duration, earliest time, and relative times for each project executed on the drum. The new project priority should be determined before introducing a new project into the current running system. If a company prefers first-in, first-out priority rule the new project calling for execution gains the lowest priority. If the new project is important for a company it may gain a higher priority than other ongoing projects. The CC of the new project must be prepared to determine when the drum resource(s) is necessary for its activities. Next new project activities have to be scheduled together with 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 is scheduled forward.

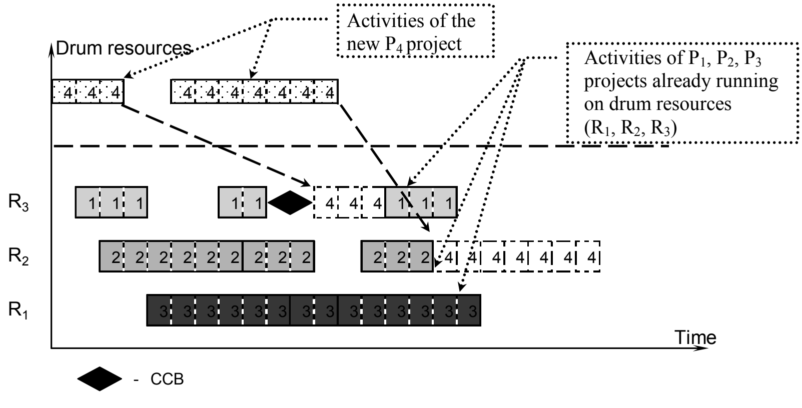

Fig.6. A drum schedule accommodates all four projects (P_1, P_2, P_3, P_4)

Let's assume that three resources (R_1, R_2, R_3) belong to the drum (see Figure 6). It is established that the capacity of the drum can not be exceeded. Three ongoing projects are running on three available resources. Activities of the new project are waiting for an idle resource in the drum schedule. The method is to schedule projects activities of the lower priority later in time. Critical Chain Buffer (CCB) ensures that the constraint resources belonging to the drum are available for the new project when they are needed. CCB is placed between the use of the constraint resource in the prior ongoing project and the first its use in the first activity of the next project. Moreover Drum Buffer (DB) ensuring that the drum resource(s) has projects to work on when they are required.

If the new project is of a higher priority than any of the running projects, the schedule of already accepted projects of lower priority might be changed. The difference between this example and the one presented before is that the new, 5-th project is of a higher priority than the ongoing one which was scheduled before (dashed contour in Fig. 7a). The project of a lower priority should be put above the new one (dashed arrows in Fig 7a). Then once again both projects should be scheduled in the proper sequence in the drum schedule starting from the project of a higher priority rule.

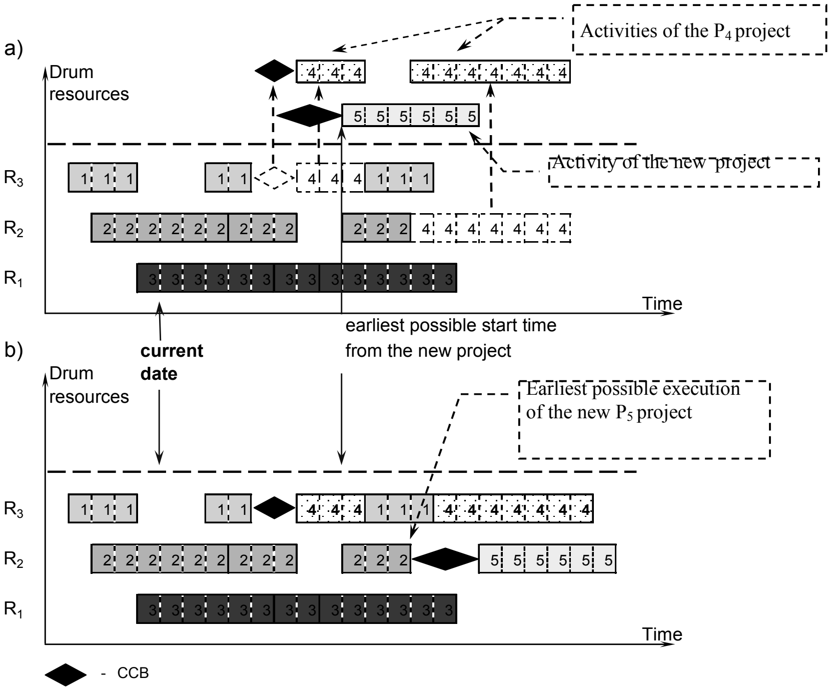

Fig.7. Introducing a new project of higher priority

The new project influencing the schedule of the drum resource and causes delay of the second activity of 4-th project is shown in Figure 7b.

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

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

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.

5. 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).

5.1. ILLUSTRATIVE EXAMPLE

It is established that the capacity of drum resources cannot be exceeded and resource occupancy of already running projects cannot be changed. In Figure 8a. three ongoing projects are running on available resources, belonging to the drum DB={R_1,R_2, R_3}. 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 figure). The project duration is determined by a critical sequence of activities which reflects both precedence of the network and interdependencies implied by activities sharing resources. The lower chart (in Figure 8b.) shows the idle resources notation.

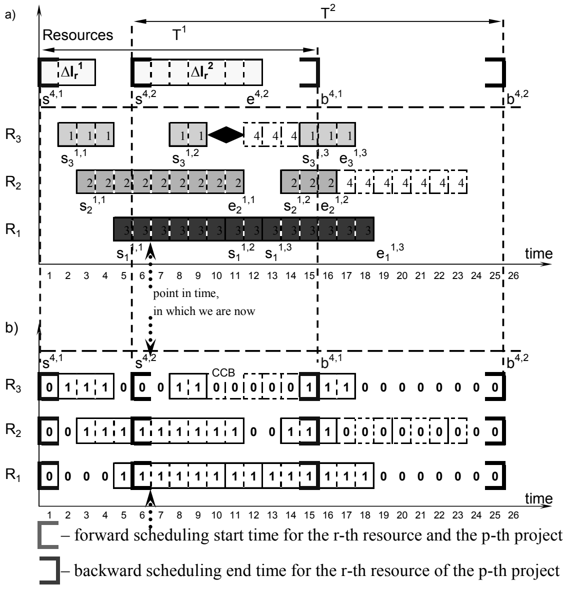

Fig.8. A method of resource occupancy notation (P1, P2, P3 - projects already running on drum resources) for a=2 $s^{4,2}$=5, $b^{4,2}$=25, Δl_r^1 =3-0=3, Δl_r^2 =12-5=7

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.

$s^{4,1}= 1$ $b^{4,1}=15$ $\Delta l_r^1=5$ $s^{4,2}= 6$ $b^{4,2}=25$ $\Delta l_r^2=7$

Resources domains values are as follow:

$V_3^1= [0,1,1,1,0,0,0,1,1,0,0,0,0,0,1]$
$V_2^1= [0,0,1,1,1,1,1,1,1,1,0,0,1,1]$
$V_1^1= [0,0,0,0,1,1,1,1,1,1,1,1,1,1]$

$V_3^2= [0,0,1,1,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0]$
$V_2^2= [1,1,1,1,1,1,0,0,1,1,1,0,0,0,0,0,0,0,0,0]$
$V_1^2= [1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0]$

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

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 7th unit (see Fig.8). This assumption constraint all vectors elements domains:

$V_3^1= [l_{3,7}^1, l_{3,8}^1,..., l_{3,15}^1]$ $V_3^1= [0,1,1,0,0,0,0,0,1]$
$V_2^1= [l_{2,7}^1, l_{2,8}^1,..., l_{2,15}^1]$ $V_2^1= [1,1,1,1,1,0,0,1,1]$
$V_1^1= [l_{1,7}^1, l_{1,8}^1,..., l_{1,15}^1]$ $V_1^1= [1,1,1,1,1,1,1,1,1]$

$V_3^2= [l_{3,7}^2, l_{3,8}^2,..., l_{3,15}^2]$ $V_3^2= [0,1,1,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0]$
$V_2^2= [l_{2,7}^2, l_{2,8}^2,..., l_{2,15}^2]$ $V_2^2= [1,1,1,1,1,0,0,1,1,1,0,0,0,0,0,0,0,0,0]$
$V_1^2= [l_{1,7}^2, l_{1,8}^2,..., l_{1,15}^2]$ $V_1^2= [1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0]$

Searching for an admissible solution, that fulfils all constraints constraining domains of R_1, R_2 and R_3 is presented in the tree form (Fig.9,10), 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* [2].

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 3rd resource and of the second activity was found on the 2nd resource.

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.

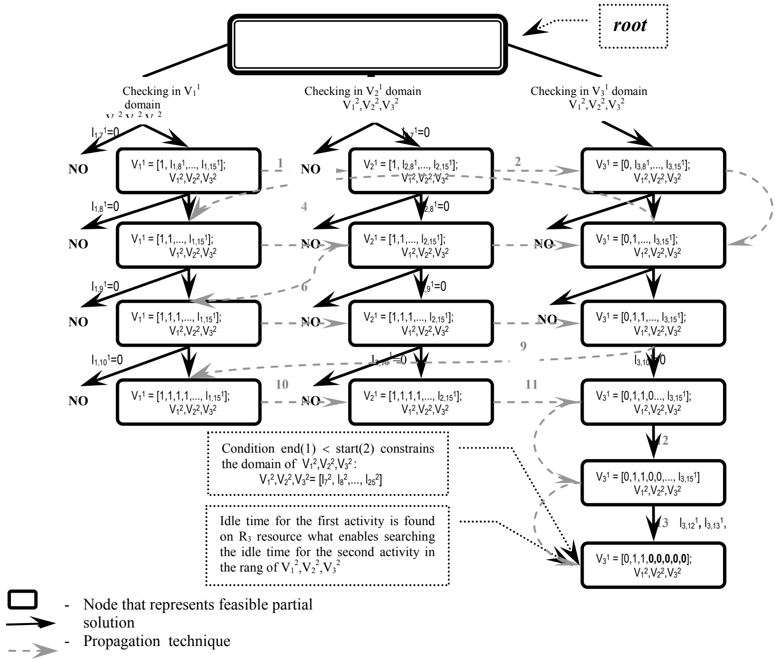

Fig.9. Examination of the solution in the tree form - partial solution of the problem

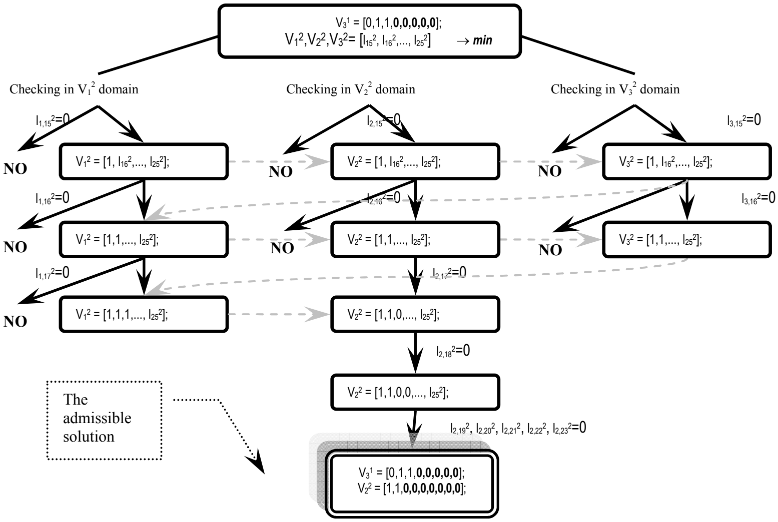

Fig.10. Examination of the solution in the tree form - finding the admissible solution

5.2. Partial resource occupancy

However considering resources as stands that are able to concurrently do work, it is useful to think of them as having partial occupancy that allows for the execution of more than one task at the same time. When talking about resource utilisation planning it is important to consider the state of partial occupancy, which can be represented as a third dimension of resource occupancy. The order consumption profile of a resource is shown in Fig. 11.

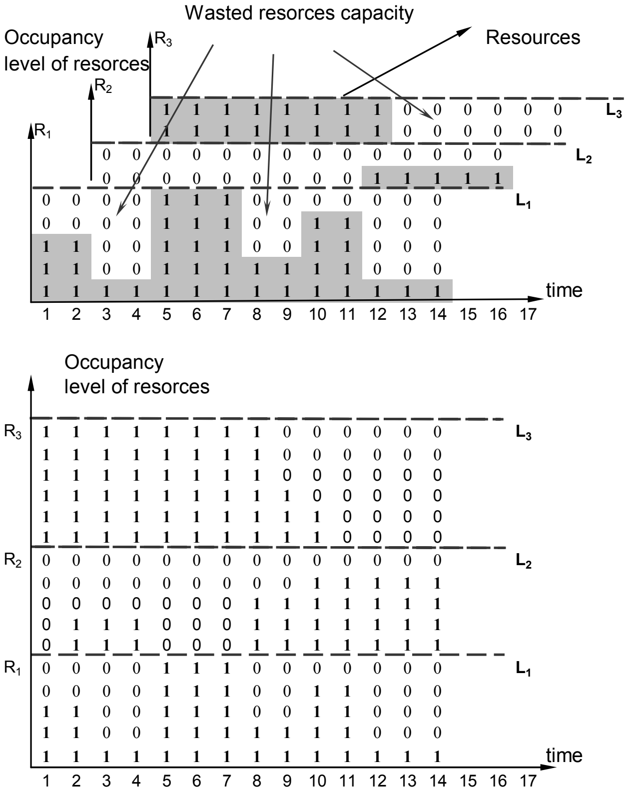

Fig.11. A resource occupancy-consumption profile of a resource

It is established that $L=L_1=L_2=L_3$. L_r – occupancy level of r-th resource; r – the number of resources, r = 1,2 …, R;
L – the maximum occupancy level of all resources, assigned to the project; $L = \max(L_r)$;

It is only possible to relocate the new project among available time units, in such way to enable their performance during specified time. The resource occupancy of already running projects cannot be changed There is admitted principle that one unit is an indivisible and it may be established by project manager. One unit values:

$l_{r,t}^a$ – V_r^a vector element of the r-th resource occupancy by the a-th activity referred to the t-th time unit, equal:

$$l_{r,t}^a = \begin{cases} 0, & \text{the r-th resource is idle,} \\ 1, & \text{the r-th resource is occupied by an already running project or the r-th resource is not available for new project.} \end{cases}$$

If the wasted capacity area is larger or equals the new project occupancy, it is possible to implement the new project into an already running project.

While a given project is being performed, the maximum level L_r of each resource r, on which the project is be executed is assigned to it . Resources are related to services per unit time of various machines. If the project that is being executed requires fewer resource units than L_r, then the remanding units will be still idle for that resource. If the considered project requires more resource units, then it will use all the L_r units for an extended period of time such that the total work served in that stage is executed.

Formulation (3) enables to state if resources occupancy guarantee the new project execution.

$$W_r + W_k \le (b^{(P+1),a} - s^{(P+1),a}) \cdot L_r \qquad (3)$$

where:
W_r – the processing time occupied by already implemented projects demanded from resource R.
W_k – the processing time demanded from r-th resource for new project;

The alternative method of notation considering the state of partial occupancy can be described in form of matrix:

- Resources occupancy matrix – it is a matrix describing the level of partial resource occupancy on each resource in given time per unit. – occupancy resulting from execution already scheduled activities during t-th time unit on the r-th resource;

$$Ro = \begin{bmatrix} V_R \\ . \\ . \\ . \\ V_1 \end{bmatrix}$$

where:
V_r^a – vector of the r-th resource occupancy by the a-th activity, $V_r^a = [l_{r,1}^a, l_{r,2}^a, ..., l_{r,t}^a, l_{r,T}^a]$,
Dimension of the vector V_r^a For each resource belonging to the drum schedule: $T^a = b^{(P+1),a} - s^{(P+1),a}$.

- Partial resources occupancy matrix

In case of considering resources as having partial occupancy, the Resources occupancy matrix should be extended of the l-th occupancy level in t-th time unit – occupancy resulting from execution the l-th occupancy level in t-th time unit on the r-th resource;

$$Rpo = \begin{bmatrix} M_R \\ . \\ . \\ . \\ M_1 \end{bmatrix} \qquad \text{where:} \qquad Mr = \begin{bmatrix} l_{r,1}^{a,L} & \dots & l_{r,T}^{a,L} \\ \dots & \dots & \dots \\ l_{r,1}^{a,1} & \dots & l_{r,T}^{a,1} \end{bmatrix}_{L \times T}$$

l=1,2,3...L – occupancy level of r-th resource;

Each partial resource occupancy matrix has L×N dimension, where $L = \max(L_r)$ and dimension of the vector V_r^a equals $T^a = b^{(P+1),a} - s^{(P+1),a}$.

6. CONCLUSION

In this chapter the approach to the project oriented production throughout multi-project management is proposed. The Backward and Forward Pass is applied for scheduling a new project into the multi-project environment, which constraints searching-field of possible solutions. In the multi-project environment it is necessary to identify constraints and use them for projects scheduling. 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 alternative method of notation considering the state of partial occupancy was proposed. Attempts of constraint-based scheduling for solving scheduling problems are taken. Its 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. Constraint-based scheduling is an efficient tool for solving real-life scheduling problems.

References

[1] BARTÁK R.: *Constraint-Based Scheduling: An Introduction for Newcomers.*

[2] BZDYRA K., SITEK P., BANASZAK Z.: *CLP-based approach to integrated planning of production flow*, Automation 2004, Warszawa, pp.344-352.

[3] Francis PATRICK: *Critical Chain Scheduling and Buffer Management*, Project Management Institute's, April, 1999.

[4] Francis PATRICK: *Program Management - Turning Many Projects into Few Priorities with TOC*, Project Management Institute Symposium, Philadelphia, October, 1999.

[5] GOLDRATT E.M., COX J.F.: *The Goal,* Coton on Hudson, NY; North River Press 1987.

[6] GOLDRATT E.M. COX J.F.: *Łańcuch krytyczny*, Werbel, Warszawa 2000.

[7] HENDRY L.C., KINGSMAN B.G.: *Production planning systems and their applicability to make to order companies*. European Journal of Operational Research, No.40, 1989, pp.1-15.

[8] NCMULLEN T.B., NEWBOLD R.C.: *Project Management in the Fast Line. Applying the Theory of Constraints*, St. Luice Press, 2000.

[9] *Project Management Body of Knowledge* (PMBOK Guide), Project Management Institute, 2004.

[10] RAND G.K.: Critical Chain: *The theory of constraints applied to project management*, International Journal of Project Management 18, 2000.

[11] SIMONIS H.: *Constraint Logic Programming*, Cosytec S.A., 11/12/1997, France.

[12] SKOŁUD B., ZIENTEK A.: *Constraints identification in multi-project scheduling*, AIMETCH 2004 Conference, 129-130.
[13] http://www.focusedperformance.com/articles/ccrisk.html
[14] http://www.sytsma.com/cism700/toc.html
[15] http://www.nysscpa.org/cpajournal/1999/0499/Departments
[16] http://www.goldratt.com/