

REFERENCE MODEL OF PROJECT PROTOTYPING PROBLEM

Marcin RELICH*, Zbigniew BANASZAK**

*Faculty of Economics and Management
University of Zielona Góra, Zielona Góra, Poland
email: m.relich@wez.uz.zgora.pl

**Faculty of Management
Warsaw University of Technology, Warsaw, Poland
email: z.banaszak@wz.pw.edu.pl

Abstract: The paper presents the idea of reference model of project prototyping problem for the projects that are at risk of failure. The hierarchical structure of declarative model connects two fields: functionalities of a typical service enterprise and management system of project execution in the enterprise. The functionalities as separate Constraints Satisfaction Problems (CSP) are described. CSP contains the sets of decision variables, their domains and constraints, which link these variables. The separated problems described as CSP, then in single main CSP are integrated. On the other hand, these problems can decompose into the subproblems concerning the functionalities of different fields. The open structure of model enables to solve the decision problems with different level of specificity. The decision problem can regard a query about the results of proposed decisions as well as the decisions guaranteeing the expected results. A declarative kind of proposed reference model in a natural way allows to implement its in constraint programming languages. The possibility of this approach illustrates an example.

Key words: project management, reference model, constraints satisfaction problem, constraint programming, alternative projects, knowledge base, decision support system, allocation planning.

1 Introduction

In the activity of present organizations more and more importance takes projects. A project is a sequence of unique, complex, and connected activities having one goal and that must be completed by a specific time, within budget, and according to specification [11]. Hence, there is an increase of the demand for new knowledge that enables solution of the problems during the complex project execution. In this case, the knowledge concerning project management has the particular significance. Especially, the identification of project success or failure is desirable, what is usually connected with specific methods and techniques.

Many cases of projects indicate that fewer than half of projects met cost and schedule targets [6, 13, 17, 20 and 23]. The findings of various other authors indicate that projects which overrun are more common than projects which complete within original time scales, overruns likely to be between 40% and 200% [16]; for instance, only one third of World Bank projects met their aims, with typical delays of 50%. Another survey showing only 17% of projects meeting all three aspects of the project triangle (cost, time, and scope), with typical cost overruns as high as 189% [7]. In the case of software projects, the surveys on estimation perfor-

mance report that 60-80 percent of all software projects encounter effort overruns [8, 12 and 21].

Project success or failure depends on many critical factors, such as the kind of project, accessibility of resources, project management, and environment [2, 16]. The reasons for project failure can be generally considered in accessibility of resources (e.g. human, financial, raw materials) and changeability of the external environment. Moreover, unstable requirements, lack of well-defined scope, quality of management, and skill of the employees can cause project failure. Another factor is that an enterprise, which carries out a few projects, can change the priority of a project.

The effective project development requires planning that supports, among other things, the estimation of project progress, resources, time, etc., which are fundamental to guide the project activities. To reduce project overruns, there are two ways to approach the problem. The first way is to increase the accuracy of the estimates through a better estimation process and the second, to increase the project control.

It is unrealistic to expect very accurate estimates of project effort because of the inherent uncertainty in development projects, and the complex and dynamic interaction of factors that influence on its development.

However, even small improvements will be valuable, especially by large-scale projects. More accurate forecasting supports the project managers in planning and monitoring the project, for instance, in the project price set, required cost, resource allocation or schedule arrangement.

If the project does not develop according to the plan, there is a need for project rescuing. The subject of project rescuing in preventing phase usually focuses on the issues of starting with well-defined requirements, more accurate planning and controlling of project execution or frequent meetings between project team members [10]. In case of essential variance between the original project schedule of due tasks and the actual project schedule of done tasks, there are a few simple rules, e.g. add resources to reduce the resource bottlenecks or change the requirements and the commitment to customers [15, 24]. However, in the research works is a lack of the approach that will generate a set of alternative variants of project completion and support the decision-maker. The alternative variant is considered as a modification of the original project that can regard time, cost or scope of a project.

Rapidly changing expectations related to supporting strategic decisions, as well as aiming to reduce cost and investment risk, result in the need to make a task-oriented decision support system. Most of the publications have separately considered the fields of enterprise and project management. This results in a separate knowledge base respectively for an enterprise and project management. Consequently, it implies the difficulty of implementation of these fields within a single tool that is used for decision support. Hence, there is a need to build a single model that combines the fields of enterprise and project management, and that provides a base for making a task-oriented decision support system.

The paper is organized as follows. A reference model concerning an enterprise and project management, which is described in a standard form of the so-called constraint satisfaction problem, is presented in section 2. A project prototyping problem is formulated in section 3. A method for obtaining alternative variants of projects is shown in section 4. An illustrative example of the approach, which presents a possibility of decision problem specification in the straight and in the reverse way, is presented in section 5. Finally conclusions and future research are presented in section 6.

2 Reference model

An enterprise as a complex system [3] may be described by multiple criteria regarding its structure and environment, as well as allowing for a variety of enterprise classifications. The exemplary fields of a enterprise activity, its environment and classification are shown in Fig. 1.

Note that each area can be described by a set of criteria, measures hereinafter referred to as decision variables. An example might regard a determination of the enterprise's size by number of employees, annual turnover or production volume. Moreover, these areas can be interrelated, e.g. volume of materials purchased in a given time depends on information concerning the planned sale, manufacturing (among others things available capacity) or stocks of these materials in storage.

In the same way, it is also possible to consider project management issues. According to the Project Management Institute [14], project management consists of nine knowledge areas (see Fig. 2).

The project implementation follows according to kind of the enterprise and its resources. For this reason, the fields of enterprise activity influence on the fields of project management. For instance, the type of enterprise activity determines the feasibility of a project. Also size or type of organisation determines a project execution, because small private business by financial and personal constraint, cannot assure the available resources for large-scale project, for example, airport building.

The proposed approach combines the fields of an enterprise and project management in form of single platform – the reference model. This way seems to be natural in case of an enterprise that executes projects and solves standard decision-making problems. In this case, a knowledge base is created that in addition to the inference strategies allows to implement a decision support system more efficient.

The fields of enterprise activity and project management (see Fig. 1 and 2) regard a general case, that allows describing any type of project in any enterprise. To illustrate an idea of a connection these fields into one reference model, the general case is thereafter limited to the medium service enterprises, which implement software project.

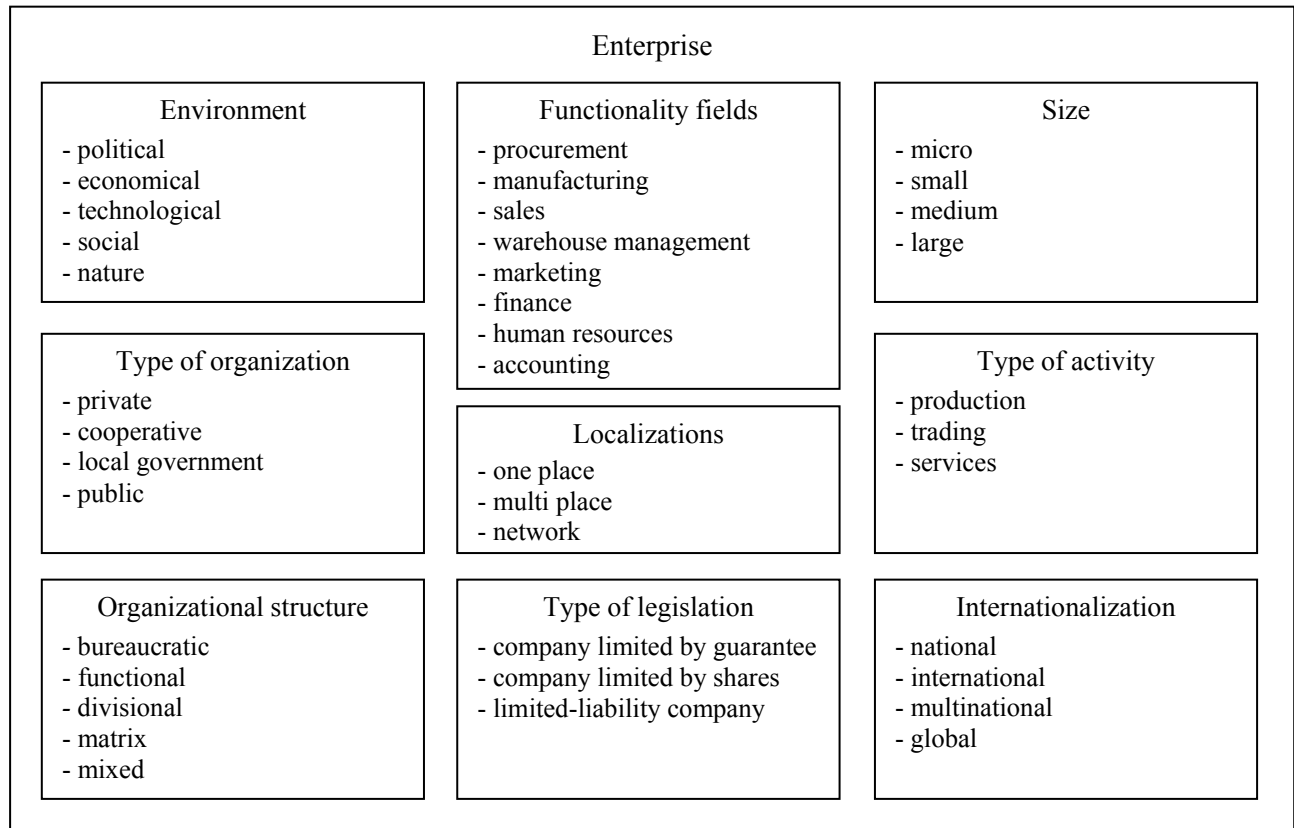


Figure 1. Fields of enterprise activity
(source: self study)

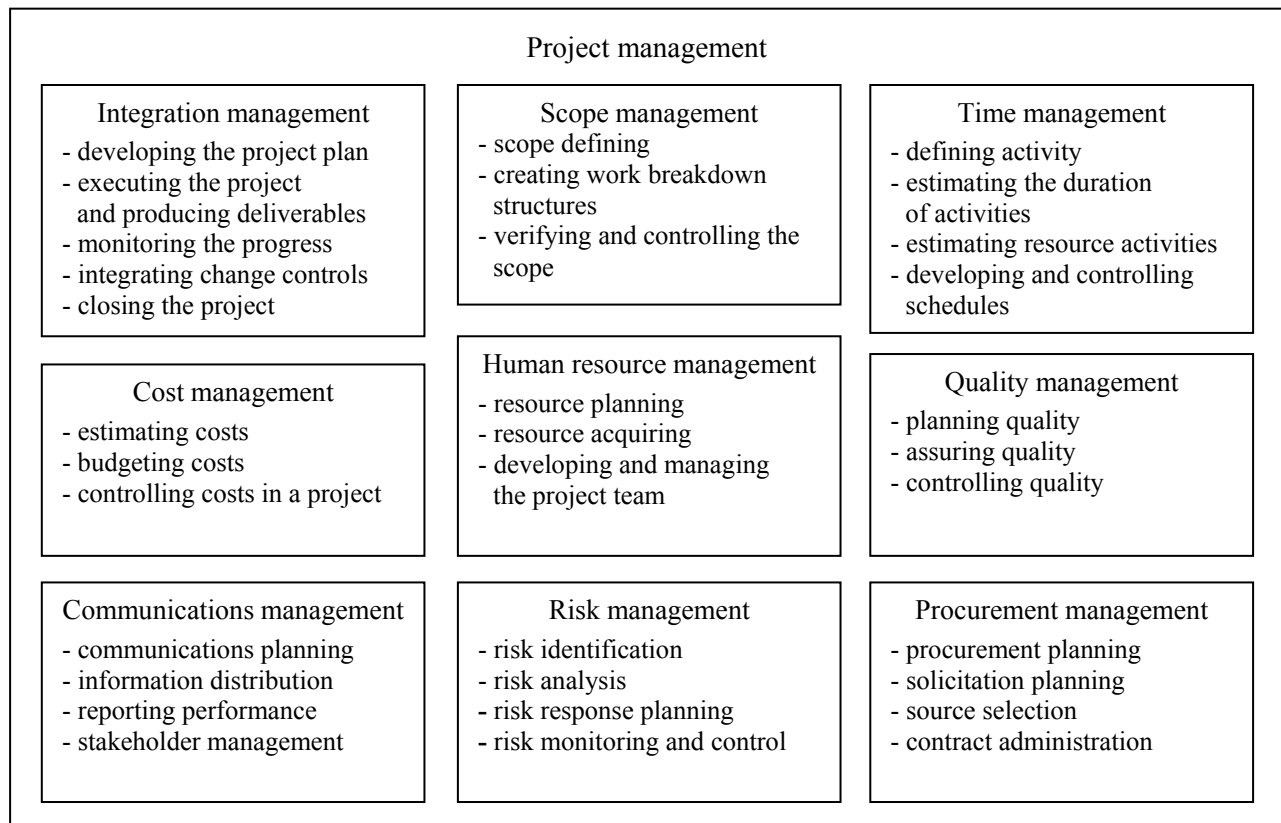


Figure 2. Description of project management
(source: self study with the use of [14])

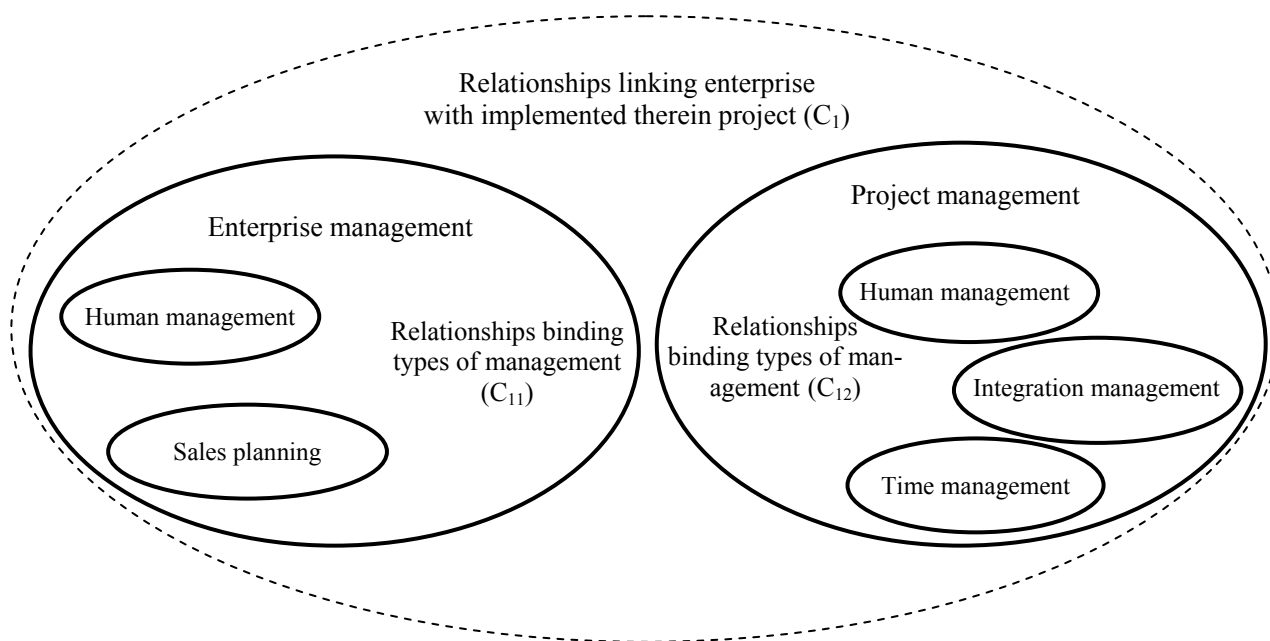


Figure 3. Hierarchical structure of reference model
(source: self study)

Both in enterprise and in project management, some functionalities occur. They may regard the same field, e.g. human resource management (both for the administrative divisions of enterprise, and for a project) as well as they may make a separate field, e.g. sales planning or project integration management.

An exemplary set of functionalities in enterprise and project management is presented in Fig. 3. The management fields are combined by constraints that are distinguished in Fig. 3 as C_1 . For instance, the number of employees of enterprise should be greater than demand connected with project implementation. Another example of constraint, that links both considered fields, may concern greater number of financial means in the enterprise than the expenditure of a project. The functionalities concerning enterprise and project management include common elements strictly regarding the management, what by C_{11} , C_{12} is distinguished in Fig. 3. For instance, human resource management encompasses planning, organizing, staffing, leading, motivation, and controlling. Application of these elements takes place in terms of enterprise activity fields (see Fig. 1). In this case, human resource management depends on, among other things, environment (e.g. circumstances on domestic and international market) or the kind of enterprise (different staffing policy in private and public organization). Implementation of other functionality concerning sales planning bases on, for example, the kind of enterprise

activity; in the case of production enterprise, the sales planning should take into account e.g. a workplace schedule.

In the same way, the relationships linking different functionalities in the project management may be described, what by C_{12} is distinguished in Fig. 3. In this case, the functionalities depend on management elements such as planning or controlling as well as these are determined by the kind of a project, for example, integration management or time management are different for software project and building project.

The enterprise model can be described by its resources. The project model is created from the requirements of the client. In these models, some parameters are determined, among which a set of decision variables and constraints may be distinguished. The constraints connect the variables that describe the capacity of the enterprise as well as the variables that concern the conditions of project implementation. For instance, the number of the enterprise's employees limits the duration of the project. This means that fulfilment of specified constraints enables project completion according to client requirements.

A way of model specification regards in general a declaration of the sets of decision variables, their domains, and constraints that imposed on subsets of variables. In this context, it seems natural to classify some decision problems as Constraints Satisfaction Problem (CSP).

A considered approach of specification, determined by constraints of reference model of decision problem, enables a simplified description of actuality, i.e. a description encompasses the assumptions of object, implementing therein tasks, and a set of routine queries (the instances of decision problems) that in framework of CSP are formulated.

It is assumed that the reference model for project prototyping problem has the structure of constraints satisfaction problem, and it may be described as follows:

$$\text{CSP} = ((V, D), C) \quad (1)$$

where:

$V = \{v_1, v_2, \dots, v_n\}$ – finite set of n variables,

$D = \{D_1, D_2, \dots, D_n\}$ – finite and discrete domains D of variables, where $D_i = \{d_{i1}, d_{i2}, \dots, d_{ir}\}$,

$C = \{c_1, c_2, \dots, c_m\}$ – finite set of m constraints limiting and linking decision variables.

Each constraint treated as a predicate can be seen as an n -ary relation defined by a Cartesian product $D_1 \times D_2 \times \dots \times D_n$. The solution to the CSP is a vector $(d_{1i}, d_{2k}, \dots, d_{nj})$ such that the entry assignments satisfy all the constraints C . So, the task is to find the values of variables satisfying all the constraints, i.e., a feasible valuation. Generally, the constraints can be expressed by arbitrary analytical and/or logical formulas as well as link variables with different non-numerical events.

Thus, a constraint can be treated as a logical relation among several variables, each one taking a value in a given (usually discrete) domain. To solve such a problem stated by the set of requirements (constraints) that specify a problem at hand, the concept of constraint programming (CP) is employed.

CP is an emergent software technology for declarative description CSP and can be considered as a pertinent framework for development of decision support system software aims. The main idea behind the CP concept is based on subsequent phases of constraint propagation and variable distribution [18].

The assumed model enables descriptive approach to the problem statement, encompasses constraint satisfaction problem structure and then allows implementation of the problem considered in the constraint programming environment. The idea behind the proposed approach assumes the system considered can be represented in terms of a knowledge base (KB).

KB comprises of facts and rules determining the system's properties and relations linking them respectively. Taking into account the concept of constraints propagation and variables distribution following from the constraint programming languages it is easy to note that any KB can be represented in a standard form of the CSP [22].

KB can be specified in terms of a system [5]. At the input of the system are the variables regarding the fundamental attributes of the object that are known and given by the user. In the considered KB for the enterprise-project model, there are, for example, variables concerning the amount of an enterprise's resources or the project structure. The output of the system is described by the attributes of the object that are unknown or are only partially known. In the considered case, there can be included variables regarding e.g. the cost or time of activity or usage of resources.

Classification of the decision variables in KB as input-output variables is arbitrarily made and permits to formulate two classes of standard queries, in a straight and in a reverse way, as follows [1, 4]:

- a straight way (i.e. corresponding to the question: what results from premises?), e.g. Does a given resources allocation guarantee the schedule does not exceed the given deadline?
- a reverse way (i.e. corresponding to the question: what implies conclusion?), e.g. What times of activity duration and number of resources guarantee the given schedule does not exceed the deadline?

The above-mentioned categories encompass the different reasoning perspectives, i.e. forward and backward reasoning. The corresponding queries can be stated in the same model that can be treated as composition of variables and constraints, i.e. assumed sets of variables and constraints limiting their values. In this context, the problem statement of project prototyping, which is specified in terms of CSP, are presented in next section.

3 Problem statement of project prototyping

Presented in Fig. 3, the hierarchical structure of reference model implies a similar structure concerning constraints satisfaction problem. The idea of this approach is presented in Fig. 4.

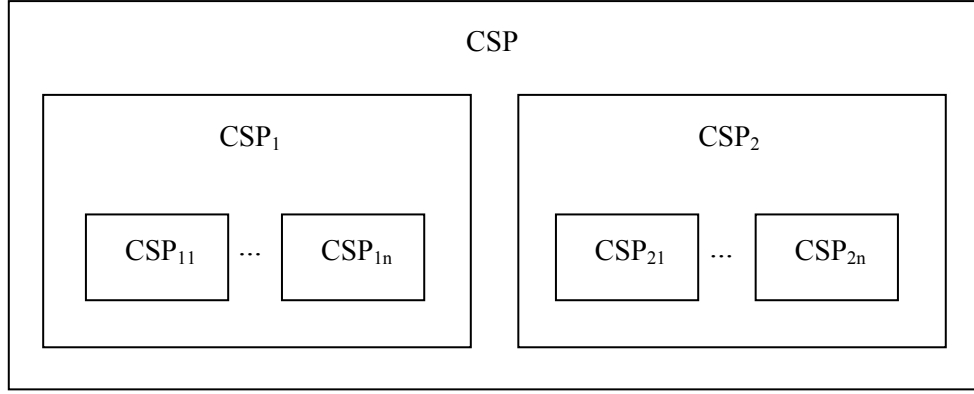


Figure 4. Structure of Constraints Satisfaction Problem
(source: self study)

The reference model in the form of single CSP can be described. The CSP consists of enterprise management (CSP_1) and project management (CSP_2). In turn, CSP_1 and CSP_2 contain other elements describing the functionalities in the considered areas (CSP_{11} , ..., CSP_{1n} and CSP_{21} , ..., CSP_{2n}).

Assuming two-level structure of reference model of project prototyping problem (RMPPP), it may be described as follows (see Fig. 3 - 4 and formula 1):

$$RMPPP = ((CSP_1, CSP_2), C) \quad (2)$$

CSP_1 regards the field of enterprise activity, CSP_2 – the field of project management, and C describes the constraints linking these fields in single Constraints Satisfaction Problem. This is defined following:

$$CSP_1 = ((\{R_1, R_2\}, \{D_{R1}, D_{R2}\}), C_{CSP1}) \quad (3)$$

where:

$R_1 = \{r1_{1,1}, \dots, r1_{m,h}, \dots, r1_{M,H}\}$ – a number of m -th financial resource (e.g. cash, deposits, short-term payments) in h -th time unit ($h = 0, 1, \dots, H$); given a set of resources $R = (R_1, \dots, R_k, \dots, R_z)$,

$R_2 = \{r2_{1,1}, \dots, r2_{n,h}, \dots, r2_{N,H}\}$ – a number of working hours for n -th group of employees (e.g. programmers) in h -th time unit,

D_{R1} – a set of admissible financial means R_1 , $r1_{m,h} \in D_{R1}$,

D_{R2} – a set of admissible working hours R_2 , $r2_{n,h} \in D_{R2}$; note that for the known values of decision variables the domain is a set with single element.

C_{CSP1} – a set of constraints:

- $C_{CSP1,1}$ – a number of available financial means in enterprise equals a amount of cash ($r1_{1,h}$), deposit

accounts ($r1_{2,h}$), short-term payments ($r1_{3,h}$) and bank loans ($r1_{4,h}$) in h -th time unit:

$$r1_{1,h} + r1_{2,h} + r1_{3,h} + r1_{4,h} = r1_h \quad (4)$$

- $C_{CSP1,2}$ – a number of working hours in enterprise is not greater than the sum of product of n -th employees group (analysts – $r2_{1,h}$, consultants – $r2_{2,h}$, programmers – $r2_{3,h}$) and a number of hours in a working day; it is assumed, a working day equals 8 hours plus available 2 hours of overtime during a project implementation:

$$10 * (r2_{1,h} + r2_{2,h} + r2_{3,h}) \leq r2_h \quad (5)$$

In case of project management area (CSP_2), a functionality concerning scheduling has been chosen. It is assumed, each project P_i consists of J activities: $P_i = \{A_{i,1}, \dots, A_{i,j}, \dots, A_{i,J}\}$. Moreover, it is assumed:

- each activity can be implemented by applying at least one of enterprise resources,
- indivisibility of activity,
- activity can start its execution only if required number of resources are available at the moments given by $Tp_{i,j}$ and after completed previous activity,
- each resource can be used by an activity only once,
- a number of resource used by an activity cannot be changed or allotted to other activity.

The resource can be allotted or released only after completion the activity that requires this resource. It is assumed, a number of available resources $r_{k,h}$ in h -th time unit is known. The planning horizon H contains a set of variables concerning the starting moments of the successive time units.

The field of project management can be described as follows:

$$CSP_2 = ((\{P_i, A_{i,l}, s_{i,j}, t_{i,j}, Tp_{i,j}, Tz_{i,j}, Dp_{i,j}\}, \{D_{Pi}, D_{Ai}, D_{Si}, D_{Ti}, D_{Tpi}, D_{Tzi}, D_{Dpi}\}), C_{CSP2}) \quad (6)$$

where:

P_i – i-th project,

$A_{i,j}$ – j-th activity of i-th project that is specified following: $A_{i,j} = (s_{i,j}, t_{i,j}, Tp_{i,j}, Tz_{i,j}, Dp_{i,j})$,

$s_{i,j}$ – the starting time of the activity $A_{i,j}$, i.e., the time counted from the beginning of the time horizon H ,

$t_{i,j}$ – the duration of the activity $A_{i,j}$,

$Tp_{i,j} = (tp_{i,j,1}, tp_{i,j,2}, \dots, tp_{i,j,z})$ – the sequence of allocation moments by the activity $A_{i,j}$ required the resources: $tp_{i,j,k}$ – the time counted since the moment $s_{i,j}$ of a number $dp_{i,j,k}$ of the k-th resource allocation to the activity $A_{i,j}$; that means a resource is allotted to an activity during its execution period: $0 \leq tp_{i,j,k} < t_{i,j}$; $k = 1, 2, \dots, z$,

$Tz_{i,j} = (tz_{i,j,1}, tz_{i,j,2}, \dots, tz_{i,j,z})$ – the sequence of moments, when the activity $A_{i,j}$ releases the resources: $tz_{i,j,k}$ – the time counted since the moment $s_{i,j}$ of a number $dp_{i,j,k}$ of the k-th resource release by the activity $A_{i,j}$; that means a resource is released by activity during its execution period: $0 < tz_{i,j,k} \leq t_{i,j}$; $tp_{i,j,k} < tz_{i,j,k}$; $k = 1, 2, \dots, z$,

$Dp_{i,j} = (dp_{i,j,1}, dp_{i,j,2}, \dots, dp_{i,j,z})$ – the sequence of number of the k-th resource is allocated to the activity $A_{i,j}$; $dp_{i,j,k}$ – a number of the k-th resource allocation to the activity $A_{i,j}$; that assumes: $0 \leq dp_{i,j,k} < R_k$; $k = 1, 2, \dots, z$;

D_{Pi} – a set of admissible number of projects in the enterprise,

D_{Ai} – a set of admissible number of activities in i-th project,

D_{Si} – a set of admissible starting times of activity $A_{i,j}$ in i-th project,

D_{Ti} – a set of admissible duration of activity $A_{i,j}$ in i-th project,

D_{Tpi} – a set of admissible allocation moments to activity $A_{i,j}$ for k-th resource in amount of $dp_{i,j,k}$, in i-th project,

D_{Tzi} – a set of admissible release moments by activity $A_{i,j}$ for k-th resource in amount of $dp_{i,j,k}$, in i-th project,

D_{Dpi} – a set of admissible number of required resources by the activity $A_{i,j}$ in i-th project,

C_{CSP2} – a set of constraints:

- $C_{CSP2,1}$ – constraint concerning horizon of project completion $H = \{0, 1, \dots, h\}$:

$$\forall s_{i,j} \forall t_{i,j} (s_{i,j} + t_{i,j} \leq H) \quad (7)$$

- $C_{CSP2,2}$ – order constraints:

- the k-th activity follows the i-th one:

$$s_{i,j} + t_{i,j} \leq s_{i,k} \quad (8)$$

- the k-th activity follows other activities:

$$\begin{aligned} s_{i,j} + t_{i,j} &\leq s_{i,k} \\ s_{i,j+1} + t_{i,j+1} &\leq s_{i,k} \end{aligned} \quad (9)$$

...

$$s_{i,j+n} + t_{i,j+n} \leq s_{i,k}$$

- the k-th activity is followed by other activities:

$$\begin{aligned} s_{i,k} + t_{i,k} &\leq s_{i,j} \\ s_{i,k} + t_{i,k} &\leq s_{i,j+1} \end{aligned} \quad (10)$$

...

$$s_{i,k} + t_{i,k} \leq s_{i,j+n}$$

In the reference model of project prototyping, the constraints C are the elements linking CSP_1 and CSP_2 (see formula 2).

These constraints contain:

C_1 – the financial means for i-th project $r1_{h,i}$ cannot be greater than total value of admissible in the enterprise financial means $r1_h$ in h-th time unit:

$$r1_{h,i} \leq r1_h \quad (11)$$

C_2 – an admissible number of working hours $r2_{h,i}$ for i-th project and n-th group of employees cannot be greater than total number of admissible in the enterprise working hours $r2_h$ in h-th time unit:

$$r2_{n,h,i} \leq r2_h \quad (12)$$

It is assumed, for each i-th project there are l alternative variants its implementation $P_{l,i}$. Alternative variant is understood as project, which parameters concerning time, cost or scope are different from the parameters of original project.

The routine queries can be formulated in the straight and reverse way for considered RMPPP. In case of the straightway, the considered problem regards the answer to the following question: is there a schedule meeting constraints for given values of variables, and if so, what are its parameters?

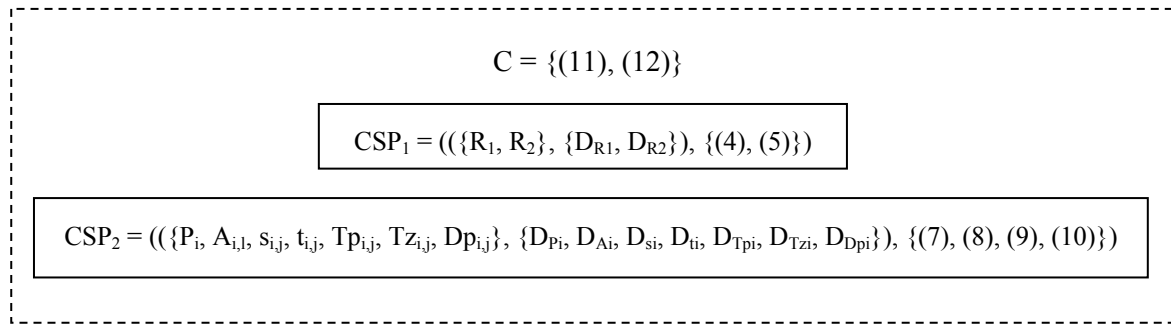


Figure 5. Reference model for declarative statement of project prototyping project
(source: *self study*)

This question can be expanded to the next, for instance, does a given schedule not exceed the given deadline H , financial resources $r1$ and working-hours $r2$ in time unit h ? It allows a class of multicriteria problems to be taken into consideration.

If for the straight way there is no schedule, then is assumed that original project execution is at risk of failure. Thus, a question concerning the reverse way can

be formulated: what values of variables guarantee the completion of the project by given constraints? The choice of variables, which values are changed according to assumed constraints, depends on the considered problem, and in an arbitrary way is determined.

The method concerning the determination of admissible solutions for the above-described problem in terms of cost estimating is presented in next section.

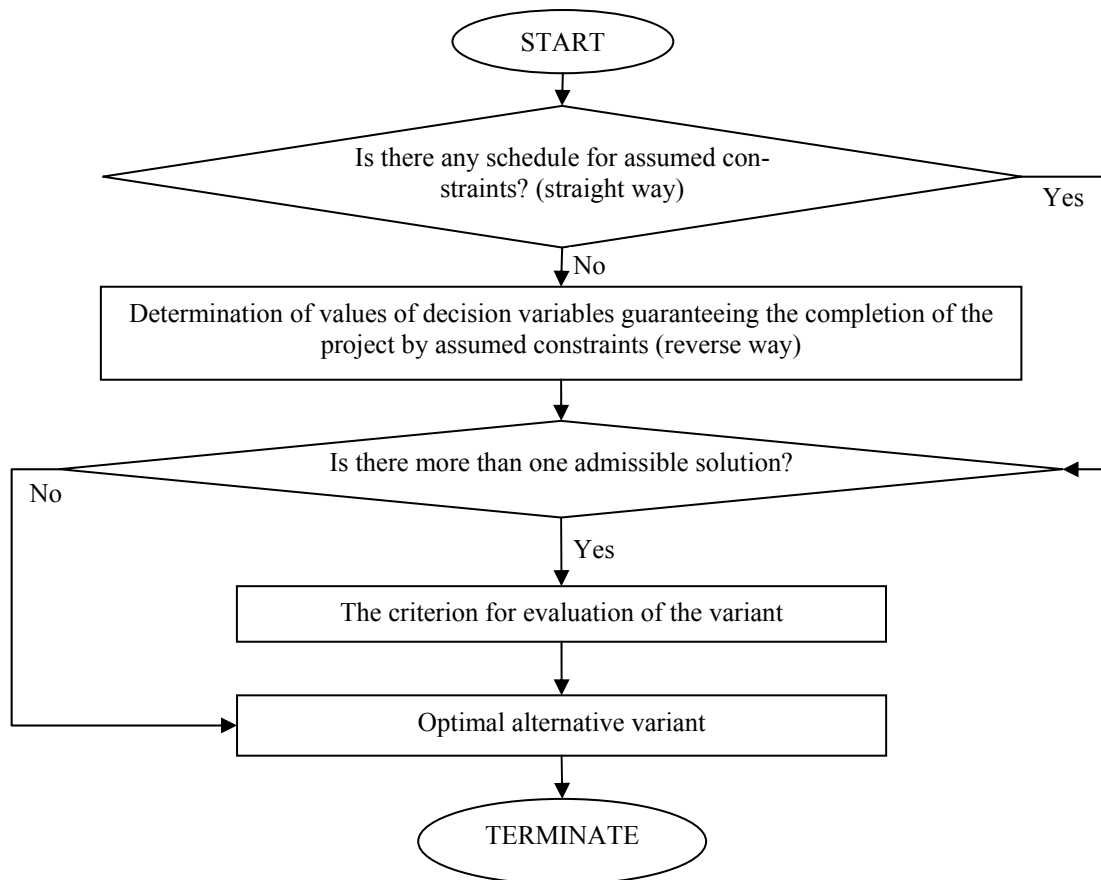


Figure 6. Project prototyping procedure for projects at risk of failure
(source: *self study*)

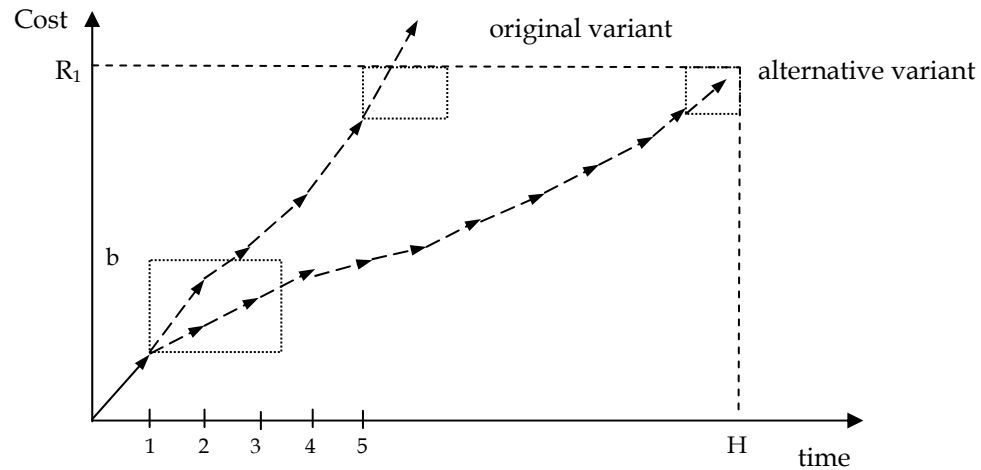


Figure 7. Example of project prototyping
(source: self study)

4 Method of obtaining feasible solutions of project

The planning issue and then the successive monitoring of the project, is one of the most important elements of project management that determines its success or failure [9].

So, there is a need to develop method that will enable an early detection of discrepancies in a project execution. Moreover, the method should determine the alternative variants that meet the goal of the project and avoid the estimated discrepancies.

The stages of proposed method is presented in Fig. 6. If for the assumed constraints there is no schedule (e.g. estimated cost of project is greater than available financial means), then with using of reverse way, the values of decision variables, which guarantee a project execution, are determined.

The functionality concerning cost estimation is chosen as an example illustrating the idea of the proposed, implemented as RMPPP, approach. Exemplary alternative variant of project is presented in Fig. 7. It is determined if cost estimation indicates a lack of possibility for original project implementation by assumed constraints. The cost estimation is an additional constraint, into RMPPP is added and in form of CSP_{2n} is described.

It is assumed, an assessment of activity completion, as well as a redetermination of an admissible solutions set follows in time unit h . Trajectory of project execution, distinguished by solid line, indicates cost of completed activities as well as activities in progress in first

time unit. Approximating the cost function can be set its values in next time units. This is presented in Fig. 7 by dashed line. An interval belongs to a set of admissible solutions that in form of rectangle is distinguished in Fig. 7. The size of set is connected with the domains of variables and the assumed constraints. The a interval depends on the order constraints between activities and project time horizon, so in result on slack time. The b interval depends on constraint concerning the financial means r_1 in time unit h .

If cost estimate is greater than assumed financial constraint (original variant), then alternative variant is sought that fulfils the assumed time (H) and financial (R_1) constraints. If a set of admissible solutions is multi-elements, then the variants according to assumed criterion are assessed. The criterion may regard, for instance, minimisation of time or cost of the project execution. An example of the assessment is presented in Fig. 8.

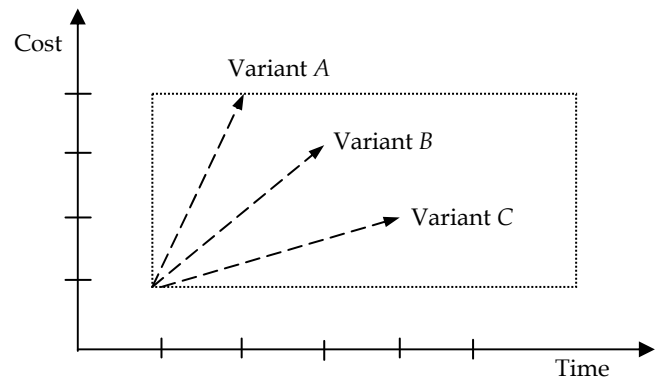


Figure 8. Example of variants evaluation
(source: self study)

The choice of a variant of the three depends on an assumed by decision-maker criterion. If criterion concerns the minimisation of time, then variant A is most profitable. In case of cost minimisation, variant C is most advantageous. For combined criterion, i.e. minimisation of time and cost with equal weights – all variants are equally advantageous.

The advantages of proposed approach contain an obtaining of admissible solutions set in time unit h . In case of predictable difficulties with project execution, the approach enables a choice of alternative variant and in general the preventing to exceed the assumed constraints. This also refers to inability of some activities execution, and searching a possibility of project completion in another form. An example concerning the described approach is presented in next section.

5 Illustrative examples

The example aims to illustrate a possibility of CSP specification for decision problem of project prototyping. Problem in the straight and in the reverse way is formulated.

5.1. Routine queries formulated in the straight way

Example 1

The project concerns a software implementation in sales field of trading company. The orderer has got software in sales, but it contains limited functionality and the integrity with other domains of software is not ensured. The required additional features include making offer, registration of order, assignment the trade credit to customer, analysis of customer loyalty (frequency of sales and payments), and assignment a few payment terms.

The project contains seven activities:

- 1) analysis of business processes, IT systems, database structure in the client company,
- 2) new software installing, initial configuration and testing,
- 3) customisation of standard software setting according to the client requirements,
- 4) customisation according to the untypical client requirements and building interfaces to link software concerning different fields of enterprise activity,
- 5) formulating a way of data migration, from previous software database to new one,

- 6) final configuration and testing software,
- 7) users training.

The activity network diagram for considered project $P = \{A_1, \dots, A_7\}$ is presented in Fig. 9.

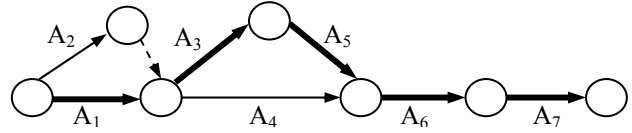


Figure 9. Activity network of the project
(source: self study)

Operation times (in working hours) for the project are determined by using past experiences as follows: $T = (16, 8, 16, 30, 16, 16, 60)$. In Fig. 9, the bolded arrows indicate the critical path with total time equals 124 working hours.

The software company can allocate for the project three employees: one programmer and two consultants. Programmer can work by activities A_2 - A_6 , in turn consultants - activities A_1 and A_7 . Thus, a number of resource $dp_{1,j}$ for activity j takes the form of the following sequence: $Dp_1 = (2, 1, 1, 1, 1, 1, 2)$. It is assumed that consultants can work independently and parallel by activity A_1 and A_7 . In this case, all activities are critical, and total time equals 116 working hours.

The analysis of the past completed projects, which belong to the same class as considered project, indicates the linear relationship between cost and activity duration: $dp_{2,j} = 1 + 0,5 \cdot t_j$. This relation consists fixed cost (e.g. cost of stay by client), and variable cost (hourly rate). Thus, a number of financial resources $dp_{2,j}$ allocated to the activity j is in the following form: $Dp_2 = (9, 5, 9, 16, 9, 9, 31)$. Total planned cost of project equals 88 monetary units (m.u.). A whole number of the resources at the starting moment of activity is allocated, and only at the moment of its completion can be released.

The order constraints according to the activity network of the project and formulas (8), (9), and (10) are following:

$$\begin{aligned} C_1: s_3 &\geq s_1 + t_1, C_2: s_3 \geq s_2 + t_2, C_3: s_4 \geq s_1 + t_1 \\ C_4: s_4 &\geq s_2 + t_2, C_5: s_5 \geq s_3 + t_3, C_6: s_6 \geq s_4 + t_4 \\ C_7: s_6 &\geq s_5 + t_5, C_8: s_7 \geq s_6 + t_6 \end{aligned}$$

Client sets the project completion at three weeks time (120 working hours – time horizon $H = \{0, 1, \dots, 120\}$), by budget equals 100 m.u.

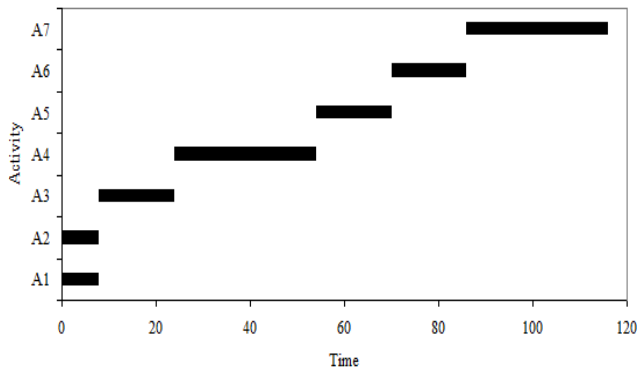


Figure 10. Gantt's chart of project
(source: self study)

The considered problem belongs to the class of “straight” ones where for a given parameters describing the enterprise-project system the activities schedule is sought. It reduces to the following question: is there, and if so, what form does a schedule have that completion time does not exceed the deadline H , and that fulfils the resource constraints? Note that the answer to above-mentioned question is connected with determination of the starting time of the activity s_j , where $0 \leq s_j < 120$; $j = 1, 2, \dots, 7$.

The CSP-based reference model has been implemented in Oz Mozart [19]. Obtaining of the solutions took less than a second (the AMD Turion(tm) II Ultra Dual-Core M600 2,40GHz, RAM 1,75 GB platform has been used). The first admissible solution has the following form: $S = (0, 0, 8, 24, 54, 70, 86)$. The project schedule fulfilled all constraints imposed by an enterprise capability and project requirements, is presented in Fig. 10.

The usage level of financial means in the time horizon is illustrated in Fig. 11.

Example 2

After completion of the A_1 and A_2 activities, the relationship between cost and duration of activity is again determined: $dp_{2,j} = 2 + 0,6 \cdot t_j$.

The sequence of the financial means can be described following: $Dp_2 = (9; 5; 11,6; 20; 11,6; 11,6; 38)$. Thus, total cost of the project equals 106,8 m.u. Other values of decision variables, their domains, as well as the constraints are the same as in Example 1.

The considered problem also belongs to the class of “straight” ones, and it can be reduced to the following question: is there, and if so, what form does a schedule have that completion time does not exceed the deadline H , and that fulfils the resource constraints?

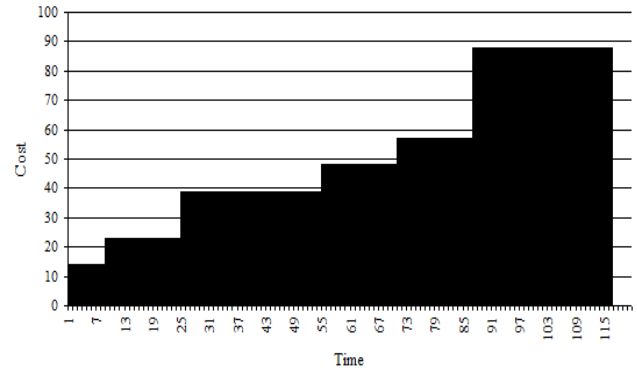


Figure 11. Gantt's-like chart of the resource usage
(source: self study)

Similarly to the previous case the solution to the problem concerns the determination of the moments activities start their execution s_j . The planned cost of the project (106,8 m.u.) exceeds the budget of the project (100 m.u.). In this case, the set of admissible solutions is empty. That means there is no schedule. Thus, there is still a possibility to reformulate the considered problem by stating it in a reverse way, i.e. the way aimed at searching for decision variables (e.g. duration of activity) guaranteeing that the completion time of the considered project will not exceed the assumed deadline H . Such case is considered in next subsection.

5.2. Routine queries formulated in the reverse way

Assumed the same activity network, time horizon, domains of decision variables, and the constraints as in previous subsection, for straight way. Taking into account the kind of the considered software project, it is assumed that planned time of last activity (users training) is changed. This activity is connected with many factors (e.g. user's perception, education, past experiences) that hinder an accurate estimation of the activity. Assumed the minimal duration of the A_7 activity equals 30 working hours.

The considered problem can be reduced to the question: what duration of the A_7 activity guarantees that completion time of the project does not exceed the deadline H , and the resources constraints?

In order to response to this question the values of the activity duration t_7 and its cost $dp_{2,7}$ are sought. The admissible solutions is as follows $t_7 = \{30, \dots, 48\}$.

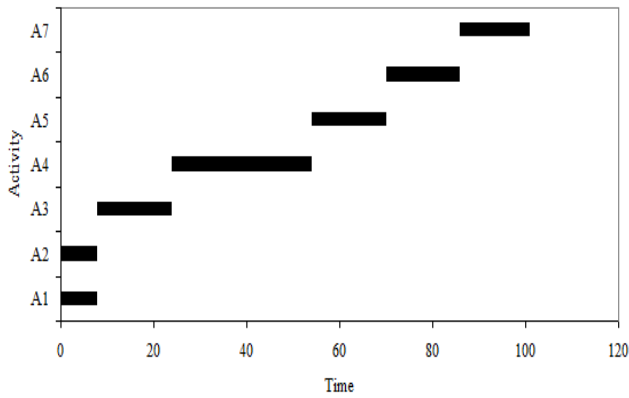


Figure 12. Gantt's chart of project according to criterion of time minimisation
(source: self study)

For assumed constraints and discrete values, 450 sequences of the S (the starting time of the activities) are determined. The obtained variants can be evaluated according to such criteria as time, cost or a number of required employees. If criterion concerns the minimisation of time or cost, then the optimal variant is for $t_7 = 30$ working hours. Thus, total time equals 101 working hours, and cost – 88,8 m.u. Gantt's chart of project and usage of financial means for first admissible solution is presented in Fig. 12 and 13.

If criterion regards minimisation of required employees (programmer and consultants), then the solutions for $t_7 = \{30, \dots, 34\}$ are equally advantageous, because a slack time for the project equals 4 working hours. Gantt's chart of project and usage of financial means for last admissible solution ($t_7 = 34$) is presented in Fig. 14 and 15. The assumed domains of decision variables and constraints determine the possible values of sought parameters.

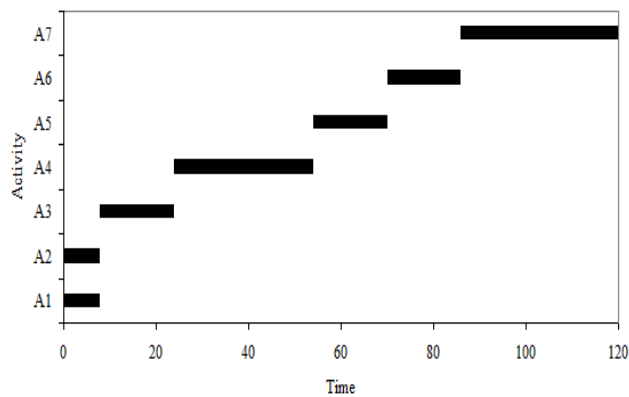


Figure 14. Gantt's chart of project according to criterion of employee's minimisation
(source: self study)

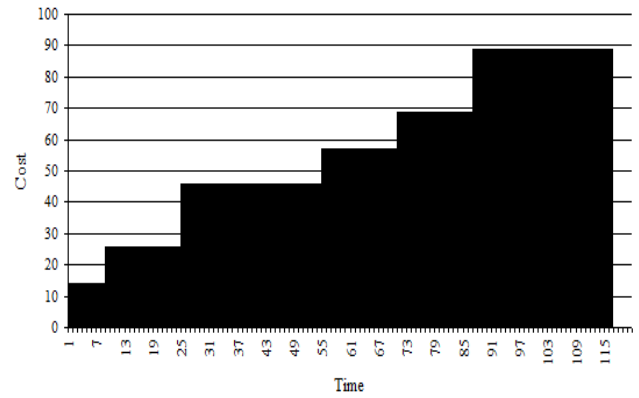


Figure 13. Gantt's-like chart of the resource usage according to criterion of time minimisation
(source: self study)

The result is a set of feasible solutions in time unit h. Note that the number of generated solutions depends not only on the knowledge base, but also on a user-declared granularity of solutions in constraint programming languages such as, for instance, ILOG or Oz Mozart [19].

6 Conclusions

In the present, changeable business environment, the quickness of response to customer needs or pressure on innovation and the effective cost management determine the success or failure in the struggle for market position. This forces more frequent and larger-scale changes in contemporary organizations. The answer to these new challenges is the application of the principles of project management.

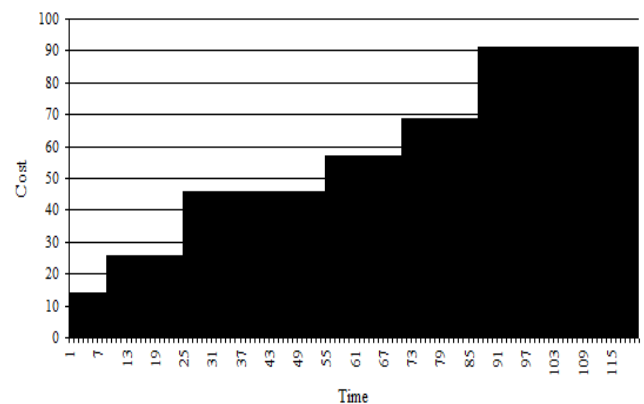


Figure 15. Gantt's-like chart of the resource usage according to criterion of employee's minimisation
(source: self study)

In case of projects carried out on a client order, erroneous estimation of expenditures and project deadlines may result penalties being accrued, as agreed upon in the contract or covering the costs with the company's own money. A wrong decision may worsen the liquidity of an enterprise or even lead to its bankruptcy. In this situation, it seems extremely important to support the decision maker.

The proposed approach assumes a kind of reference model encompassing open structure enabling to take into account different sorts of variables and constraints as well as to formulate straight and reverse kinds of project planning problems.

Since a constraint can be treated as a logical relation among several variables, each one taking a value in a given (usually discrete) domain, the idea of CP is to solve problems by stating the requirements (constraints) that specify a problem at hand, and then finding a solution satisfying all the constraints. Because of its declarative nature, it is particularly useful for applications where it is enough to state what has to be solved instead of how to solve it [1].

The advantages of the proposed approach include the possibility of the description of enterprise and project management in terms of single knowledge base. Moreover, in the presented approach it is possible to obtain a set of feasible solutions in the different phases of the project life cycle. This is especially attractive in the absence of the possibility of continuing the project in its original form and can support the decision maker in obtaining the alternative variants of the project.

Further research focuses on the presentation of the model reference for the project prototyping problem, when some activity cannot be completed. It should also include a comparison of the proposed approach to another approach concerning considered field. Moreover, the further research can be aimed at carrying out verification of the knowledge base of described object.

7 References

- [1] Banaszak Z., Bocewicz G., Bach I. - *CP-driven production process planning in multiproject environment* [in] Decision Making in Manufacturing and Services, Vol. 2, No.1-2, 2008, pp. 5-32.
- [2] Belassi W., Tukul O.I. - *A new framework for determining critical success/failure factors in projects* [in] International Journal of Project Management, Vol. 14, 1996, pp. 141-151.
- [3] Bertalanffy L. - *General system theory - A Critical Review* [in] General Systems, pp. 1-20, Vol. 7, 1962.
- [4] Bocewicz G., Bach-Dąbrowska I., Banaszak Z. - *Declarative approach to computer aided project planning systems design*. Akademicka Oficyna Wydawnicza EXIT, Warszawa 2009.
- [5] Bubnicki Z. - *Logic-algebraic method for knowledge-based relation systems* [in] Systems Analysis Modelling and Simulation, Vol. 33, 1998, pp. 21-35.
- [6] Cooke-Davies T.J., Arzymanow A. - *The maturity of project management in different industries: an investigation into variations between project management models* [in] International Journal of Project Management, Vol. 21, 2003, pp. 471-478.
- [7] Jorgensen M., Sjoberg D.I. - *The impact of customer expectation on software development effort estimates* [in] International Journal of Project Management, Vol. 22, 2004, pp. 317-325.
- [8] Jorgensen M., Gruschke T. - *The impact of lessons-learned sessions on effort estimation and uncertainty assessments* [in] IEEE Transactions on Software Engineering, Vol. 35, No. 3, 2009, pp. 368-383.
- [9] Kerzner H. - *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. Tenth Edition, John Wiley and Sons, New York 2009.
- [10] Koenig D. - *Project Rescuing* [in] PM Network, No. 1, 2006, pp. 64-68.
- [11] Meredith J.R., Mantel S.J. - *Project Management - a managerial approach*. Third Edition, John Wiley and Sons, New York 1995.
- [12] Molokken-Ostfold K., Jorgensen M. - *A comparison of software project overruns* [in] IEEE Transactions on Software Engineering, Vol. 31, No. 9, 2005, pp. 754-766.
- [13] Nitithamyong P., Skibniewski M.J. - *Success/failure factors and performance measures of web-based construction project management systems: professionals' viewpoint* [in] Journal of Construction Engineering and Management, Vol. 132, 2006, pp. 80-87.

- [14] Project Management Institute - *A Guide to the Project Management Body of Knowledge*. Fourth Edition, Paperback PMI, PMBOK Books, Newton Squire 2008.
- [15] Rakitin S. - *Creating accurate estimates and realistic schedules* [in] SQP, Vol. 4, No. 2, 2002, pp. 30-36.
- [16] Reichelt K., Lyneis J. - *The dynamics of project performance: benchmarking the drivers of cost and schedule overrun* [in] European Management Journal, Vol. 17, 1999, pp. 135-150.
- [17] Robertson S., Williams T. - *Understanding project failure: using cognitive mapping in an insurance project* [in] Project Management Journal, Vol. 37, 2006, pp. 55-71.
- [18] Rossi F. - *Constraint (logic) programming: A survey on research and applications* [in] New Trends in Constraints (Apt K.R. et al.), Springer-Verlag, Berlin 2000, pp. 40-74.
- [19] Schutle H., Smolka G., Wurtz J. - *Finite Domain Constraint Programming in Oz*. German Research Center for Artificial Intelligence, Saarbrücken 1998.
- [20] Shore B. - *Systematic biases and culture in project failures* [in] Project Management Journal, Vol. 39, 2008, pp. 5-16.
- [21] Singh R., Keil M., Kasi V. - *Identifying and overcoming the challenges of implementing a project management office* [in] European Journal of Information Systems, Vol. 18, 2009, pp. 409-427.
- [22] Van Roy P., Haridi S. - *Concepts, techniques and models of computer programming*. Helion, Gliwice 2005.
- [23] Yeo K.T. - *Critical failure factors in information system projects* [in] International Journal of Project Management, Vol. 20, 2002, pp. 241-246.
- [24] Yourdon E. - *Death march: The complete software developer's guide to surviving 'mission impossible' projects*. Englewood Cliffs. Prentice-Hall, New York 1997.