Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Knowledge Based Approach to Project Prototyping

Izabela Tomczuk-Piróg*, Peter Nielsen*, Wojciech Muszyński, Zbigniew Banaszak*****

** Department of Production, Aalborg University,*
Fibigerstraede 16, 9220 Aalborg Oest, Denmark (e-mail: i.tomczuk@po.opole.pl, peter@production.aau.dk)
*** Wrocław University of Technology, Institute of Computer Science, Automation and Robotics,*
50-372 Wrocław, Poland,(e-mail: wojciech.muszynski@pwr.wroc.pl)
**** Technical University of Koszalin, Department of Computer Science and Management,*
Śniadeckich 2, 75-453 Koszalin, Poland, (e-mail: banaszak@tu.koszalin.pl)

Abstract: Decision making supported by task-oriented software tools plays a pivotal role in modern enterprises, because commercially available ERP systems are unable to respond in an interactive on-line/real-time mode. It opens up for a new generation of decision support system (DSS) that enable a fast prototyping of production flows in multi-project environment as well as integrating approaches to project execution evaluation. In that context our goal is to provide a knowledge base approach allowing one to be independent of context or representation data as well as allowing for the design of an interactive and task-oriented DSS. The assumed knowledge base mode of specifying a production system leads to solving a logic-algebraic method (LAM) decision problem. The results obtained are implemented in a software package supporting project management in SMEs. Illustrative example of the ILOG-based software application is provided.

1. INTRODUCTION

It is a tendency in modern trade that production is connected with the customers' requirements. Quick appreciation of the market's needs or fast reaction to the needs, is one of the critical factors giving firm's good position on the market (Banaszak Z., Zaremba M., 2006). The organizations themselves, customizations of products, technological development, shorter product life cycles are all the cause of the changes which has occurred within recent years. The basis of the enterprise's activity are work orders, frequently handled as projects (Tomczuk-Piróg I., Muszyński W., Bocewicz G., 2006).

Taking on new production projects requires task planning in the production system. The tasks includes establishing final products production programmes, planning of positions (machinery) and determining surplus necessary for the execution of the plans (Banaszak Z., Zaremba M., 2006). The problems regarding allocation under limited resources (companies operate under limited resources and customers requirements) belong to a class of NP-hard problems.

Most of the publications on project management have been dedicated to planning tasks within a single project. In recent years there has however been a growing interest in problems related to project scheduling in a multi-project environment. A dominant criterion in single-project problem is the satisfaction of time constraints (Banaszak Z., Zaremba M., 2006). Scheduling of multiple projects with shared constrained resources has to take into account other criteria such as idle resources, resource levelling, in process inventory and projects splitting (Lova A. et al. 2000).

Available software applications, supporting project management processes, facilitate searching for possible solutions for the production system capacity constraints (e.g. availability of production resources, transportation means and warehouses capacity) and the customer requirements (e.g. the work order execution deadline). They facilitate defining conditions which guarantee calculation efficiency of a given procedure; however, they do not guarantee obtaining optimum solutions. As a result a set of alternative possible schedules is obtained. Project planning systems available in the market support the planning expert in generating production plans; they do however not offer the possibility to evaluate plans on subjective company requirements concerning e.g. production programme execution deadline, resources charge distribution, processes execution cost (Wójcik R., Tomczuk-Piróg I., Banaszak Z., 2007).

There is, therefore, a need to develop and implement methods and tools facilitating both fast generation of project execution plans and a multi-criterion evaluation of choice decision on the basis of expert knowledge, included directly in the decision system.

When searching for solutions within venture portfolios, a variety of solutions for the allocation of alternative resources adhering to the assumptions can be found. Solving allocation problems in constraint conditions is a complex calculative problem, therefore a search for solutions should take place only if we are certain that at least one exists. It means that conditions should be known which, if met, guarantee obtaining solutions in the search space.

The standard methods, based on mathematical programming concepts (leading to the combinatorial explosion of the space of potential solutions), are not efficient. In turn, heuristic paradigms based approaches, e.g. taboo search, simulated annealing, and genetic algorithm, will provide suboptimal solutions. It should however be noted, that in both abovementioned cases it is assumed the searched space consists admissible solutions. In many cases, such guarantees do not hold, e.g. a class of timetabling problems. Of course, in many real-life problems focusing on equilibrium between capabilities possessed by an enterprise and requirements imposed by work orders considered, such as guarantee does not exist. It means there is no guarantee the time consuming searching process will end with any valid result.

The idea standing behind of the approach proposed assumes the system considered can be represented in terms of a Knowledge Base (*KB*) (Dubois D., Fargier H., Fortemps P., 2003). Taking into account a 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 so called, Constraint Satisfaction Problem (CSP) (Van Roy P., Haridi S., 2005). So, the main problem regarding of the guarantee of the existence of an admissible can be seen as a problem of *KB* consistency checking, which in turn can be seen as the solution of the relevant CSP. It is assumed that *KB* is specified in terms of so-called logic-algebraic method (LAM) (Bubnicki Z., 1998), which allows one to specify problems through a set of logic propositions. This gives the possibility to determine whether sufficient conditions guaranteeing the existence of solutions for decision problems considered are present. The inference engine applied in the LAM is then easily implemented in a kind of constraint programming/constraint logic programming (CP/CLP) language (Wójcik R., Tomczuk-Piróg I., Banaszak Z., 2007).

Chosen representation allows to omit disadvantages following from typically applied methods as those using production-rules (time consuming inference process) and Horn's clauses (limiting expert's knowledge representation) formalisms.

The problem under analysis deals in its first stage with knowledge management, understood as execution of three tasks: monitoring (knowledge base verification, taking into account appropriate conditions and relations), planning (prototyping of sufficient conditions) and control (time efficient solutions searching strategies).

The approach proposed complements the decision system with an additional module (evaluation module) and facilitates searching for possible solutions meeting company production programme execution evaluation criteria. This facilitates selection of the generated schedules directly in the system, in accordance with unified arbitrary evaluation criteria. This gives the possibility to generate the "best" solution with regard to the subjective company requirements, to save time as well as to reduce the production programme preparation cost, with uninterrupted execution of all incoming orders, while at the same time optimizing the production process as far as company specific criteria are concerned.

The paper is organized as follows. Assumptions concerning the considered class of systems and the main problem of the paper are formulated in section 2. The considered decision problem consists in determination of triples (production and transportation operations commencement times and priority rules determining the projects execution sequence) guaranteeing a given projects portfolio is completed in arbitrarily assumed period of time. In section 3 an introduction to logic-algebraic method (LAM) is provided, and following this its implementation to the knowledge generation and a decision problem resolution is presented. Furthermore the concept of a constraint satisfaction problem is introduced, and then implemented to a knowledge base specification. An illustrative example of the approach provided to the projects portfolio prototyping and approach to imprecise data handling is shown in section 4. Finally conclusions and future research are presented in section 5.

2. PROBLEM STATEMENT

Given is a production system for the executed project portfolio. Production system (PR) includes: PR=({ R_i | i=1,...,r}, { T_i | i=1,...,s}, { L_i | i=1,...,p}), determining a set of production, transportation and human resources where: R_i - i-th production resource, T_i - i- th transportation resource, L_i - i-ty th human resource.

Given is a set of projects: $P = \{P_1, P_2, P_3, \ldots, P_q\}$. Each project is a sequence of a finite number of operations, where: $q_k = \left(AP^k_{i,j}, AT^k_{i,j}\right)$ – is a sequence of production and transportation operations executed on resources in k-th project. Production and transportation operations commencement time vectors are also defined:

$$ST^k_{i,j} = (ST^1_{1,1}, ST^1_{1,2}, \ldots, ST^1_{1,n}, ST^1_{2,1}, \ldots, ST^1_{m,n}, \ldots, ST^r_{m,n}),$$

where $ST^k_{i,j}$ – i-th transportation operation commencement time at j-th resource in k-th project,

$SP^k_{i,j} = (SP^1_{1,1}, SP^1_{1,2}, \ldots, SP^1_{1,n}, SP^1_{2,1}, \ldots, SP^1_{m,n}, \ldots, SP^r_{m,n})$, where – i-th production operation commencement time at j-th resource in k-th project,

$SP^k_{0,j}$ – the first production operation commencement time at j-th resource in k-th project,

$ST^k_{0,j}$ – the first transportation operation commencement time at j-th resource in k-th project,

$Q_k = (Q_1, Q_2, \ldots, Q_p)$ – priority rules, determining project execution sequence.

Production and transportation operation commencement times at j-th resource in k-th project have been presented in the following matrixes:

$$SP_{i,j}^{k}=\begin{bmatrix} SP_{1,1}^{k} & SP_{1,2}^{k} & \cdots & SP_{1,n}^{k} \\ \vdots & & & \vdots \\ SP_{m,1}^{k} & SP_{m,2}^{k} & \cdots & SP_{m,n}^{k} \end{bmatrix}$$

$$ST_{i,j}^{k}=\begin{bmatrix} ST_{1,1}^{k} & ST_{1,2}^{k} & \cdots & ST_{1,n}^{k} \\ \vdots & & & \vdots \\ ST_{m,1}^{k} & ST_{m,2}^{k} & \cdots & ST_{m,n}^{k} \end{bmatrix}$$

Operation times of individual operations $T_{A^{k}_{i,j}}$ are described in the following way:

$$T_{A^{k}_{i,j}}=\begin{bmatrix} T_{A^{k}_{1,1}} & T_{A^{k}_{1,2}} & \cdots & T_{A^{k}_{1,n}} \\ \vdots & & & \vdots \\ T_{A^{k}_{m,1}} & T_{A^{k}_{m,2}} & \cdots & T_{A^{k}_{m,n}} \end{bmatrix}$$

Operation assignment to a resource in the k-th project is determined in accordance with the relation:

$$\begin{cases} 1 - \text{if operation } A_j \text{ is executed at j-th resource in the k-th project} \\ 0 - \text{otheKBise} \end{cases}$$

Operations executed in the k-th project matrix are as follows:

$$\begin{bmatrix} A_{1,1}^{k} & A_{2,2}^{k} & \cdots & A_{1,n}^{k} \\ \vdots & \vdots & & \vdots \\ A_{m,1}^{k} & A_{m,2}^{k} & \cdots & A_{m,n}^{k} \end{bmatrix}$$

For such a defined projects portfolio it is necessary to answer the question: for which values of possible i-th operation commencement states on j-th resource in k-th project ($SP_{i,j}^{k}, ST_{i,j}^{k}$) and in which(Q) processes execution sequence is projects execution in the set time possible?

The problem of multi-criterion projects portfolio execution efficiency is evaluated. The solution includes a series of project portfolio executions, i.e. solutions of resources conflicts (making decisions which assure correct company work) together with evaluation of alternative project portfolio execution variants in accordance with the company specification and preferences. It is therefore necessary to answer the question: taking into account company evaluation criteria which order portfolio variant is best? The answer to the question requires solving a series of subproblems e.g. can a project portfolio be executed in determined deadlines and at set cost in conditions of constrained availability to shared resources?

Due to the complexity of the issue, the solution searching commencement procedure should take place under the conditions that a solution exists. It is therefore necessary to search for sufficient conditions (features and properties) which guarantee the existence of possible solution(s). In other words values of $SP_{i,j}^{k}, ST_{i,j}^{k}$, Q are sought, which guarantee the existence of a solution.

The first subproblem of the analysed issue includes the answer to the question: is the P project execution in an arbitrarily set time possible for the $SP_{i,j}^{k}, ST_{i,j}^{k}$ and Q determined values? If so, what is the individual stages execution schedule? And are there alternative solutions variants for the execution of a group of projects?

Sufficient conditions in form of order commencement times, initial resources assignment to orders and resources conflicts settlement rules are sought. Sufficient conditions are sought among operation times, initial states (operations assignment to resources), priority choice rules. It means that the proposed approach is used to verify knowledge base coherence using the logic-algebraic method applied in CP/CLP techniques.

3. LOGIC-ALGEBRAIC METHOD BASED APPROACH TO KNOWLEDGE BASE SPECIFICATION

3.1 Logic-algebraic method

Elements of the considered systems class may be considered as knowledge representation (*KB*). Knowledge representation is presented as *C*, *W*, *Y*, sets which determine the *c*, *w*, *y* and variables domains describing certain system properties at quantitative level. Variables *c* are input variables, determining system input properties, variables *w* are support variables, variables *y* are output variables determining system output properties. Knowledge determining properties of the system, is represented as a set of facts *F*(*c*,*w*,*y*). Facts *F*(*c*,*w*,*y*) are tasks which characterize (on a logical level) relations between the variables *c*, *w*, *y*.

Information used for the construction of facts may be of various linguistic, algebraic expression form etc.

Triples *c*, *w*, *y*, for which all *F*(*c*,*w*,*y*) facts are true, are presented as *RE* relations. Knowledge representation therefore has a form of:

$$KB=<C,W,Y,RE> \qquad (1)$$

where: $RE = \{(c,w,y)\colon F(c,w,y) = 1\}$ – relation being the set of all triples (c,w,y), for which the facts F describing the system are true,

$F(c,w,y) = (F_1(c,w,y)\ F_2(c,w,y), \ldots , F_K(c,w,y))$ - is the sequence of the logic fact values being the functions of the variables c, w, y; $c = (c_1, c_2, \ldots ,c_m)$ – set of input variables; $y = (y_1, y_2, \ldots ,y_n)$ – set of output variables; $w = (w_1, w_2, \ldots ,w_o)$ – set of support variables; c C, y Y, w W, C, Y, W – sets determining c, y, w variables domains.

The project portfolio knowledge representation will therefore be as follows:

$$KB=\left\langle SP_{i,j}^{k}, ST_{i,j}^{k}, Q, X, RE\right\rangle \qquad (2)$$

where:

$RE=\left\{\left(SP_{i,j}^{k}, ST_{i,j}^{k}, Q, x, RE\right)\colon F\left(SP_{i,j}^{k}, ST_{i,j}^{k}, Q, x, RE\right)=1\right\}$ - relation between individual variables:

$F(SP_{i,j}^{k}, ST_{i,j}^{k}, Q, x) = 1$ – facts determining relations between variables.

An input relation RE_x, is sought for project portfolio described by *KB* knowledge representation, which would facilitate meeting a known output relation RE_y. Relations RE_x and RE_y are defined as follows:

$RE_x = \{(SP^k_{i,j}, ST^k_{i,j}, Q)\text{: } F_x(SP^k_{i,j}, ST^k_{i,j}, Q) = 1\}$ – set of $SP^k_{i,j}, ST^k_{i,j}, Q$ values, which meet system input property $F_x(SP^k_{i,j}, ST^k_{i,j}, Q)$.

where:

$F_x(SP^k_{i,j}, ST^k_{i,j}, Q)$ – is a set of logical sentences describing system input properties depending on the initial state of operation execution and priority rules, determining project execution sequence.

$RE_y = \{x\text{: } F_y(x) = 1\}$ – set of x, values which meet system output property $F_y(x)$.

$F_y(x)$ – is a set of logical statements describing system output properties depending on x sequence value.

Determining RE_x relations (and at the same time $F_x(SP^k_{i,j}, ST^k_{i,j}, Q)$) takes place on the basis of logical-algebraic method (Bubnicki Z., 1998). Relations is sought on the basis of previously determined sets S_{x1} and S_{x2}:

$$\boldsymbol{S_{x1}}\mathbf{=\{}(SP^k_{i,j}, ST^k_{i,j}, Q)\mathbf{:}\ \boldsymbol{F}(SP^k_{i,j}, ST^k_{i,j}, x, Q)\mathbf{=1,}\ \boldsymbol{F_y(x)}\mathbf{=1\}}; \qquad (3)$$

$$\boldsymbol{S_{x2}}\mathbf{=\{}(SP^k_{i,j}, ST^k_{i,j}, Q)\mathbf{:}\ \boldsymbol{F}(SP^k_{i,j}, ST^k_{i,j}, x, Q)\mathbf{=1,}\ \boldsymbol{F_y(x)}\mathbf{=0\}}; \qquad (4)$$

$\mathrm{RE_x = S_{x1} \backslash S_{x2}}$

The RE_x set includes input parameters values which constitute sufficient conditions; if met – they guarantee the existence of a non-empty solutions space for the analysed decision problem.

3.2 Knowledge base specification

Every *KB* knowledge representation of the portfolio project execution could be presented as a *CSP* – constraint satisfaction problem.

The reasons for choosing to represent and solve a decision problem at hand as a *CSP*, is that the representation as a *CSP* is often much closer to the original problem: the variables of the *CSP* directly correspond to problem entities, and the constraints can be expressed without having to be translated into linear inequalities. This makes the formulation simpler, the solution easier to understand, and thereby making the choice of the best solution easier.

CSP problem $=((X, D), C)$ is defined in the following way:

a finite discreet decision variables set $X = \{x_1, x_2,...,x_n\}$, a family of finite variables domains $D = \{D_i \mid D_i = \{d_{i1}, d_{i2}, ..., d_{ij}, ..., d_{im}\}, i = 1..n\}$ and a finite constraints set $C = \{C_i \mid i = 1..L\}$ limiting the values of decision variables.

A solution is such an assignment of the variable values that all constraints are satisfied.

In case of a *CSP* transforming *KB* knowledge representation, facts which are included in $F(SP^k_{i,j}, ST^k_{i,j}, x, Q)$ perform the function of C constraints and variables values perform the function of X variables. Variables domains have a form of sets D. The *CSP* takes the following form:

$$\boldsymbol{CSP}\mathbf{=(((}SP^k_{i,j}, ST^k_{i,j}, x, Q\mathbf{),}\boldsymbol{D}\mathbf{),\ \{}\boldsymbol{F}(SP^k_{i,j}, ST^k_{i,j}, x, Q)\mathbf{=1\})}, \qquad (5)$$

The solution of so understood decision problem with regard to *CSP* is related with solving the following problems:

$$\boldsymbol{CSP_{Sx1}}\mathbf{=((}SP^k_{i,j}, ST^k_{i,j}, x, Q\mathbf{),}\ \ \boldsymbol{D}\mathbf{),}\ \ \mathbf{\{}\boldsymbol{F}(SP^k_{i,j}, ST^k_{i,j}, x, Q)\mathbf{=1,}\ \boldsymbol{F_y(x)}\mathbf{=1\}),} \qquad (6)$$

$$\boldsymbol{CSP_{Sx2}}\mathbf{=((}SP^k_{i,j}, ST^k_{i,j}, x, Q\mathbf{),}\ \ \boldsymbol{D}\mathbf{),}\ \ \mathbf{\{}\boldsymbol{F}(SP^k_{i,j}, ST^k_{i,j}, x, Q)\mathbf{=1,}\ \boldsymbol{F_y(x)}\mathbf{=0\}).} \qquad (7)$$

The solution of problems presented (searching result of all possible solutions) are sets S_{x1} and S_{x2}.

Existence of sufficient conditions substantiates commencement of the solutions searching process, a search for alternative resources allocations which guarantee project execution factors balance.

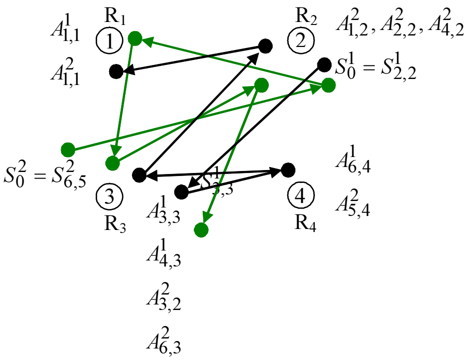
The considered problem can be implemented using the concept of Constraint (Logic) Programming (*C*(*L*)*P*). *C*(*L*)*P* techniques can be applied in DSS's, both in production and in service enterprises, e.g. at the goods transportation planning stage in distribution networks, projects management and production planning (Rossi F., 2000). *C*(*L*)*P* is an emergent software technology for declarative description *CSP* and can be considered as a pertinent framework for the development of *DSS*.

The most important issues that contribute to the efficiency of *CP*/*CLP* techniques are the procedures of a feasible solution selection: constraints propagation and variable distribution. Constraint propagation procedures eliminate decision variables not meeting constraints. This is supplemented with a mechanism (variables distribution), which assigns certain values to the variables. Linking constraint propagation with variables distribution facilitates setting a feasible solution or indicates the lack of a such solution.

CP/*CLP* techniques constitute an alternative (facilitating on-line work) for the currently available systems. It may refer especially to the construction of task oriented interfaces (which facilitate making decisions without necessary operator's interference).

4. ILLUSTRATIVE EXAMPLE

An example of a situation, when two projects with six operations are to be executed have been included in fig. 1. Operations are executed using four resources. Three of them (R_1, R_2 and R_3) are shared by two projects. R_1, R_2 are human resources and R_3 and R_4 are transportation resources. It was assumed that resources R_1 and R_2 are alternative resources and may mutually be used for the execution of project tasks.



$T_{A^k_{i,j}}$ - A_j operation times in k-th project,

$A^k_{i,j}$ - operation assignment to j-th resource in k-th project,

$SP^k_{0,j}$ – the first operation commencement time at j-th resource in k-th project,

$\{R_i \mid i=1,\dots,r\}$ – a set of resources,

R_1, R_2 and R_3 – shared resources.

Fig. 1. Shared resources processes

Operation assignments to a given resource in project planned to be executed are presented as matrixes:

$$A^1_{i,j} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad A^2_{i,j} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Operation execution time on resources and operation assignment to a given resource within a project are presented in the following matrixes:

$$T_{A^1_{i,j}} = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 \\ 0 & 0 & 12 & 0 \\ 0 & 0 & 15 & 0 \\ 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 20 \end{bmatrix} \qquad T_{A^2_{i,j}} = \begin{bmatrix} 20 & 0 & 0 & 0 \\ 0 & 15 & 0 & 0 \\ 0 & 0 & 14 & 0 \\ 0 & 8 & 0 & 0 \\ 0 & 0 & 0 & 10 \\ 0 & 0 & 12 & 0 \end{bmatrix}$$

The R_1 resource use cost is 8 u.j.k. (cost units), R_2 resource - 9 u.j.k, R_3 resource - 11 u.j.k, R_4 resource - 7 u.j.k. Assumed project execution deadline is 35 u.j.c. (time units). Priority rules have been defined and vectors of executed operations sequences for projects has been determined:

$$P_1 = \left(A^1_{2,2}, A^1_{3,3}, A^1_{6,4}, A^1_{4,3}, A^1_{5,1}, A^1_{1,1}\right)$$

$$P_2 = \left(A^2_{6,3}, A^2_{4,2}, A^2_{1,1}, A^2_{3,3}, A^2_{2,2}, A^2_{5,3}\right)$$

Examples of facts defined in the knowledge base:

$$F_1(S_0, Q): A^1_{2,2} \Rightarrow \left(S^1_{2,2} = S_0\right)$$

$$F_2(S_0, Q): A^1_{3,3} \Rightarrow \left(S^1_{3,3} \geq S^1_{2,2} + T_{A^1_{2,2}}\right)$$

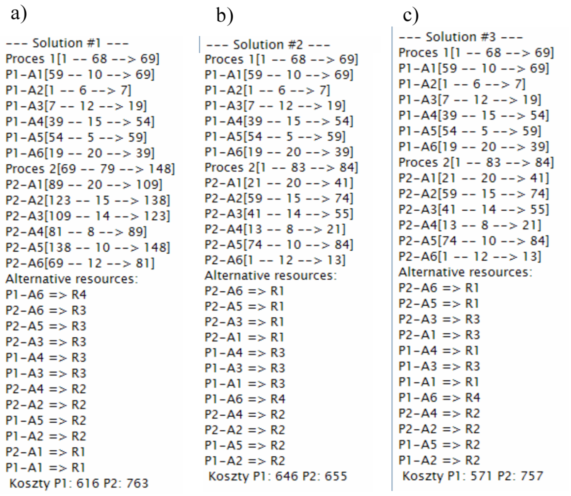
$$F_3(S_0, Q): A^1_{6,4} \Rightarrow \left(S^1_{6,4} \geq S^1_{3,3} + T_{A^1_{3,3}}\right)$$

We need to answer the question: is a project execution for a determined resources number at a scheduled project execution deadline possible? If so, what possible solution variant is to be chosen (taking into account company specific features and demand, especially in chosen company departments e.g. marketing, production and logistics)?

To answer the questions posed by the decision maker it is necessary to develop a computer system, facilitating generation of possible project execution variants and their evaluation directly in the system, without the necessity of user interference.

According to the approach presented in section 3, the CSP has been implemented by means of CP techniques in Ilog.

Examples of schedules generated in the system have been presented in fig. 2. There are three different schedules with different costs, deadlines and using alternative resources.



a)

```
--- Solution #1 ---
Proces 1[1 -- 68 --> 69]
P1-A1[59 -- 10 --> 69]
P1-A2[1 -- 6 --> 7]
P1-A3[7 -- 12 --> 19]
P1-A4[39 -- 15 --> 54]
P1-A5[54 -- 5 --> 59]
P1-A6[19 -- 20 --> 39]
Proces 2[69 -- 79 --> 148]
P2-A1[89 -- 20 --> 109]
P2-A2[123 -- 15 --> 138]
P2-A3[109 -- 14 --> 123]
P2-A4[81 -- 8 --> 89]
P2-A5[138 -- 10 --> 148]
P2-A6[69 -- 12 --> 81]
Alternative resources:
P1-A6 => R4
P2-A6 => R3
P2-A5 => R3
P2-A3 => R3
P1-A4 => R3
P1-A3 => R3
P2-A4 => R2
P2-A2 => R2
P1-A5 => R2
P1-A2 => R2
P2-A1 => R1
P1-A1 => R1
 Koszty P1: 616 P2: 763
```

b)

```
--- Solution #2 ---
Proces 1[1 -- 68 --> 69]
P1-A1[59 -- 10 --> 69]
P1-A2[1 -- 6 --> 7]
P1-A3[7 -- 12 --> 19]
P1-A4[39 -- 15 --> 54]
P1-A5[54 -- 5 --> 59]
P1-A6[19 -- 20 --> 39]
Proces 2[1 -- 83 --> 84]
P2-A1[21 -- 20 --> 41]
P2-A2[59 -- 15 --> 74]
P2-A3[41 -- 14 --> 55]
P2-A4[13 -- 8 --> 21]
P2-A5[74 -- 10 --> 84]
P2-A6[1 -- 12 --> 13]
Alternative resources:
P2-A6 => R1
P2-A5 => R1
P2-A3 => R1
P2-A1 => R1
P1-A4 => R3
P1-A3 => R3
P1-A1 => R3
P1-A6 => R4
P2-A4 => R2
P2-A2 => R2
P1-A5 => R2
P1-A2 => R2
 Koszty P1: 646 P2: 655
```

c)

```
--- Solution #3 ---
Proces 1[1 -- 68 --> 69]
P1-A1[59 -- 10 --> 69]
P1-A2[1 -- 6 --> 7]
P1-A3[7 -- 12 --> 19]
P1-A4[39 -- 15 --> 54]
P1-A5[54 -- 5 --> 59]
P1-A6[19 -- 20 --> 39]
Proces 2[1 -- 83 --> 84]
P2-A1[21 -- 20 --> 41]
P2-A2[59 -- 15 --> 74]
P2-A3[41 -- 14 --> 55]
P2-A4[13 -- 8 --> 21]
P2-A5[74 -- 10 --> 84]
P2-A6[1 -- 12 --> 13]
Alternative resources:
P2-A6 => R1
P2-A5 => R1
P2-A3 => R3
P2-A1 => R3
P1-A4 => R1
P1-A3 => R3
P1-A1 => R1
P1-A6 => R4
P2-A4 => R2
P2-A2 => R2
P1-A5 => R2
P1-A2 => R2
 Koszty P1: 571 P2: 757
```

Fig. 2. Solutions obtained in Ilog system

To choose one of the three generated project execution programmes it is necessary to take into account the opinion and preferences of users e.g. company department managers. A choice of one solution, within the numerous solutions variants, taking into account various evaluation criteria, may be done by means of methods such as: a method of weighted criteria, hierarchic optimization method, limited criteria method or global criterion method. The methods are efficient when the criteria evaluation values are of deterministic character. The assumption does not always allow for reflection of actual conditions in which evaluation information is frequently of an approximated subjective character.

5. CONCLUDING REMARKS

Decision makers face the problem of making optimal decision that meets an organisation's objectives and priorities in different situation under given constraints with various sources of knowledge.

The commercial software packages currently available do not offer a possibility to plan the projects execution in the multi-project environment that is characteristic for small and medium sized enterprises. This gives rise to an increased demand for decision support packages for these companies. Such tools should facilitate answering the question: is a project execution for a determined resources number at a scheduled project execution deadline possible? If so, what possible solution variant is to be chosen (taking into account company specific features and demand)?

The proposed approach to projects portfolio prototyping provides the framework allowing one to take into account both: generating sufficient conditions (which guarantee that a non empty solution set exists) and subsequently choosing the best solution on the basis of chosen evaluation criteria. System properties are presented in formalism of the logic-algebraic method (LAM), which is then easily implemented in a variant of the constraint programming (CP) language. The analysed issue deals with ventures efficiency evaluation in a multi-project environment in constraint conditions e.g. resource, time, sequence and cost constraints.

Further research should be aimed at developing the approach proposed by the possibility of decision support systems design for fuzzy problems. This could be done by linking constraint logic programming including decomposition methods, which are currently used in solving logic algebraic method problems.

REFERENCES

Anavi-Isakow S. and Golany B. (2003). *Managing multi-project environments through constant work-in-process*. International Journal of Project Management, **Vol. 21**, pp. 9-18.

Banaszak Z. and Zaremba M. (2006). *Project-driven planning and scheduling support for virtual manufacturing*, Springer, Journal of Intelligent Manufacturing, **Vol. 17**, No. 6, pp. 641- 651.

Brucker. O., Drexl A., Möhring R., Neumann K. and Pesch E. (1999). *Resource-constrained project scheduling: Notation, classification, models, and methods*. European Journal of Operational Research, **Vol. 112**, pp. 3-41.

Bubnicki Z. (2001). *Uncertain logics, variables and systems*, Proceedings of 13th International Conference on Systems Research, Informatics and Cybernetics, s. 1-5, Baden-Baden.

Bubnicki Z. (1998). *Logic-algebraic method for knowledge-based relation systems*, Systems Analysis Modeling and Simulations, **Vol. 33**.

Dubois D., Fargier H. and Fortemps P. (2003). *Fuzzy scheduling: Modeling flexible constraints vs. coping with incomplete knowledge*, European Journal of Operational Research, **Vol. 147**, pp. 231-252.

Ilog Solver (1995). *Object oriented constraint programming*, Ilog S.A., 12, Av. Raspail, BP 7, 94251 Gentilly cedex, France.

Kerzner H. (1984). *Project Management – A system approach to planning, scheduling and controlling*. Van Nostrand Reinhold Company Inc., New York.

Kolisch R. and Hartmann S. (1998). *Heuristic algorithms for the resource-constrained project scheduling problem: Classification and computational analysis*. (Węglarz J. (Ed)), Project Scheduling: Recent Models, Algorithms and Applications, Kluwer Academic Publishers, Amsterdam, pp. 147-178.

Lova A., Maroto C. and Tormos P. (2000). *A multicriteria heuristic method to improve resource allocation in multiproject scheduling*. European Journal of Operational Research, **Vol. 127**, pp. 408-424.

Rossi F. (2000). *Constraint (Logic) programming: A Survey on Research and Applications*, K.R. Apt et al. (Eds.), New Trends in Constraints, LNAI 1865, Springer-Verlag, Berlin, pp. 40-74.

Sevtsenko E., Rittner R. and Karaulova T. (2006). *Using of Multi-Agents in an Intelligent Decision Support System for Collaborative SME-s*, Nordic Conference on Product Lifecycle Management, pp. 123-134.

Srivastava and Kambhampati S. (2001). *Planning the project management way: Efficient planning by effective integration of causal and resource reasoning in real plan*. Artificial Intelligence, No.131, pp. 73-134.

Tomczuk-Piróg I., Muszyński W. and Bacewicz G. (2006). *CLP – Based Approach to Decision Making Aimed at Production Orders Prototyping*, Proc. of the 12th IEEE Int. Conf. on Methods and Models in Automation and Robotics, pp. 1091-1097.

Tsubakitani S. and Deckro F. R. (1990). *A heuristic for multi project scheduling with limited resources in the housing industry*. European Journal of Operation Research, **Vol. 49**, No. 1, pp. 80-91.

Van Hentenryck P., Perron L. and Puget J. (2000). *Search and Strategies in OPL*, ACM Transactions on Computational Logic, **Vol. 1**, No. 2, pp. 1-36.

Van Roy P. and Haridi S. (2005). *Concepts, Techniques and Models of Computer Programming*, Helion, Gliwice.

Wójcik R., Tomczuk-Piróg I., Banaszak Z. (2007), *Towards Interactive CLP – Based and Project Driven Oriented DSS Design Digital Enterprise Technology, Perspectives and Future Challenges*, Cunha, Pedro Filipe; Maropoulos, Paul G. (Eds.), Springer, XVIII, pp. 351- 359.