A KNOWLEDGE BASED APPROACH TO PRODUCTION PLANNING AND SCHEDULING IN A METALLURGICAL COMPANY

Marcin RELICH [a], Krzysztof WITKOWSKI [a], Sebastian SANIUK [a], Jana ŠUJANOVÁ [b]

[a] *University of Zielona Gora, Zielona Gora, Poland, EU, m.relich@wez.uz.zgora.pl, k.witkowski@wez.uz.zgora.pl, s.saniuk@wez.uz.zgora.pl*

[b] *Slovak University of Technology in Bratislava, Trnava, Slovakia, EU, jana.sujanova@stuba.sk*

Abstract

Decision-making supported by task-oriented software tools plays more and more role in the production companies, including metallurgical plants. Decision-making in production planning and scheduling requires the response in an interactive real-time mode. It is an incentive for developing decision support system (DSS) that enables a fast prototyping of production flows in multi-project environment. The paper aims at providing 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 decision problem formulated in terms of constraint satisfaction problem (CSP). Possible scenarios of the CSP decomposition as well as possibility of different programming languages application lead to a problem of searching for a distribution strategy that enables a real-time mode. A declarative form of the description of a multicriteria decision problem allows its implementation in constraint programming languages and facilitates the development of DSS. Illustrative example concerns optimal steelmaking process scheduling with constraints such as processing time, limited waiting time between adjacent tasks, and amount of resources allocated to tasks. Numerical experiments present the use of constraint programming approach, including various search strategies, to production planning and scheduling in the context of a metallurgical company.

Keywords: production flow, decision support system, constraint programming, search strategies

1. INTRODUCTION

Modern iron and steel companies are moving towards continuous, high-speed and automated production process with large devices. The focus is placed on high quality, low cost, just-in-time delivery and small lot with different varieties. In order to enhance their competitive power, many companies tend to developing computer integrated manufacturing systems which can improve productivity of large devices, shorten waiting-time between operations, reduce material, energy consumption, and finally production costs [1]. Effective production scheduling is a key component of these information systems, especially in today's highly competitive global steel market. The steel-making scheduling problem is to decide unit assignment (the starting and ending times of jobs on the machines) and charge sequence based on established cast plan and steel-making process constraints to achieve some optimization objectives [2].

Scheduling algorithms can be categorized as exact (mathematical programming), heuristic, and search (stochastic) methods. Traditional approach to production scheduling consists mainly of mathematical programming and expert system methods [2]. Because most scheduling problems are NP-hard, it is often difficult to solve them optimally, and combining heuristic methods to get near optimal solutions is often necessary. However, heuristic methods occur the problem with their optimality gap hard to estimate. The expert system can be used to tackle the difficult problems, offering approximate solution to various combinatorial optimization and industrial application problems, being more generic than heuristic methods [2]. The impact of real-life constraints on the decision-making is therefore of great importance, especially for designing interactive and task oriented decision support systems [3-4].

Constraint Programming (CP) environment seems to be especially well suited for modelling real-life and day-to-day decision-making processes in a production company [5-8]. CP is qualitatively different from the other programming paradigms, in terms of declarative, object-oriented and concurrent programming. Compared to these paradigms, constraint programming is much closer to the ideal of declarative programming: to say what we want without saying how to achieve it [9]. CP is an emergent software technology for a declarative Constraints Satisfaction Problem (CSP) description and can be considered as a pertinent framework for the development of decision support system software.

In the field of constraint-based scheduling two strengths emerge: natural and flexible modelling of scheduling problems as CSP and powerful propagation of temporal and resource constraints. Thus, the scheduling problem is modelled as CSP at hand in the required real-life detail and it enables to avoid the classical drawbacks of being forced to discard degrees of freedom and side constraints [9]. The model formulated in terms of CSP determines a single knowledge base and it enables effective implementation in constraint programming languages, as well as the development of a task-oriented decision support system for multi-product scheduling. As a result, the problem specification is closer to the original problem, obtaining solutions that are unavailable with imperative programming. This provides motivation to consider production planning and scheduling in connection with the nature of a company and to develop a reference model that combines both these fields and can be described as a knowledge base. The proposed approach aims at specifying multi-product scheduling in terms of CSP, and using constraint programming to seek a solution to the problem.

The remaining sections of this paper are organized as follows: section 2 presents a problem formulation in terms of CSP for scheduling, section 3 shows an illustrative example of the proposed approach, in turn, some concluding remarks are contained in section 4.

2. CONSTRAINTS SATISFACTION PROBLEM FOR SCHEDULING

The knowledge base approach includes the constraints concerning company and multi-product environment that can be considered in the context of single platform - the reference model. This type of approach seems to be natural in the case of a company that solves standard decision-making problems in the production process. It is assumed that the reference model has the structure of constraints satisfaction problem, and it may be described as follows [10-12]:

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

where: $V = \{v_1, v_2, ..., v_n\}$ - finite set of n variables; $D = \{D_1, D_2, ..., D_n\}$ - finite and discrete domains D of variables; $C = \{c_1, c_2, ..., c_m\}$ - finite set of m constraints limiting and linking decision variables.

The solution of CSP is a vector such that the entry assignments satisfy all the constraints C. Hence, the task is to find the values of variables satisfying all the constraints, i.e. a feasible variant of schedule. Generally, the constraints can be expressed by arbitrary analytical and/or logical formulas as well as bind 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 is employed. Constraint programming is an emergent software technology for declarative description CSP and can be considered as a pertinent framework for development of decision support system software. The main idea behind the CP concept is based on subsequent phases of constraint propagation and variable distribution [9].

In general, the CSP solver performance decreases as the number of variables and constraints increase. However, performance tends to be better when the planning effort to solve the causal structure of the problem is not very high. This situation usually happens when the existence of strict constraints limits the number of valid alternatives for a solution plan and, as a consequence, the propagation mechanism enables solution of the planning component of the problem [13].

In the model, some parameters are determined, among which a set of constraints and decision variables may be distinguished. The constraints combine the variables that describe the capacity of the company (its resources) and the variables concerning the multi-product scheduling. The company and scheduling model containing examples of decision variables and constraints is shown in **Fig. 1**.

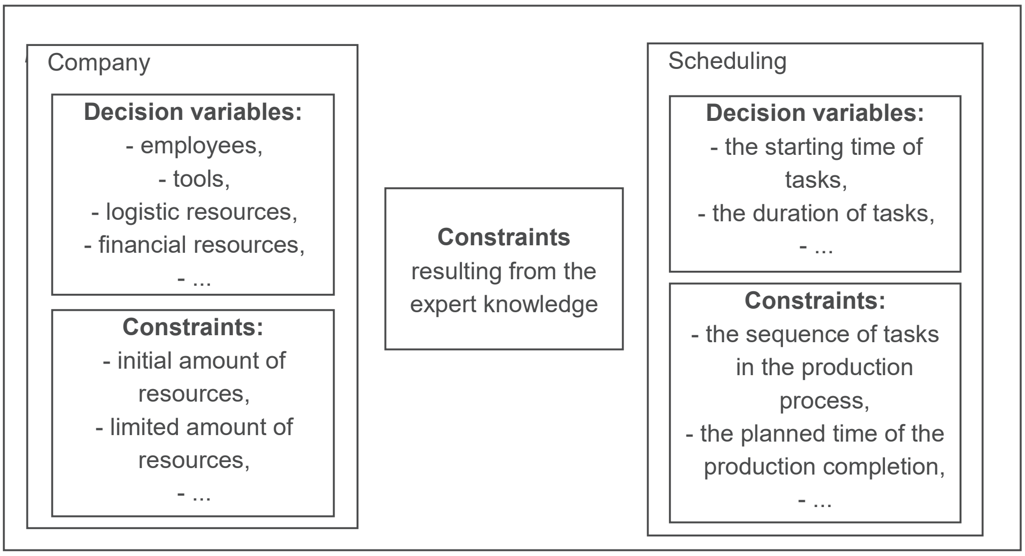

Fig. 1 Company and scheduling model as a common knowledge base

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 KB can be represented in a standard form of CSP [5].

KB can be specified in terms of a system. 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, there are, for example, variables concerning the amount of an enterprise's resources and the multi-product structure. The output of the system is described by the attributes of the object that are unknown or are only partially known. Classification of the decision variables in KB as input-output variables is arbitrarily made and allows the formulation of two classes of standard questions: what results from premises? (e.g. does a given resources allocation ensure the schedule does not exceed the given deadline?) and what implies conclusion? (e.g. what activity duration times and resources amount ensure the given schedule does not exceed the deadline?). 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 [14-16].

In that context, the problem statement of scheduling can be specified in terms of CSP. Given amount z of discrete resources r_k (e.g. workers, robots, money) specified by: $R = (r_1, r_2, ..., r_z)$. Given a product P_i is specified by the set composed of J activities: $P_i = \{O_{i,1}, O_{i,2}, ..., O_{i,J}\}$. The activity $O_{i,j}$ is specified as follows:

$$O_{i,j} = (s_{i,j}, t_{i,j}, Tp_{i,j}, Tz_{i,j}, Dp_{i,j}) \quad (2)$$

where: $s_{i,j}$ and $t_{i,j}$ are the starting time and the duration of the activity $O_{i,j}$, respectively; $Tp_{i,j} = (tp_{i,j,1}, tp_{i,j,2}, ..., tp_{i,j,z})$ - the sequence of moments the activity $O_{i,j}$ requires the *k*-th resource; $Tz_{i,j} = (tz_{i,j,1}, tz_{i,j,2}, ..., tz_{i,j,z})$ - the sequence of moments the activity $O_{i,j}$ releases the *k*-th resource; $Dp_{i,j} = (dp_{i,j,1}, dp_{i,j,2}, ..., dp_{i,j,z})$ - the sequence of the *k*-th resource amount $dp_{i,j,k}$ that is allocated to the activity $O_{i,j}$.

The constraints concerning the company include the initial and available amounts of the resources. Moreover, the product should be completed within the given time horizon $H = \{0, 1, ... , h\}$. It is assumed the activities cannot be suspended during their execution, each activity can request any kind and quantity (not exceeding the resource's limited amount) of any resource, each resource can be uniquely used by an activity, the amount of resource used by an activity cannot be changed or allotted to other activity, and an activity can start its execution only if required amounts of resources are available at the moments given by $Tp_{i,j}$. An example of the above-described problem is presented in next section.

3. ILLUSTRATIVE EXAMPLE

Consider the job shop composed of nine workplaces where with the use of two semi products K_1, K_2, two products P_1 and P_2 are manufactured. The production route with the required resources for executing each task is presented in **Fig. 2**. The problem of production flow prototyping can be seen as an iterative process of adjustment and evaluation of decision variables. At the workplaces three kinds of manufacturing operations are considered: decomposition $\{O_{1,1}, O_{1,2}, O_{1,4}\}$, processing $\{O_{1,3}, O_{1,5}, O_{1,6}, O_{1,7}\}$, and composition $\{O_{1,8}, O_{1,9}\}$,. The workplaces are serviced by three robots (r_1, r_2, r_3) and two workers (r_4, r_5). At least one robot and/or worker is allocated to each $O_{i,j}$. Since production routes specify an order of operation execution, in further considerations, the operations will be treated as activities and the production route as activity network [10].

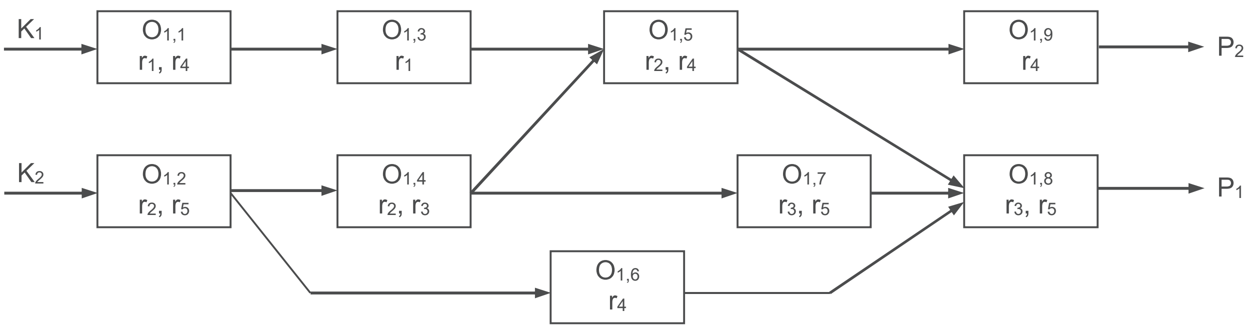

Fig. 2 Production route

The activity times are as follows: T = (2, 1, 1, 4, 2, 2, 5, 4, 1). Type of decision variables, e.g. operation times for robots or workers, is specified in the precise form. Since the amount of common shared resources is limited, their allocation to simultaneously executed activities has to avoid occurrence of closed loop resource requests, i.e. deadlocks. In that context, the problem of multi-robot task allocation in a multi-product job shop reduces to a class of dispatcher's routine questions, such as: does a given resource allocation ensure that production order completion time does not exceed the deadline *h* and the amount of resources is positive at any moment of time horizon $H = \{0, 1, . . . , 15\}$? The answer to above-mentioned question is connected with determination of the starting time of the activity s_j. The constraints concerning the order of production process take the following form: C_1: $s_3 \geq s_1 + t_1$, C_2: $s_4 \geq s_2 + t_2$, C_3: $s_5 \geq s_3 + t_3$, C_4: $s_5 \geq s_4 + t_4$, C_5: $s_6 \geq s_2 + t_2$, C_6: $s_7 \geq s_4 + t_4$, C_7: $s_8 \geq s_5 + t_5$, C_8: $s_8 \geq s_6 + t_6$, C_9: $s_8 \geq s_7 + t_7$, C_{10}: $s_9 \geq s_5 + t_5$. The sought schedule (first admissible solution: S = (0, 0, 2, 1, 5, 1, 5, 10, 7)) is presented in **Fig. 3**.

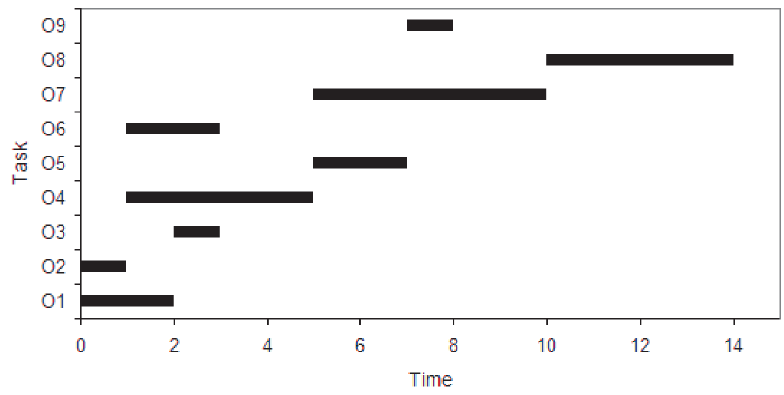

Fig. 3 Schedule for first admissible solution

The reference model encompassing the assumptions of the considered example was implemented in the Oz Mozart programming environment and tested on an AMD Turion(tm) II Ultra Dual-Core M600 2.40GHz, RAM 2 GB platform. **Table 1** presents the results of solution seeking for the different strategies of variable distribution.

Table 1 Comparison of strategies for variable distribution

Distribution strategy	Depth	Time [sec]
Naïve	24	19.62
First-fail	26	19.29
Split	17	18.25

The number of solutions equals 5,252 and they are sought for about 20 seconds. The results show that the Split distribution strategy outperforms the First-fail and Naïve distribution strategy, both in the depth of variable distribution and time.

CONCLUSIONS

The activity of a present enterprise comprises turbulent changes concerning technology, economics, and society. In the present, changeable business environment, quickness of response to customer needs or pressure on innovation and effective cost management determine the success or failure in the struggle for market position [17-18]. This forces more frequent and larger-scale changes in contemporary organizations. The answer to these new challenges can be the application of the decision support systems. Decision-making supported by task-oriented software tools seems to be significant in the modern metallurgical companies, because commercially available ERP systems do not ensure the respond in an interactive online/real-time mode for multi-products planning and scheduling problem [19-20].

The presented knowledge based approach describes a company and manufactured products in terms of the CSP and it enables production planning and scheduling. The proposed approach assumes a kind of reference model encompassing open structure, enabling one to take into account different sorts of variables and constraints as well as to formulate planning and scheduling 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 constraint programming 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

[7]. The advantages of the proposed approach include the possibility of the description of company and manufactured products in terms of a knowledge base.

Further research focuses on the comparison of constraint programming approach with other approaches to planning and scheduling problem, such as exact techniques (mathematical programming, e.g. with the use of mixed integer linear or non-linear programming algorithms), heuristic approaches (e.g. heuristic improvement method), and stochastic approaches (e.g. simulated annealing).

REFERENCES

[1] TANG, L., LIU, J., RONG, A., YANG, Z. A mathematical programming model for scheduling steelmaking-continuous casting production. *European Journal of Operational Research*, 2000, vol. 120, pp. 423-435.

[2] LI, T., LI, Y., ZHANG, Q., GAO, X., DAI, S. Constraint programming approach to steelmaking process scheduling. *Communications of the IIMA*, 2005, vol. 5, no 3, pp. 17-24.

[3] RELICH, M. A decision support system for alternative project choice based on fuzzy neural networks. *Management and Production Engineering Review*, 2010, vol. 1, no 4, pp. 46-54.

[4] RELICH, M., JAKABOVA, M. A decision support tool for project portfolio management with imprecise data. In Proceedings of the *10th International Conference on Strategic Management and its Support by Information Systems*, Valasske Mezirici 2013, Czech Republic, pp. 164-172.

[5] VAN ROY, P., HARIDI, S. *Concepts, techniques and models of computer programming*. Massachusetts Institute of Technology 2004.

[6] RELICH, M. CP-based decision support for scheduling. *Applied Computer Science*, 2011, vol. 7, no 1, pp. 7-17.

[7] RELICH, M. Project prototyping with application of CP-based approach. *Management*, 2011, vol. 15, no 2, pp. 364-377.

[8] RELICH, M. Fuzzy project scheduling using constraint programming. *Applied Computer Science*, 2013, vol. 9, no 1, p. 3-16.

[9] ROSSI, F., VAN BEEK, P., WALSH, T. *Handbook of Constraint Programming*. Elsevier Science 2006.

[10] BANASZAK, Z., BOCEWICZ, G., BACH, I. CP-driven production process planning in multiproject environment, *Decision Making in Manufacturing and Services*, 2008, vol. 2, no 1-2, pp. 5-32.

[11] RELICH, M. A declarative approach to new product development in the automotive industry. In *Environmental Issues in Automotive Industry*, Springer Berlin Heidelberg 2014, pp. 23-45.

[12] SITEK, P., WIKAREK, J. A hybrid approach to supply chain modeling and optimization. In Proceedings of the *Federated Conference on Computer Science and Information Systems,* 2013, Krakow 2013, Poland, pp. 1223-1230.

[13] GARRIDO, A., ARANGU, M., ONAINDIA, E. A constraint programming formulation for planning: from plan scheduling to plan generation. *Journal of Scheduling*, 2009, vol. 12, pp. 227-256.

[14] BANASZAK, Z., ZAREMBA, M. Project-driven planning and scheduling support for virtual manufacturing. Journal of Intelligent Manufacturing, Springer 2006, vol. 17, pp. 641-651.

[15] RELICH, M., BANASZAK, Z. Reference model of project prototyping problem. *Foundations of Management - International Journal*, 2011, vol. 3, no 1, pp. 33-46.

[16] DOSKOCIL, R. Microsoft project as a knowledge base for project management. In *Creating Global Competitive Economies*, Roma 2013, Italy, International Business Information Management Association, pp. 1412-1418.

[17] CAMBAL, M., CAGANOVA, D., SUJANOVA, J. The industrial enterprise performance increase through the competency model application. In Proceedings of the *4th European Conferences on Intellectual Capital*, Helsinki 2012, Finland, pp. 118-126.

[18] SAMAKOVA, J., SUJANOVA, J., KOLTNEROVA, K. Project communication management in industrial enterprises. In Proceedings of the *7th European Conference on Information Systems Management and Evaluation*, Gdansk 2013, Poland, pp. 155-163.

[19] RELICH, M. Knowledge acquisition for new product development with the use of an ERP database. In Proceedings of the *Federated Conference on Computer Science and Information Systems*, Krakow 2013, Poland, pp. 1285-1290.

[20] RELICH, M., WITKOWSKI, K., SANIUK, S., SUJANOVA, J. Material demand forecasting: an ERP system perspective. *Applied Mechanics and Materials*, 2014, vol. 527, pp. 311-314.