*M.Sc. Krzysztof Bzdyra*
*Technical University of Koszalin*
*Ph.D. Paweł Sitek*
*Technical University of Kielce*
*Prof. Zbigniew Banaszak*
*Technical University of Koszalin*

# CLP-BASED APPROACH TO INTEGRATED PLANNING OF PRODUCTION FLOW

*A methodology of propagation of constraints specifying particular production flow planning subproblems is presented. Its implementation to an integration of production planning processes within the CLP framework is considered. Multiple examples illustrate the concept proposed.*

## 1. INTRODUCTION

Production planning integrate many decision making processes which provide solutions to interacting each other problems, i.e. including manufacturing and transportation routing, manufacturing and transportations batch sizing as well as scheduling of manufacturing and transportations tasks [6]. Since production planning has a NP-hard character, hence any admissible solution seems to be satisfactory especially for SMs enterprises. That is because an admissible plan of production flow can be treated as a positive response to the question: Whether in an enterprise a production order might be completed while following a given set of due time and production cost constraints. In this context the CLP languages employing a constraint propagation concept seem to the well suited to solve this problem [2], [3], [4]. It means a methodology of constraints propagation, i.e. constraints specifying particular production planning subproblems is the main contribution of the paper.

In Section 2 the background of CLP is recalled as well as an adapted concept of DSS aimed at production order prototyping is presented. Section 3 introduces into a problem of a manufacturing flow planning a structure of which consists of the qualitative and quantitative flow planning as well as batch sizing problems. In Section 4 an illustrative example followed by a structure of transportation flow planning is provided. Concluding remarks and further work are pointed out in Section 5.

## 2. PROBLEM FORMULATION

### 2.1 Constraint programming

Constraint programming is based on the two basic techniques: the constraints propagation and constraints distribution [1], [4]. Constraint propagation is an efficient inference mechanism obtained with a help of concurrent propagators accumulating information in constraints store. Alternative propagation and distribution may eventually determine a solution to a problem.

A *constraint* is a formula of logic predicate. A *finite domain problem* is a finite set $P$ of constraints, such that constraints, constraint every variable occurring in $P$. A *variable assignment* is a function mapping variables to integers. A solution of a finite domain problem $P$ is a variable assignment that satisfies every constraint in $P$. *Constraint propagation* is an interference rule for finite domain problems that narrows the domains of variables. For instance, for a given inequality $X < Y$ and domain constraints $X = \{23,24,\ldots,100\}$ and $Y = \{1,2,\ldots,33\}$, the result of constraints propagation can narrow the domains of $X$ and $Y$ as follows $X = \{23,24,\ldots,32\}$ and $Y = \{24,\ldots,33\}$.

The *constraint store* stores a conjunction of constraints. A propagator for a constraint $C$ is a concurrent computational agent that tries to narrow the domains of the variables occurring in $C$. Two propagators that share the variable X can communicate with each other through the constraint store.

A variable assignment is called a *solution* of a space if it satisfies the constraints in the constraint store and the all constraints imposed by the propagators.

The combination of constraints propagation and distribution yields a complete solution method for finite domain problems. Distribution process gives a *search tree*. Each node of the tree corresponds to a space. Each leaf of the tree corresponds to a space that is either solved or failed (see Fig.1). The search tree is always finite since there is only finitely many variables all a priori constrained to finite domains.

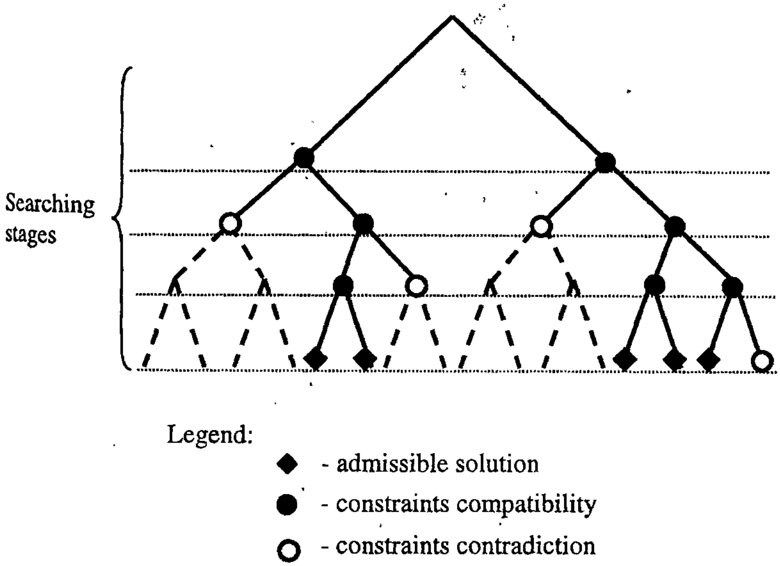


*Fig.1 Search tree*

A distributor is a computational agent implementing a *distribution strategy*. Usually, a distribution strategy is defined on a sequence $x_1,\ldots,x_n$ of variables. When distribution step is necessary, the strategy selects a not yet determined variable in the sequence and distributes on this variable.

A *model of a problem* is a representation of the problem as the finite domain problem. A model specifies the variables and the constraints representing the problem. Nontrivial problems will admit different models and different distribution strategies, coming with different computational properties and search trees of different size. The art of constraint programming consists in finding for a problem a model and a distribution strategy that yield a computationally feasible search tree

## 2.2 Decision support system

The most frequently observed decision problem a producer is faced with concerns a question whether a capability of his manufacturing system can meet the constraints imposed by a given production order. The relevant situation is shown in Fig. 2.

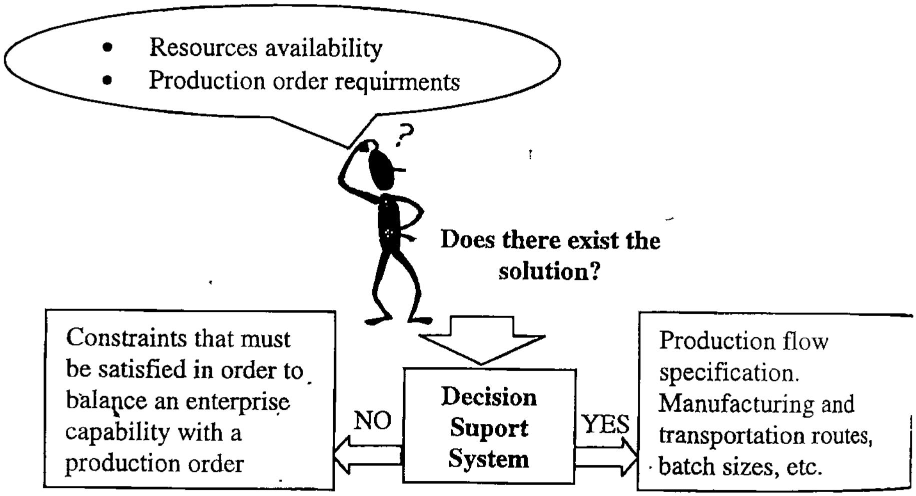


*Fig. 2. Concept of decision making support system*

It means, the question considered is: Whether the constraints imposed by a production order can be satisfied by an enterprise capability, i.e. whether the consumer's requirements can be balanced with producer's resources availability? A positive response to this question means there exists a detailed schedule of production flow containing the manufacturing and transportation routes, production and delivery batch sizes, etc.

The approach proposed can be seen as an alternative to the one based on a computer simulation. From one side it allows to respond to the same question "what if", providing detailed plan of production flow if a balance holds. From another side, however, it provides suggestions (e.g. how to change a consumer requirements and/or a producer capability) supporting negotiation aimed at a production order acceptance.

## 2.3 Production order prototyping

Production flow planning aimed at above mentioned decision problem includes both the manufacturing flow planning and transportation flow planning (see. Fig.2). Assuming capacity of buffers and warehouses as well as productivity of particular workstations the manufacturing flow planning includes selection of a manufacturing route, production batch sizes, and their relevant schedules. In turn the transportation flow planning focuses on the transportation routes, delivery batch sizes, and their schedules – under assumption of a given transportation paths network, a given number and capacity of transportation means.

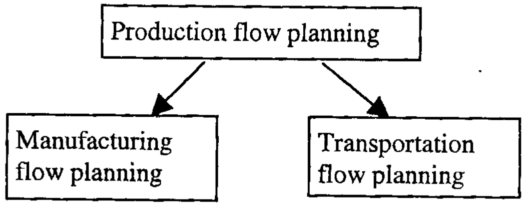


*Fig. 3 Components of the production planning.*

There are two mutually excluding approaches to the production flow-planning problem. Due to the first one a transportation flow planning follows manufacturing flow planning, i.e. manufacturing routes, production batches and their schedules are treated as an input data (constraints) to the transportation problem. The second one, in opposite to the first, approach assumes that manufacturing flows planning problem takes into account the constraints resulting from preceding transportation flows planning one. Depends on domains of considered decision variables one of those two approaches may be observed as more perspective.

In order to illustrate that fact let us assume the domain of the manufacturing flow-planning variable consists of three values while the domain of the transportation flow planning variable consists of seven values. Illustration of the alternative approaches is provided in Fig. 4 and Fig. 5, respectively.

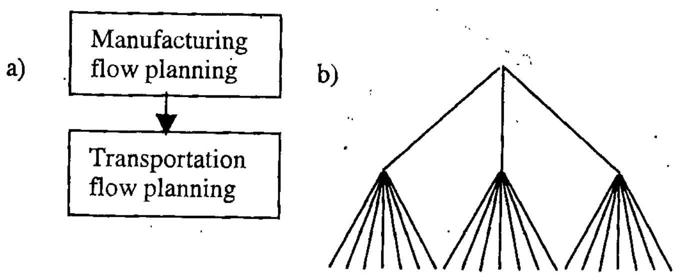


*Fig. 4 Transportation flow planning follows manufacturing flow planning*

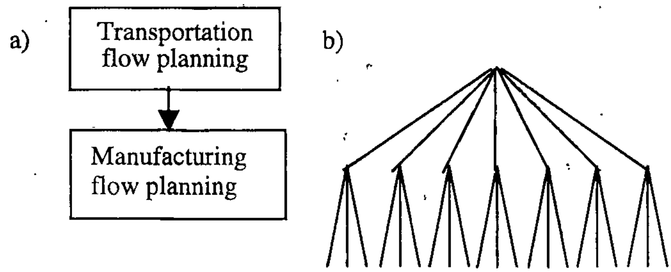


*Fig. 5 Manufacturing flow planning follows transportation flow planning*

The difference in searching capability requirements can be directly seen from the search trees complexity. The search tree from Fig. 4 (b) has 3 + 3 x 7 = 24 branches while the similar tree from Fig. 5 (b) has 7 + 7x3 = 28 branches. It means the first case searching (see Fig. 4) requires less back tracings, than in the second one.

## 3. MANUFACTURING FLOW PLANNING

A manufacturing flow can be seen as a pipeline-like flow of batches following manufacturing route passing through workstations. In case of simultaneously executed concurrently flowing processes deadlocks may occur caused by limited number of resources. So, in order to take into processes competition to the limited shared resource, i.e. in order to take into account the consequences of locally undertaken decisions (dispatching rules) a set of admissible, i.e. deadlock-free flows has to be considered. In general case of multi-flows one my consider subproblems concerning batch sizing and their both qualitative (deadlock-free) and quantitative (deadlock-freeness is not guaranteed) flows (see, Fig.6).

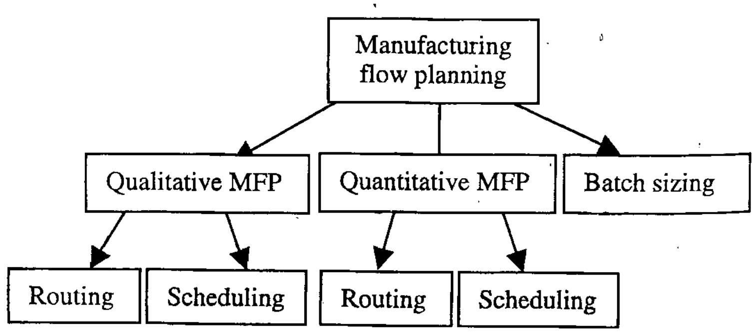


*Fig. 6 Manufacturing flow-planning taking into account concurrent processes execution.*

It should be noted that qualitative as well as quantitative manufacturing flow planning (MFP) concerns both batch routing and batch scheduling. Qualitative MFP, however takes only into account the deadlock-free flows. It means it takes into account a subset of the all-possible quantitative MFP.

# 4. TRANSPORTATION FLOW PLANNING

## 4.1 Structure of the searching problem

A transportation flow can be seen as a pipeline-like flow of delivery batches following transportation route passing transportation paths. In case of simultaneously executed concurrently flowing processes the deadlocks and starvations may occur. The transportation processes may be seen as AGV flows following specified routes connecting workstations waiting for delivery batches. In general case the same manufacturing route can be served by AGVs providing different delivery batches while passing different transportation routs. Therefore, similarly to the MFP the transportation flow-planning (TFP) problem consists of batch sizing and both quantitative and qualitative TFP subproblems (see Fig.7).

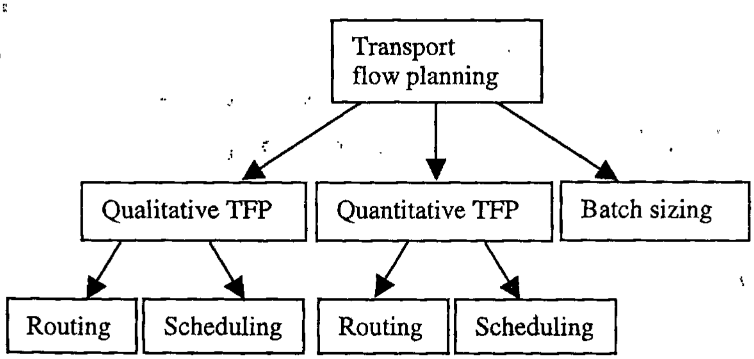


*Fig. 7 Transportation flow-planning taking into account concurrent processes execution.*

## 4.2 An example

Consider Flexible Manufacturing System including two workstations, two AGVs, and a warehouse (see Fig.8). Assume the Workstation_1 has to ship three, and the Workstation_2 two delivery batches to the warehouse. Let the Workstation_1 be served by the AGV_1while the Workstation_2 by the AGV_2. Transportation times assigned to particular path sections are given in Table 1.

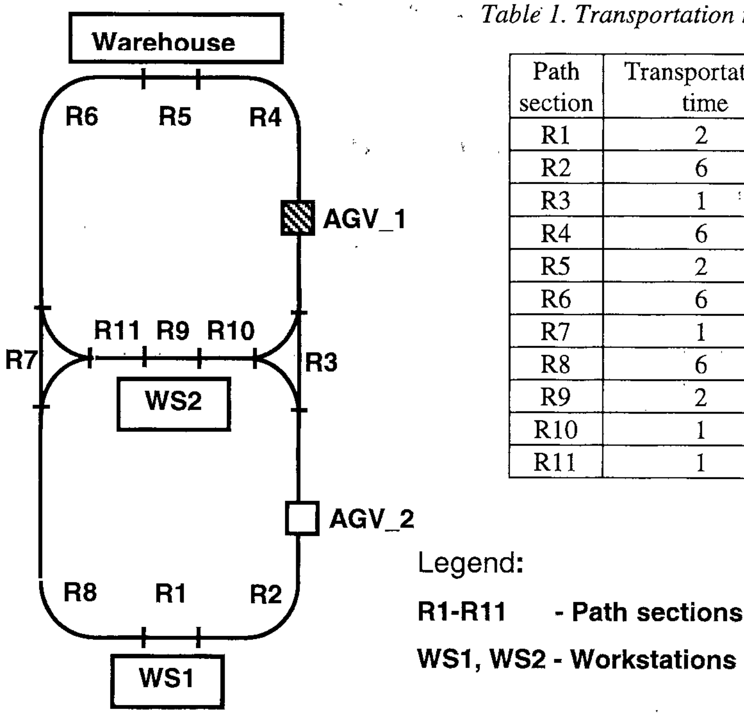


*Table 1. Transportation times*

| Path section | Transportation time |
|---|---|
| R1 | 2 |
| R2 | 6 |
| R3 | 1 |
| R4 | 6 |
| R5 | 2 |
| R6 | 6 |
| R7 | 1 |
| R8 | 6 |
| R9 | 2 |
| R10 | 1 |
| R11 | 1 |

*Fig. 8 Flexible Manufacturing System*

Delivery batches not exceed each AGV's capacity. The question regards of admissible transportation flow, i.e. a flow subject to the following constraints:

- each patch section can be simultaneously occupied by no more than one AGV,
- the all delivers from the Workstation_2 has to reach the warehouse within the period equal to 60 units of time.

The admissible solution assumes the AGV_1 is passing the route R9 – R10 – R3 – R4 – R5 – R6 – R7 – R11 – R9, and the AGV_2 is passing the route R1 – R2 – R3 – R4 – R5 – R6 – R7 – R8 – R1.

The Gantt's chart in Fig. 9 provides the detailed schedule of AGVs determining their access to particular, common shared sectors. It can be easily found that the assumed time limit for workstation_2 deliveries holds.

The solution obtained in CLP language MOZART, guarantees the collision-free, and deadlock-free as well as loop less paths motion of AGVs. The first module works in of-line mode and generates a set of feasible candidate paths. Of course, besides of above kind decision like problems the optimisation ones, e.g. regarding of AGV's minimum time motions can be considered, too.

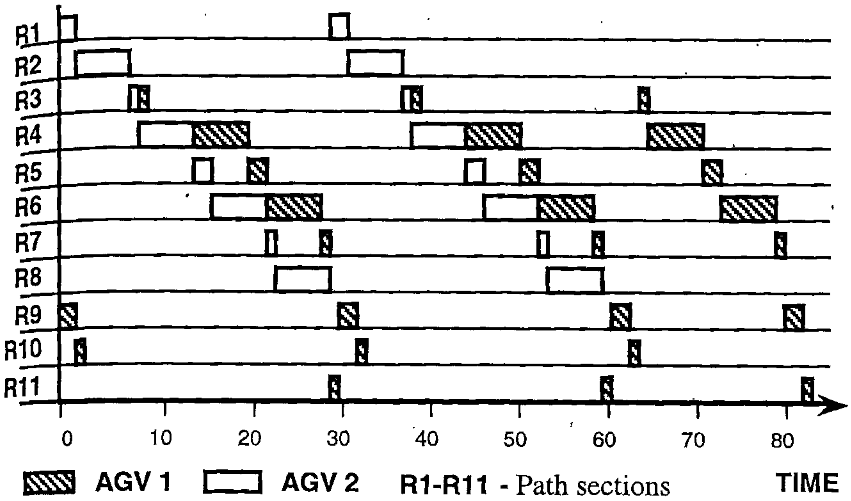


*Fig. 9 Gantt's chart of the feasible transportation flow schedule.*

## 5. CONCLUDING REMARKS

A methodology of propagation of constraints specifying particular production flow planning problems within the CLP framework is considered. Its objective is to provide a computer-implemented procedure allowing coping with the problems of a production order prototyping. The admissible solutions provide workflows scheduling in an enterprise, where resource capacities are limited while taking into account interactions between the work order completion constraints and limited in time system resources availability.

An effective Constraint Logic Programming framework has been implemented in a simple batch delivery problem. Its extension to the whole aggregated production flow planning is a subject of our currently conducted research. Constraint Logic Programming framework has been implemented.

## REFERENCES


1. T. Borowiecki T., D'Souza K., Banaszak Z.A.: *A constraint programming approach for concurrent workflow scheduling*. Proc. of the 6th International Conference on Methods and Models in Automation and Robotics, August 28-31, 2000, Międzyzdroje, Poland, Vol.2, pp.889-894.
2. Pisz I., Tomczuk I., Bzdyra K.: *Towards multi-project decision support tools for small and medium size enterprises*. Methods of Artificial Intelligence. Eds.

T.Burczyński, W. Cholewa, W. Moczulski, Silesian University of Technology, Gliwice, 2003, s.243-248

3. Bzdyra K., Pisz I., Banaszak Z.: *Planowanie przepływu produkcji z wykorzystaniem metod programowania z ograniczeniami.* Prace Naukowe Inst. Technologii Maszyn i Automatyzacji Politechniki Wrocławskiej, Nr 84, seria konferencje nr 41, Oficyna Wyd. Politechniki Wrocławskiej, Wrocław, 2003, s.395-405
4. Schulte CH., Smolka G., Wurtz J.: *Finite Domain Constraint Programming in Oz.* DFKI OZ documentation series, German Research Center for Artificial Intelligence, Stuhlsaltzenhausweg 3, D-66123 Saarbrucken, Germany, 1998.
5. Bzdyra K., Banaszak Z.: *Planowanie przebiegu procesów produkcyjnych z wykorzystanie technik programowania z ograniczeniami.* Komputerowo Zintegrowane zarządzanie – materiały konferencyjne. Wydawnictwa Naukowo-Techniczne 2004.
6. Sobolewski K.: *Podstawy Organizacji Procesu Produkcyjnego,* Wydawnictwo Uczelniane Politechniki Koszalińskiej, Koszalin 1997.