

Application of constraint logic programming to decision support for the supply chain management

Paweł Sitek

Kielce University of Technology, Institute of Management Control Systems

Abstract: Supply Chain Management (SCM) decisions can be considered at different levels of detail. At a strategic level they apply to the architecture in the supply chain, at the tactical level to transport fleet selection, selection of supply sources and distribution, and at the operational level, to the distribution of supplies and route selection. Many models of decision-making SCM have been developed. These are the linear (LP-linear programming) or mixed (MIP/MILP-Mixed Integer/Linear Integer Programming) models. These models are equipped with a smart form. Although they are well known in the OR (Operation Research) environment, they have significant drawbacks. First of all, they must support only linear constraints. For problems of larger dimensions search for solutions is long and inefficient. This paper proposes a CSP-based decision model for SCM and its implementation in the CLP (Constraint Logic Programming). In addition, it presents a novel way of constraints propagation using the structure of the problem.

Keywords: Supply Chain Management (SCM), decision support, Constraint Satisfaction Problem (CSP), Constraint Logic Programming (CLP), hybrid modeling

Integration of SCM is essential. The objective of Supply Chain Integration is to use Information Technology, so that companies may better share information and achieve significant reduction in inventory carrying and others costs. Problems related to the management of the supply chain affect many aspects of production, distribution, warehouse management, supply chain structure etc. The problems are usually closely related with each other, some may influence one another to a greater or lesser extent. Because of the interconnectedness and a very large number of constraints (resource, time, technological, and financial), the environments based on the constraints are suitable for the modeling of those issues in a natural way. We argue that the Constraint Satisfaction Problem (CSP) [1] offers a very good framework for representing the knowledge and information needed for supply chain management. A CSP consists of a set of variables and a set of constraints that must be satisfied. In the supply chain domain, many business rules can be easily represented as constraints. The aim of this paper is to present a hybrid model for decision support in supply chains as a CSP-based model. The paper also presents a novel

method of constraints propagation associated with the new formalization of the problem in Constraint Logic Programming (CLP) environment.

1. Introduction

Huang et al. [3] studied the shared information of supply chain production. They considered and proposed four classification criteria:

- **Supply chain structure:** It defines the way various organizations within the supply chain are arranged and related to each other. The supply chain structure falls into four main types [4]. Convergent: each node in the chain has at least one successor and several predecessors; divergent: each node has at least one predecessor and several successors; conjoined: a combination of each convergent chain and one divergent chain; network: it cannot be classified as convergent, divergent or conjoined, and is more complex than the three previous types.
- **Decision level:** Three decision levels may be distinguished in terms of the decision to be made: strategic, tactical and operational, with their corresponding period, i.e. long-term, mid-term and short-term.
- **Supply chain analytical modeling approach:** This approach consists in the type of representation, in this case, mathematical relationships, and the aspects to be considered in the supply chain. Most literature describes and discusses the linear programming-based modeling approach, mixed integer linear programming models, in particular [5–9].
- **Shared information:** This consists in the information shared between each network node determined by the model, which enables production, distribution and transport planning dependent on the purpose. The shared information process is vital for effective supply chain production, distribution and transport planning. In terms of centralized planning, the information flows from each node of the network where the decisions are made. Shared information includes the following groups of parameters: resources, inventory, production, transport, demand, etc. Minimization of total costs is the main purpose of the models presented in the literature [9–13], while maximization of revenues or sales is considered to a smaller scale [7, 14].

- In our approach [15, 18] we consider the case where:
- the shared information process in the supply chain consists of resources (capacity, versatility, costs), inventory (capacity, versatility, costs, time), production (capacity, versatility, costs), product (volume), transport (cost, mode, time), demand etc.,
 - part of the supply chain has a structure as in fig. 1,
 - the transport is multimodal (several modes of transport, a limited number of means of transport for each mode),
 - the environmental aspects of use of transport modes,
 - different products are combined in one batch of transport,
 - the cost of supplies is presented in the form of a function (in this approach linear function of fixed and variable costs),
 - knowledge related to supply chain management is presented in a linear and logical constraints,
 - a decision model is formulated as a Constraint Satisfaction Problem,
 - a novel method of constraints propagation fundamentally improves the efficiency of finding the solution.

2. Constraint programming

Constraint satisfaction problems (CSPs) are mathematical problems defined as a set of elements whose state must satisfy a number of constraints or limitations. CSPs represent the entities in a problem as a homogeneous collection of finite constraints over variables, which are solved by constraint satisfaction methods. CSPs are the subject of intense research in both Artificial Intelligence (AI) and operations research, since the regularity in their formulation provides a common basis to analyze and solve problems of many unrelated families [2].

Formally, a constraint satisfaction problem is defined as a triple (X, D, C) , where X is a set of variables, D is a domain of values, and C is a set of constraints. Every constraint is in turn a pair (t, R) (usually represented as a matrix), where t is an n -tuple of variables and R is an n -ary relation on D . An evaluation of the variables is a function from the set of variables to the domain of values, $v: X \rightarrow D$. An evaluation v satisfies a constraint $((x_1, \dots, x_n), R)$ if $(v(x_1), \dots, v(x_n)) \in R$. A solution is an evaluation that satisfies all constraints.

Constraint satisfaction problems on finite domains are typically solved using a form of search. The most used techniques are variants of backtracking, constraint propagation, and local search. CSPs are used often in constraint programming. Constraint programming implies the use of constraints as a programming language to encode and solve problems. Constraint logic programming is a form of constraint programming, in which logic programming is extended to include concepts from constraint satisfaction. A constraint logic program is a logic program that contains constraints in the body of clauses. Constraints can

also be present in the goal. These environments are declarative.

3. A CSP-based model for SCM

Previous models [15] of decision support in supply chain management were formulated in the form of mixed integer linear programming (MILP) [16]. Due to the nature of these models and a large number of discrete decision variables, they can only be applied to small problems. Additional disadvantage is the need to use only linear constraints. In practice, the issues related to the production, distribution and supply chain constraints are often logical, nonlinear etc.

For those reasons, the problem was formulated in a new way. The idea was to build a model and to find a framework to be able to:

- implement constraints of previous MILP models,
- introduce new types of constraints (logical, nonlinear etc.),
- increase the efficiency of finding solutions to the problems of larger sizes.

All of the above requirements fulfilled constraint logic programming (CLP). In this framework, a CSP-based decision support model for SCM was developed. It was a hybrid model for combined properties of MILP previous models with a group of logical constraints.

3.1. Model formulation

The model was formulated as a hybrid (mixed linear integer programming [16] and constraint logic [1, 2]) CSP-based under constraints (2) .. (24). Indices, parameters and decision variables in the model together with their descriptions are provided in tab. 1. The simplified structure of the supply chain network for this model is shown in fig. 1. The proposed model is a cost model that takes into account three other types of parameters, i.e. the spatial parameters (area/volume occupied by the product, distributor capacity and capacity of transport unit), time (duration of delivery and service by distributor, etc.) and the transport mode.

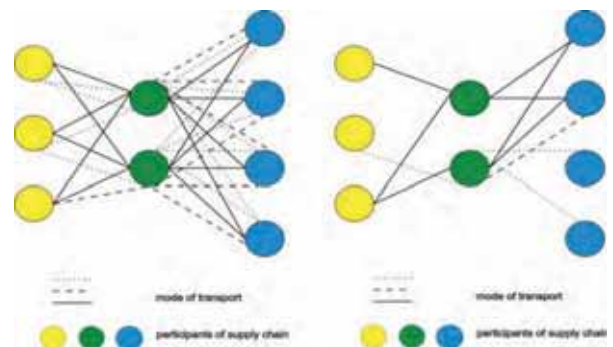


Fig. 1. The simplified structure of the supply chain network (all routes – left, routes for feasible solution – right)

Rys. 1. Uproszczona struktura łańcucha dostaw (wszystkie marszruty – lewy, marszruty dla przykładowego rozwiązania – prawy)

Tab. 1. Summary indices, parameters and decision variables of the model

Tab. 1. Indeksy, parametry i zmienne decyzyjne modelu

Symbol	Description
<i>Indices</i>	
k	product type (k = 1..O)
j	delivery point/customer/city (j = 1..M)
i	manufacturer/factory (i = 1..N)
s	distributor /distribution center (s = 1..E)
d	mode of transport (d = 1..L)
N	number of manufacturers/factories
M	number of delivery points/customers
E	number of distributors
O	number of product types
L	number of mode of transport
<i>Input parameters</i>	
F _s	the fixed cost of distributor/distribution center s (s = 1..E)
P _k	the area/volume occupied by product k (k = 1..O)
V _s	distributor s maximum capacity/volume (s = 1..E)
W _{i,k}	production capacity at factory i for product k (i = 1..N) (k = 1..O)
C _{i,k}	the cost of product k at factory i (i = 1..N) (k = 1..O)
R _{s,k}	if distributor s (s=1..E) can deliver product k (k = 1..O) then R _{sk} = 1, otherwise R _{sk} = 0
TP _{s,k}	the time needed for distributor s (s = 1..E) to prepare the shipment of product k (k = 1..O)
TC _{j,k}	the cut-off time of delivery to the delivery point/customer j (j = 1..M) of product k (k = 1..O)
Z _{j,k}	customer demand/order j (j=1..M) for product k (k = 1..O)
Z _{t,d}	the number of transport units using mode of transport d (d = 1..L)
P _{t,d}	the capacity of transport unit using mode of transport d (d = 1..L)
Tf _{i,s,d}	the time of delivery from manufacturer i to distributor s using mode of transport d (i = 1..N) (s = 1..E) (d = 1..L)
K _{i,s,k,d}	the variable cost of delivery of product k from manufacturer i to distributor s using mode of transport d (d = 1..L) (i = 1..N) (s = 1..E) (k = 1..O)
R _{i,s,d}	if manufacturer i can deliver to distributor s using mode of transport d then R _{isd} = 1, otherwise R _{isd} = 0 (d = 1..L) (s = 1..E) (i = 1..N)
A _{i,s,d}	the fixed cost of delivery from manufacturer i to distributor s using mode of transport d (d = 1..L) (i = 1..N) (s = 1..E)
Koa _{s,j,d}	the total cost of delivery from distributor s to customer j using mode of transport d (d = 1..L) (s = 1..E) (j = 1..M)
Tm _{s,j,d}	the time of delivery from distributor s to customer j using mode of transport d (d = 1..L) (s = 1..E) (j = 1..M)
K2 _{s,j,k,d}	the variable cost of delivery of product k from distributor s to customer j using mode of transport d (d = 1..L) (s = 1..E) (k = 1..O) (j = 1..M)

R2 _{s,j,d}	if distributor s can deliver to customer j using mode of transport d then R2 _{s,j,d} =1, otherwise R2 _{s,j,d} = 0 (d = 1..L) (s = 1..E) (j = 1..M)
G _{s,j,d}	the fixed cost of delivery from distributor s to customer j using mode of transport d (s = 1..E) (j = 1..M) (k = 1..O)
Kog _{s,j,d}	the total cost of delivery from distributor s to customer j using mode of transport d (d = 1..L) (s = 1..E) (j = 1..M) (k = 1..O)
Od _d	the environmental cost of using mode of transport d (d = 1..L)
<i>Decision variables</i>	
X _{i,s,k,d}	delivery quantity of product k from manufacturer i to distributor s using mode of transport d
Xa _{i,s,d}	if delivery is from manufacturer i to distributor s using mode of transport d then Xa _{i,s,d} = 1, otherwise Xa _{i,s,d} = 0
Xb _{i,s,d}	the number of courses from manufacturer i to distributor s using mode of transport d
Y _{s,j,k,d}	delivery quantity of product k from distributor s to customer j using mode of transport d
Ya _{s,j,d}	if delivery is from distributor s to customer j using mode of transport d then Ya _{s,j,d} = 1, otherwise Ya _{s,j,d} = 0
Yb _{s,j,d}	the number of courses from distributor s to customer j using mode of transport d
Tc _s	if distributor s participates in deliveries, then Tc _s = 1, otherwise Tc _s = 0
CW	Arbitrarily large constant

3.2. Objective function

The objective function (1) defines the aggregate costs of the entire chain and consists of five elements. The first is the fixed costs associated with the operation of the distributor involved in the delivery (e.g. distribution center, warehouse, etc.). The second part sets out the environmental costs of using various means of transport. On one hand, those costs are dependent on the one hand on the number of courses of the given means of transport; on the other hand, they depend on the environmental levy, which in turn may depend on the use of fossil fuels and carbon-dioxide emissions. The third component determines the cost of supply from the manufacturer to the distributor. Another component is responsible for the costs of supply from the distributor to the end user (the store, the individual client, etc.). The last component of the objective function determines the cost of manufacturing the product by the given manufacturer.

$$\begin{aligned}
 & \sum_{s=1}^E F_s * Tc_s + \sum_{d=1}^L Od_d (\sum_{i=1}^N \sum_{s=1}^E Xb_{i,s,d} + \sum_{s=1}^E \sum_{j=1}^M Yb_{j,s,d}) + \\
 & + \sum_{i=1}^N \sum_{s=1}^E \sum_{d=1}^L Koa_{i,s,d} + \sum_{s=1}^E \sum_{j=1}^M \sum_{d=1}^L Kog_{s,j,d} + \\
 & + \sum_{i=1}^N \sum_{k=1}^O (C_{ik} * \sum_{s=1}^E \sum_{d=1}^L X_{i,s,k,d})
 \end{aligned} \tag{1}$$

3.3. Constraints

The model was developed subject to constraints (2) .. (24) Constraint (2) specifies that all deliveries of product k produced by the manufacturer i and delivered to all distributors s using mode of transport d do not exceed the manufacturer's production capacity.

Constraint (3) covers all customer j demands for product k ($Z_{j,k}$) through the implementation of supply by distributors s (the values of decision variables $Y_{i,s,k,d}$). The constraint was designed to take into account the specificities of the distributors resulting from environmental or technological constraints (i.e. whether the distributor s can deliver the product k or not). The balance of each distributor s corresponds to constraint (4). The possibility of delivery in due to its technical capabilities – in the model, in terms of the distributor's volume/capacity is defined by constraint (5). Constraint (6) ensures the fulfillment of the terms of delivery time. Constraints (7a), (7b), (8) guarantee deliveries with available transport taken into account. Constraints (9), (10), (11) respectively set values of decision variables based on binary variables Tc_s , $Xa_{i,s,d}$, $Ya_{s,j,d}$. Dependencies (12) and (13) represent the relationship by which total costs are calculated. In general, these may be any linear functions. The remaining constraints (14) .. (23) arise from the nature of the model (MILP).

Constraint (24) allows service modeling for one of the two selected products in the distribution center s . This constraint is the result of technological, marketing, sales or safety reasons. Therefore, some products may not be distributed together. The constraint can be re-used for different pairs of product k and for some or all of the distribution centers s . A logical constraint like this cannot easily be implemented in a linear model. Only declarative application environments based on constraints satisfaction problem (CSP) makes it possible to implement constraints such as (24). Obviously, the addition of this type of constraint changes the model class. It is a CSP-based model.

$$\sum_{s=1}^E \sum_{d=1}^L X_{i,s,k,d} R_{s,k} \leq W_{i,k} \text{ for } i=1..N, k=1..O \quad (2)$$

$$\sum_{s=1}^E \sum_{d=1}^L (Y_{s,j,k,d} * R_{s,k}) \geq Z_{j,k} \text{ for } j=1..M, k=1..O \quad (3)$$

$$\sum_{i=1}^N \sum_{d=1}^L X_{i,s,k,d} = \sum_{j=1}^M \sum_{d=1}^L Y_{s,j,k,d} \text{ for } s=1..E, k=1..O \quad (4)$$

$$\sum_{k=1}^O (P_k * \sum_{i=1}^N \sum_{d=1}^L X_{i,s,k,d}) \leq Tc_s * V_s \text{ for } s=1..E \quad (5)$$

$$Xa_{i,s,d} * Tf_{i,s,a} + Xa_{i,s,d} * Tp_{s,k} + Ya_{s,j,d} * Tm_{s,j,d} \leq Tc_{j,k} \text{ for } i=1..N, s=1..E, j=1..M, k=1..O, d=1..L \quad (6)$$

$$R1_{i,s,d} * Xb_{i,s,d} * Pt_d \geq X_{i,s,k,d} * P_k \text{ for } i=1..N, s=1..E, k=1..O, d=1..L \quad (7a)$$

$$R2_{s,j,d} * Yb_{s,j,d} * Pt_d \geq Y_{s,j,k,d} * P_k \text{ for } s=1..E, j=1..M, k=1..O, d=1..L \quad (7b)$$

$$\sum_{i=1}^N \sum_{s=1}^E Xb_{i,s,d} + \sum_{j=1}^M \sum_{s=1}^E Yb_{j,s,d} \leq Zt_d \text{ for } d=1..L \quad (8)$$

$$\sum_{i=1}^N \sum_{d=1}^L Xb_{i,s,d} \leq CW * Tc_s \text{ for } s=1..E \quad (9)$$

$$Xb_{i,s,d} \leq CW * Xa_{i,s,d} \text{ for } i=1..N, s=1..E, d=1..L \quad (10)$$

$$Yb_{s,j,d} \leq CW * Ya_{s,j,d} \text{ for } s=1..E, j=1..M, d=1..L \quad (11)$$

$$Koa_{i,s,d} = A_{i,s,d} * Xb_{i,s,d} + \sum_{k=1}^O K1_{i,s,k,d} * X_{i,s,k,d} \text{ for } i=1..N, s=1..E, d=1..L \quad (12)$$

$$Kog_{s,j,d} = G_{s,j,d} * Yb_{j,s,d} + \sum_{k=1}^O K2_{s,j,k,d} * Y_{s,j,k,d} \text{ for } s=1..E, j=1..M, d=1..L \quad (13)$$

$$X_{i,s,k,d} \geq 0 \text{ for } i=1..N, s=1..E, k=1..O, d=1..L \quad (14)$$

$$Xb_{i,s,d} \geq 0 \text{ for } i=1..N, s=1..E, d=1..L, \quad (15)$$

$$Yb_{s,j,d} \geq 0 \text{ for } s=1..E, j=2..M, d=1..L, \quad (16)$$

$$X_{i,s,k,d} \in C \text{ for } i=1..N, s=1..E, k=1..O, d=1..L, \quad (17)$$

$$Xb_{i,s,d} \in C \text{ for } i=1..N, s=1..E, d=1..L \quad (18)$$

$$Y_{s,j,k,d} \in C \text{ for } s=1..E, j=1..M, k=1..O, d=1..L \quad (19)$$

$$Yb_{s,j,d} \in C \text{ for } s=1..E, j=1..M, d=1..L, \quad (20)$$

$$Xa_{s,j,d} \in \{0,1\} \text{ for } i=1..N, s=1..E, d=1..L, \quad (21)$$

$$Ya_{s,j,d} \in \{0,1\} \text{ for } s=1..E, j=2..M, d=1..L, \quad (22)$$

$$Tc_s \in \{0,1\} \text{ for } s=1..E \quad (23)$$

$$\text{Exclusion}(X_{i,s,k,d}, X_{i,s,l,d}, s) \text{ for } k \neq l, s=1..S \quad (24)$$

3.4. The concept of model implementation

To implement the proposed model (1) .. (24) a CLP framework was used. The motivation was to offer a declarative way of modeling constraint satisfaction problems (CSP). A constraint logic program is a logic program that contains constraints in the body of clauses. Similarly, as in regular logic programming, programs are queried about the probability of a goal, which may contain constraints in addition to literals. A proof for a goal is composed of clauses whose bodies are satisfiable constraints and literals that can in turn be proved using other clauses. CLP can use Artificial Intelligence (AI) techniques to improve the search: propagation, data-driven computation, "forward checking" and "lookahead" [1, 2]. From a variety of frameworks for the implementation of the CSP model Eclipse software [17] was selected. Eclipse is an open-source software system for the cost-effective development and deployment of constraint programming applications [17].

Due to the nature of decision problems in SCM, in particular, summing up decision variables and constraints involving a lot of variables, the constraints propagation efficiency decreases dramatically. Constraints propagation is one of the most important methods in CLP and therefore its effectiveness affects the effectiveness and scope of the CLP. For that reason, research into more effective methods of constraints propagation for these problems was conducted. A different representation of the problem and manner of implementation was proposed (fig. 3).

In the classical method of implementation (fig. 2) on the basis of the facts contained in the files *orders.ecl* and *configuration.ecl*, adequate representation of the problem is generated and, together with those facts, used in the file *op.ecl*. The file *op.ecl* contains a set of predicates implementing the decision model (1) .. (24).

The proposed novel implementations of the problem introduced additional step generation marked with a dashed line in fig. 3. The generation process is based on

the facts of the files *configuration.ecl* and *orders.ecl* and results in placing all feasible routes as well as other feasible facts in files *routes.ecl* and *others.ecl* in a sequential order.

In this approach, the representation of the problem is also different because it contains only one value that is not set while in the classical approach there are five such values. Details of the problem of representation and the implementation are presented in [18]. Then, all feasible facts and the facts of *orders.ecl* file are transferred to the main file *opn.ecl* (fig. 3). The intermediate step associated with the generation of feasible facts based on the knowledge of the problem structure fundamentally increases the scope of propagation of constraints and narrows the domains of decision variables.

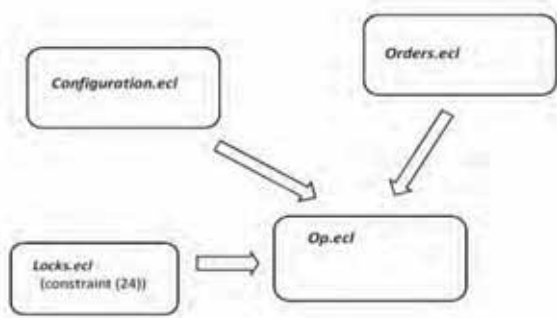


Fig. 2. Block diagram of the implementation of the decision-making model in the CLP framework – classical approach

Rys. 2. Schemat blokowy implementacji modelu decyzyjnego w środowisku CLP – podejście klasyczne

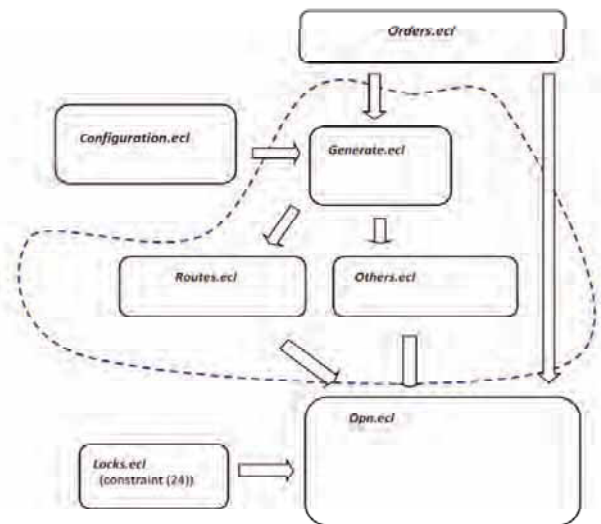


Fig. 3. Block diagram of the implementation of the decision-making model in the framework CLP – a novel approach, extra step marked by the dashed

Rys. 3. Schemat blokowy implementacji modelu decyzyjnego w środowisku CLP – nowe podejście z zaznaczonym dodatkowym krokiem

4. Computational examples

In order to verify and evaluate the proposed approach, many computational experiments were performed. The details of these experiments, the input data sets and the results are presented in [18]. This section presents only a summary of the results (FC-value evaluation function) and the time necessary to find a solution.

All the cases relate to the supply chain with two manufacturers ($i = 1..2$), three distributors ($s = 1..3$), four customers ($j = 1..4$), four mode of transport ($d = 1..4$), and five types of products ($k = 1..5$).

Numerical examples with different input data sets from *orders.ecl* were computed. The number of orders (Orders_N) in specific examples varied from 2 to 12.

The objective function value obtained for the classical approach (FCs), the novel approach (FCn) and computation time (in seconds) is shown in tab. 2.

Tab. 2. The results of numerical examples for both approaches
Tab. 2. Wyniki przykładów liczbowych dla obu podejść

Orders_N	2	4	6	8	10	12
Ts	0.01	0.02	0.06	18.72	6012	----*
Tn	0.02	0.03	0.04	0.05	0.08	0.09
Fcs	3424	8555	14881	36858	51234	----
Fcn	3424	8555	14881	36363	57285	57 285

*The calculation was discontinued after 45 000 seconds

It is clear that the proposed approach with an additional propagation of constraints is much more effective than the classical approach (Ts) because of the computation time (Tn). Moreover, through the use of an innovative approach to the problem of representation and constraints propagation, it is possible to solve problems of sizes larger than the classical approach. Detailed results of experiments and analysis of solutions are presented in [18]. On this basis, you can support the decisions concerning the choice of distributors, routes, modes of transports, factories, etc.

5. Conclusions

The experiments confirmed the correctness of the assumptions. We found that an increase in the propagation of constraints has a critical influence on the process of finding a solution. For larger examples, finding a feasible solution is a long and difficult process if the constraints propagation is insufficient. Therefore, the proposed solution is highly recommended for all types of decision problems in SCM or a similar structure. This structure is characterized by the constraints of many decision variables and their summing. The proposed hybrid modeling method, which combines both MILP and CLP, gives much greater opportunities.

References

1. Apt K., Wallace M., *Constraint Logic Programming using Eclipse*, Cambridge University Press, 2006.
2. Williams, H.P., *Logic and Integer Programming*. Springer, Berlin 2009.
3. Huang, G.Q., Lau, J.S.K., Mak, K.L., *The impacts of sharing production information on supply chain dynamics: a review of the literature*, "International Journal of Production Research" 41, 2003, 1483–1517.
4. Beamon, B.M., Chen, V.C.P., *Performance analysis of conjoined supply chains*, "International Journal of Production Research" 39, 2001, 3195–3218.
5. Kanyalkar, A.P., Adil, G.K., *An integrated aggregate and detailed planning in a multi-site production environment using linear programming*, "International Journal of Production Research" 43, 2005, 4431–4454.
6. Perea-lopez, E., Ydstie, B.E., Grossmann, I.E., *A model predictive control strategy for supply chain optimization*, "Computers and Chemical Engineering" 27, 2003, 1201–1218.
7. Park, Y.B., *An integrated approach for production and distribution planning in supply chain management*, "International Journal of Production Research" 43, 2005, 1205–1224.
8. Jung, H., Jeong, B., Lee, C.G., *An order quantity negotiation model for distributor-driven supply chains*, "International Journal of Production Economics" 111, 2008, 147–158.
9. Rizk, N., Martel, A., D'amours, S., *Multi-item dynamic production–distribution planning in process industries with divergent finishing stages*, "Computers and Operations Research" 33, 2006, 3600–3623.
10. Selim, H., Am, C., Ozkarahan, I., *Collaborative production–distribution planning in supply chain: a fuzzy goal programming approach*, "Transportation Research Part E-Logistics and Transportation Review" 44, 2008, 396–419.
11. Lee, Y.H., Kim, S.H., *Optimal production–distribution planning in supply chain management using a hybrid simulation-analytic approach*, Proceedings of the 2000 Winter Simulation Conference 1 and 2, 2000, 1252–1259.
12. Chern, C.C., Hsieh, J.S., *A heuristic algorithm for master planning that satisfies multiple objectives*, "Computers and Operations Research" 34, 2007, 3491–3513.
13. Jang, Y.J., Jang, S.Y., Chang, B.M., Park, J., *A combined model of network design and production/distribution planning for a supply network*, "Computers and Industrial Engineering" 43, 2002, 263–281.
14. Timpe, C.H., Kallrath, J., *Optimal planning in large multi-site production networks*, "European Journal of Operational Research" 126, 2000, 422–435.
15. Sitek P., Wikarek J., *Cost optimization of supply chain with multimodal transport*, [in:] Proceedings of the Federated Conference on Computer Science and Information Systems, 2012, 1111–1118.
16. Schrijver, A., *Theory of Linear and Integer Programming*, ISBN 0-471-98232-6, John Wiley & Sons. 1998.
17. [www.eclipseclp.org] – EclipseHome (22.11.2012).
18. Wikarek J., *Implementation of decision support model of supply chain management in the environment of constraint logic programming*, "Pomiary Automatyka Robotyka", 2/2013. ■

Zastosowanie programowania w logice z ograniczeniami do wspomagania decyzji zarządzania łańcuchem dostaw

Streszczenie: Decyzje w zarządzaniu łańcuchem dostaw mogą być rozpatrywane na różnych poziomach szczegółowości. Na poziomie strategicznym dotyczą samej struktury i architektury łańcucha, na poziomie taktycznym wyboru floty transportowej, a na poziomie operacyjnym wyboru tras dostaw itd. Opracowano wiele formalnych modeli zarządzania łańcuchem dostaw. Najczęściej były to modele programowania matematycznego liniowego (LP) oraz całkowitoliczbowego (MILP). Chociaż posiadały struktury dobrze rozumiane w środowiskach (OR-Badań Operacyjnych), posiadały istotne wady. Po pierwsze, mogły zawierać jedynie ograniczenia liniowe. Po drugie nie były efektywne przy większych rozmiarach problemów decyzyjnych. W Artykule zaproponowano model decyzyjny dla łańcucha dostaw oparty na problemie spełnienia ograniczeń (CSP-based) oraz jego implementacji w środowisku programowania w logice z ograniczeniami (CLP). Dodatkowo zaprezentowano nowatorski sposób propagacji ograniczeń wykorzystujący strukturę problemu.

Słowa kluczowe: zarządzanie łańcuchem dostaw, wspomaganie decyzji, programowanie w logice z ograniczeniami, modelowanie hybrydowe

Paweł Sitek, PhD

He graduated from the Faculty of Electrical Engineering and Automation Kielce University of Technology. He received a PhD at the Department of Automatic Control, Electronics and Computer Science, Silesian University of Technology. He is the author or co-author of over 100 articles. Main research interests include optimization and decision support for the processes of production, logistics and distribution using conventional MIP (Mixed Integer Programming) and declarative CLP (Constraint Logic Programming) programming environments.

e-mail: sitek@tu.kielce.pl

