Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 35 (2014) 1557 – 1566
Procedia Computer Science

18th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems - KES2014

A hybrid multi-agent approach to the solving supply chain problems

Paweł Sitek[a,*], Izabela Ewa Nielsen[b], Jarosław Wikarek[a]

[a]*Kielce University of Technology, Institute of Management and Control Systems, 25-314 Kielce, Poland*
[b]*University, Department of Mechanical and Manufacturing Engineering, Alborg, Denmark*

Abstract

The paper presents a concept and application of a hybrid multi-agent approach to modeling and optimization the supply chain problems. Two environments (mathematical programming (MP) and constraint logic programming (CLP)) and two types of agents were integrated. This integration and hybridization, complemented with an adequate transformation of the problem, facilitates a significant reduction of the combinatorial problem. The whole process takes place at the implementation layer, which makes it possible to use the structure of the problem being solved, implementation environments and the very data. The strengths of MP and CLP, in which constraints are treated in a different way and different methods are implemented, were combined to use the strengths of both. The proposed approach is particularly important for the decision models with an objective function and many discrete decision variables added up in multiple constraints. The presented multi-agent approach will be compared with classical mathematical programming on the same data sets and models. In addition, the proposed approach will be used for modelling and optimization of the hybrid model, which has not only linear constraints but also logical constraints.

Peer-review under responsibility of KES International.

Keywords: multi-agent; constraint logic programming; mathematical programming; optimization; supply chain management; hybridization

1. Introduction

A supply chain (SC) may be considered an integrated process in which a group of several organizations, such as suppliers, producers, distributors and retailers, work together to acquire raw materials with a view to converting them into end products which they distribute to retailers[1]. Simultaneously considering supply chain production, transport and distribution planning problems greatly advances the efficiency of the all processes. Supply chain (SC)

* Corresponding author. Tel.: +48 413424200
E-mail address: sitek@tu.kielce.pl

Peer-review under responsibility of KES International.
doi:10.1016/j.procs.2014.08.239

optimization involves making decisions for proper organization, and all these processes of supply chains which are vital for retaining the competitive edge of companies in a global economy. These problems are often very large and complex due to the large number of facilities of the supply chain such as the number of plants, warehouses and retailers, and due to complex interactions and constraints among these facilities such as the modes of transport, or relocation of warehouses, the different nature of the demand, and resource/capacity constraints[2] studied the shared information of supply chain production. They considered and proposed four classification criteria: supply chain structure, decision level, modelling approach and shared information. The last criteria 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[3], while maximization of revenues or sales is considered to a smaller scale. This paper deals with a problem of supply chain modelling, optimization and analysis. An important contribution of the presented approach is to propose a hybrid multi-agent implementation platform that supports the modelling, optimization and analysis of decision and optimization problems in the supply chain. In this platform two environments, mathematical programming (MP) and constraint logic programming (CLP), in which constraints are treated in different ways and different methods are implemented, were combined to use the strengths of both in the presented approach for solving complex and constrained problems. As the best structure for the implementation of this approach, multi-agent systems were chosen[4].

The rest of the paper is organized as follows: Section 2 describes our motivation and analyses the state of the art in this domain. Section 3 gives the concept of the novel hybrid multi-agent approach and implementation platform. The optimization models as the illustrative examples are described in Section 4. Computational examples and tests of the implemented model are presented in Section 5. The discussion on possible extensions of the proposed approach and conclusions is included in Section 6.

2. Introduction

Constraints are logical relations between variables, each variable taking values from a specific domain. Thus a constraint restricts the possible values that a variable can take, i.e. it represents some partial information about the variables of interest. Constraints are:

- declarative, they specify a relationship between entities (variables) without determining a specific computational procedure;
- additive, we are interested in the conjunction of constraints and not in the order in which they are imposed;
- rarely independent, normally constraints share variables.

Thus constraints are a natural medium and form for all people (researchers, practitioners, professionals and end-users) to express problems in many fields, especially in logistic, manufacturing, SC etc.

We strongly believe that the constraint-based environment[5-8,15-17] offers a very good framework for representing the knowledge and information needed for the decision support in SC. The central issue for a constraint-based environment is a constraint satisfaction problem (CSP)[6]. Constraint satisfaction problem is the mathematical problem defined as a set of elements whose state must satisfy a number of constraints. Constraint satisfaction problems (CSPs) on finite domains are typically solved using a form of search. The most widely used techniques include variants of backtracking, constraint propagation, and local search. Constraint propagation embeds any reasoning that consists in explicitly forbidding values or combinations of values for some variables of a problem because a given subset of its constraints cannot be satisfied otherwise[8]. There are two reasons, which encourage such a constraint representation. The first reason is that it is closer to the natural problem statement, since variables represent entities and the constraints do not have to be translated, just stated over the entities they concern. The second is that CSP methods are in many cases more efficient in solving such problems than other methods.

CSPs are frequently used in constraint programming. Constraint programming is the use of constraints as a programming language to encode and solve problems. Constraint logic programming (CLP) is a form of constraint programming (CP), 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.

The declarative approach and the use of logic programming provide incomparably greater possibilities for decision problems modeling than the pervasive approach based on mathematical programming. Unfortunately, discrete optimization is not a strong suit of these environments.

Based on[6,8,9,15,16], and previous work[7,10], we observed some advantages and disadvantages of these environments. An integrated approach of constraint logic programming (CLP) and mathematical programming (MP) can help to solve optimization problems that are intractable with either of the two methods alone[11-13]. In both MP and CLP, there is a group of constraints that can be solved with ease and a group of constraints that are difficult to solve.

The vast majority of decision-making and optimization models for the problems of production, logistics, supply chain are formulated in the form of mathematical programming (MIP, MILP, IP)[3]. Due to the structure of these models (adding together discrete decision variables in the constraints and the objective function) and a large number of discrete decision variables (integer and binary), they can only be applied to small problems. Another weakness is that only linear constraints can be used. In practice, the issues related to the production, distribution and supply chain constraints are often logical, nonlinear, etc. For these reasons the problem was formulated in a new way.

The motivation and contribution behind this work was to create a hybrid method and implementation platform for supply chain decision problems modeling and optimization instead of using mathematical programming or constraint programming separately. It follows from the above that what is difficult to solve in one environment can be easy to solve in the other.

The best structure for implementation above approach are Multi Agent Systems (MAS). MAS have become the key technology for decomposing complex problems like SC in order to solve them more efficiently[14]. Furthermore, such a hybrid multi-agent approach allows the use of all layers of the problem (data, structure, methods) to solve it. And finally, the transformation of the problem to a form that can fully exploit the strengths of the constraint propagation.

3. The concept and implementation aspects of the hybrid multi-agent approach

In our approach to modelling and optimization SC problems we proposed the environment, where:

- knowledge related to the problem can be expressed as linear, logical and symbolic constraints;
- the decision models solved using the proposed approach can be formulated as a pure model of MIP/MILP or a hybrid model;
- the problem is modelled in the constraint logic programming environment by CLP-based agents, which is far more flexible than the mathematical programming environment;
- transforming the optimization model to explore its structure has been introduced by CLP-agents;
- constrained domains of decision variables, new constraints and values for some variables are transferred from CLP into MILP/MIP/IP environment by CLP-agents;
- merging and final generation of the model is performed by MP-based agents;
- optimization is performed by MP-based agents.

The schema of the Hybrid Multi-Agent Platform (HMAP) and the concept of this framework with its agents (CA1..CA4, MA1, MA2) is presented in Fig. 1. The names and descriptions of the agents are shown in Table 1.

Table 1. Description of agents in HMAP.

Agent	Description
CA1 *CLP environment*	The implementation of the model in CLP, the term representation of the problem in the form of predicates.
CA2 *CLP environment*	The transformation of the original problem aimed at extending the scope of constraint propagation. The transformation uses the structure of the problem. The most common effect is a change in the representation of the problem by reducing the number of decision variables, and the introduction of additional constraints and variables, changing the nature of the variables, etc.
CA3 *CLP environment*	Constraint propagation for the model. Constraint propagation is one of the basic methods of CLP. As a result, the variable domains are narrowed, and in some cases, the values of variables are set, or even the solution can be found.
CA4 *CLP environment*	Generation by the CA4: • the model for mathematical programming; • additional constraints on the basis of the results obtained by agent CA3; • domains for different decision variables and other parameters based on the propagation of constraints. Transmission of this information in the form of fixed value of certain variables and/or additional constraints to the MP.
MA1 *MP environment*	Merging files generated by CA4 into one file. It is a model file format in MP system.
MA2 *MP environment*	The solution of the model from the previous stage by MP solver. Generation of the report with the results and parameters of the solution.

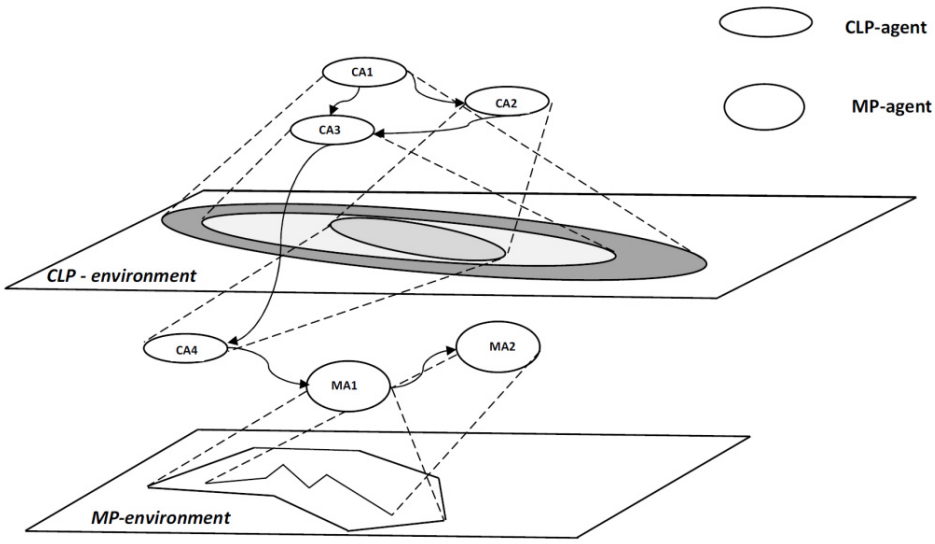

Fig. 1. The scheme of the Hybrid Multi-Agent Platform (HMAP).

4. Examples of supply chain optimization using HMAP

The proposed approach was used and tested on two supply chain optimization models. First model was formulated as a mixed linear integer programming (MILP) problem based on[10,17] under constraints (2) .. (23) in order to test the proposed approach (Fig. 1) against the classical integer programming approach[10]. Then the hybrid model (1) .. (25) was implemented and solved. Indices, parameters and decision variables used in the models together with their descriptions are summarized in Table 2. The simplified structure of the supply chain network for this model, composed of producers, distributors and customers is presented in Fig. 2.

Table 2. Indices, parameters and decision variables.

Symbol	Description
Indices	
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
Input parameters	
F_s	the fixed cost of distributor/distribution center *s*
P_k	the area/volume occupied by product *k*
V_s	distributor *s* maximum capacity/volume
$W_{i,k}$	production capacity at factory *i* for product *k*
$C_{i,k}$	the cost of product *k* at factory *i*
$R_{s,k}$	if distributor *s* can deliver product *k* then $R_{s,k}$=1, otherwise $R_{s,k}$=0
$Tp_{s,k}$	the time needed for distributor *s* to prepare the shipment of product *k*
$Tc_{j,k}$	the cut-off time of delivery to the delivery point/customer *j* of product *k*
$Z_{j,k}$	customer demand/order *j* for product *k*
Zt_d	the number of transport units using mode of transport *d*
Pt_d	the capacity of transport unit using mode of transport *d*
$Tf_{i,s,d}$	the time of delivery from manufacturer *i* to distributor *s* using mode of transport *d*
$K1_{i,s,k,d}$	the variable cost of delivery of product *k* from manufacturer *i* to distributor *s* using mode of transport *d*
$R1_{i,s,d}$	if manufacturer *i* can deliver to distributor *s* using mode of transport *d* then $R1_{i,s,d}$=1, otherwise $R1_{i,s,d}$=0
$A_{i,s,d}$	the fixed cost of delivery from manufacturer *i* to distributor *s* using mode of transport *d*
$Koa_{s,j,d}$	the total cost of delivery from distributor *s* to customer *j* using mode of transport *d*
$Tm_{s,j,d}$	the time of delivery from distributor *s* to customer *j* using mode of transport *d*
$K2_{s,j,k,d}$	the variable cost of delivery of product *k* from distributor *s* to customer *j* using mode of transport *d*
$R2_{s,j,d}$	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
$G_{s,j,d}$	the fixed cost of delivery from distributor *s* to customer *j* using mode of transport *d*
$Kog_{s,j,d}$	the total cost of delivery from distributor *s* to customer *j* using mode of transport *d*
Od_d	the environmental cost of using mode of transport *d*
Decision variables	
$X_{i,s,k,d}$	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

Both models are the cost minimization models that take 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.

The main assumptions made in the construction of these models were as follows:

- 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.2.;
- 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 are taken into account;
- 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);
- the models have linear or linear and logical constraints (hybrid model).

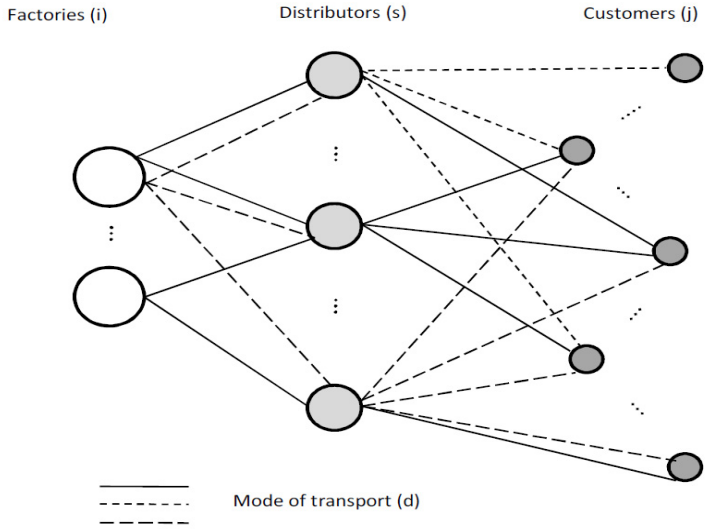

Fig. 2. The simplified structure of the supply chain network.

4.1. Objective function

The objective function (1) is minimized during the optimization corresponds to minimize the costs. It defines the aggregate costs of the entire chain and consists of five elements. The first element comprises the fixed costs associated with the operation of the distributor involved in the delivery (e.g. distribution centre, warehouse, etc.). The second element corresponds to environmental costs of using various means of transport. Those costs are dependent on the number of courses of the given means of transport, and on the other hand, 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 the delivery from the manufacturer to the distributor. Another component is responsible for the costs of the delivery 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. Formulating the objective function in this manner allows comprehensive cost optimization of various aspects of supply chain management. Each subset of the objective function with the same constrains provides a subset of the optimization area and makes it much easier to search for a solution.

$$\sum_{s=1}^{E} F_s \cdot Tc_s + \sum_{d=1}^{L} Od_d \left(\sum_{i=1}^{N}\sum_{s=1}^{E} Xb_{i,s,d} + \sum_{s=1}^{E}\sum_{j=1}^{M} Yb_{j,s,d}\right) + \sum_{i=1}^{N}\sum_{s=1}^{E}\sum_{d=1}^{L} Koa_{i,s,d} + \sum_{s=1}^{E}\sum_{j=1}^{M}\sum_{d=1}^{L} Kog_{s,j,d} + \sum_{i=1}^{N}\sum_{k=1}^{O}\left(C_{ik} \cdot \sum_{s=1}^{E}\sum_{d=1}^{L} X_{i,s,k,d}\right) \quad (1)$$

4.2. Constraints

The model was based on constraints (2)..(25) 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 delivery by distributors *s* (the values of decision variables $Y_{i,s,k,d}$). The flow balance of each distributor *s* corresponds to constraint (4). The possibility of delivery is dependent on the distributor's technical capabilities - constraint (5). Time constraint (6) ensures the terms of delivery are met. Constraints (7a), (7b), (8) guarantee deliveries with available transport taken into account. Constraints (9), (10), (11) 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 based on 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 the production of exclusively one of the two selected products in the factory *i*.

Constraint (25) allows the transport of exclusively one of the two selected products in the same route and transport unit.

Those constraints result from technological, marketing, sales or safety reasons. Therefore, some products cannot be distributed and/or produced and/or transported together. The constraint can be re-used for different pairs of product *k* and for some of or all distribution centres *s* and factories *i*. A logical constraint like this cannot be easily implemented in a linear model. Only declarative application environments based on constraint satisfaction problem (CSP) make it possible to easy implement constraints such as (24), (25). The addition of constraints of that type changes the model class. It is a hybrid model (1)..(25).

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

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

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

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

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

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

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

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

$$\sum_{i=1}^{N}\sum_{d=1}^{L} Xb_{i,s,d} \le CW \cdot Tc_s \quad \text{for } s = 1..E \tag{9}$$

$$Xb_{i,s,d} \le CW \cdot Xa_{i,s,d} \quad \text{for } i = 1..N, s = 1..E, d = 1..L \tag{10}$$

$$Yb_{s,j,d} \le CW \cdot Ya_{s,j,d} \quad \text{for } s = 1..E, j = 1..M, d = 1..L \tag{11}$$

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

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

$$X_{i,s,k,d} \ge 0 \text{ for } i = 1..N, s = 1..E, k = 1..0, d = 1..L \tag{14}$$

$$Xb_{i,s,d} \ge 0 \text{ for } i = 1..N, s = 1..E, d = 1..L, \tag{15}$$

$$Yb_{s,j,d} \ge 0 \text{ for } s = 1..E, j = 1..M, d = 1..L, \tag{16}$$

$$X_{i,s,k,d} \in C \quad \text{for } i = 1..N, s = 1..E, k = 1..0, d = 1..L, \tag{17}$$

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

$$Y_{s,j,k,d} \in C \quad \text{for } s = 1..E, j = 1..M, k = 1..0, d = 1..L \tag{19}$$

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

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

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

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

$$\text{ExclusionP}(k_1,k_2,i) \text{ for } k_1,k_2 \in 1..O,\ i \in 1..N,\ k_1 \neq k_2 \tag{24}$$

$$\text{ExclusionT}(k_1,k_2) \text{ for } k_1 \in 1..O,\ k_2 \in 1..O,\ k_1 \neq k_2 \tag{25}$$

4.3. Model transformation

The ability to transform the problem of using CLP agents is one of the most important features of the proposed approach. Due to the nature of the decision problem (adding up decision variables and constraints involving a lot of variables), the constraint propagation efficiency decreases dramatically. Constraint propagation is one of the most important methods in CLP affecting the efficiency and effectiveness of the CLP and hybrid multi-agent implementation platform (Fig. 1). For that reason, research into more efficient and more effective methods of constraint propagation was conducted. The results included different representation of the problem and the manner of its implementation. The classical problem modeling in the CLP environment consists in building a set of CLP

predicates with parameters. Each CLP predicate has a corresponding multi-dimensional vector representation. While modeling both problems, (1) .. (23) and (1) .. (25), quantities *i, s, k, d* and decision variable $X_{i,s,k,d}$ were vector parameters. The process of finding the solution may consist in using the constraints propagation methods, labeling of variables and the backtracking mechanism[6]. The quality of constraints propagation and the number of backtrackings are affected to a high extent by the number of parameters that must be specified/labeled in the given predicate/vector. In both models presented above, the classical problem representation included five parameters: *i, s, k, d* and $X_{i,s,k,d}$. Considering the domain size of each parameter, the process is complex and time-consuming. Our idea was to transform the problem by changing its representation without changing the very problem. All permissible routes were first generated by CA2 agent based on the fixed data and a set of orders, then the specific values of parameters *i, s, k, d* were assigned to each of the routes. In this way, only one parameter-decision variables $X_{i,s,k,d}$ (deliveries) had to be specified. This transformation fundamentally improved the efficiency of the constraint propagation and reduced the number of backtracks. A route model is a name adopted for the models that underwent the transformation (MILPT).

4.4. Decision support

The proposed models can support decision-making in the following areas:

- the optimization of total cost of the supply chain (objective function, decision variables);
- the selection of the transport fleet number, capacity and modes for specific total costs;
- the sizing of distributor warehouses and the study of their impact on the overall costs;
- the selection of transport routes for optimal total cost.

Detailed studies of these topics are being conducted and will be described in our future articles. We use the hybrid multi-agent approach to both modeling and solving.

5. Numerical experiments

In order to verify and evaluate the proposed approach, many numerical experiments were performed. All the examples relate to the supply chain with seven manufacturers *(i=1..7),* three distributors *(s=1..3)*, ten customers *(j=1..10),* three modes of transport *(d=1..3),* and twenty types of products *(k=1..20).* Experiments began with three examples of P1 .. P9 for the optimization MILP model (1) .. (23). The examples differ in the number of orders (*No*). Distributors capacity is assumed to be $V_1 = V_2 = V_3 = 4000$, while the number of different modes of transport units Ztd_1=25, Ztd_2=40, Ztd_3=80. Other fixed data were taken from the[10]. The first series of experiments was designed to show the benefits and advantages of the hybrid approach. For this purpose the model (1) .. (23) was implemented in both the hybrid (MILPT) and integer programming (MILP) environments.

Table 3. The results of numerical examples for both approaches.

P(No)	**Hybrid Approach**		**Integer Programming**		**Hybrid Approach**	
	MILPT		MILP		HM	
	Fc	*T*	*Fc*	*T*	*Fc*	*T*
P1(100)	10791	416	15459*	900**	13390	578
P2(90)	9263	323	9636*	900**	11164	418
P3(80)	8388	522	8854*	900**	10287	532
P4(60)	6330	345	6330*	900**	8258	385
P5(40)	4473	203	4473	743	5373	234
P6(30)	3488	83	3488	503	3888	97
P7(20)	2877	23	2877	383	3278	32
P8(15)	2266	7	2266	503	2860	15
P9(10)	1756	2	1756	355	2351	9

Symbol	**Description**
Fc	the value of the objective function
T	time of finding solution (in seconds)
*	the feasible value of the objective function after the time T
**	calculation was stopped after 900 s
MILP	MILP model implementation in the MP environment.
MILPT	MILP model after transformation-implementation in the hybrid implementation platform
HM	Hybrid model after transformation-implementation in the hybrid implementation platform.

The experiments that follow were conducted to optimize examples which are implementations of the model (1) .. (25) for the hybrid approach. Examples HM were obtained from MILP by the addition of logical constraints (24), (25).

For each example the solution for the MILPT implementation was found faster from 3 to 150 times than that for the MILP implementation. Moreover, for examples P1 .. P4, the traditional approach based on integer programming gives only feasible solution (calculation was stopped after 900 s) despite using highly efficient MILP solvers. It is obvious that the solution of the hybrid model (HM) was, due to its nature, only possible using the hybrid multi-agent platform. Also, the proposed environment brought the expected results. The results were obtained in only a slightly longer period of time than that necessary for MILPT (examples P1 .. P9).

6. Conclusion

The efficiency of the proposed approach is based on the reduction of the combinatorial problem. This means that using the hybrid approach practically for all models of this class, the same or better solutions are found faster (the optimal instead of the feasible solutions). Another element contributing to the high efficiency of the method is a possibility to determine the values or ranges of values for some of the decision variables (CA3 agent) and transformation of the problem (CA2 agent). Therefore, the proposed solution is highly recommended for all types of decision and optimization problems in supply chain or for other problems with similar structure. This structure is characterized by the constraints of many discrete decision variables and their summation. Furthermore, this method can model and solve problems with more logical constraints. Further work will focus on running the optimization models with non-linear and other logical constraints, multi-objective, uncertainty etc. in the hybrid multi-agent platform.

References

1. Beamon B.M. Supply chain design and analysis: models and methods. *International Journal of Production Economics 55*, 1998, 281–294.
2. 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.
3. Mula J., Peidro D., Diaz-Madronero M. Vicens E. Mathematical programming models for supply chain production and transport planning. *European Journal of Operational Research,* 204*,* 2010, 377–390.
4. Barbati M., Bruno G., Genovese A. Applications of agent-based models for optimization problems: A literature review. *Expert Systems with Applications*, Vol. 39, 2012, 6020-6028.
5. Bocewicz, G., Banaszak, Z.: Declarative approach to cyclic steady states space refinement: periodic processes scheduling. *International Journal of Advanced Manufacturing Technology* 67(1-4), 2013, 137–155.
6. Apt K., Wallace M. Constraint Logic Programming using Eclipse, *Cambridge University Press*, 2006.
7. Sitek P., Wikarek J. A Declarative Framework for Constrained Search Problems. *New Frontiers in Applied Artificial Intelligence*, *Lecture Notes in Artificial Intelligence*, Nguyen, NT., et al. (Eds.), Vol. 5027, Springer-Verlag, Berlin-Heidelberg, 2008, 728-737.
8. Rossi F., Van Beek P., Walsh T**.** Handbook of Constraint Programming (Foundations of Artificial Intelligence)**.** *Elsevier Science Inc*. New York, NY, USA, 2006.
9. Bocewicz G., Wójcik R., Banaszak Z., and Pawlewski P., "Multimodal Processes Rescheduling: Cyclic Steady States Space Approach," *Mathematical Problems in Engineering*, vol. 2013, Article ID 407096, 24 pages, 2013. doi:10.1155/2013/407096.
10. Sitek P., Wikarek J. Cost optimization of supply chain with multimodal transport. *Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2012, 1111-1118.
11. Milano M., Wallace M. Integrating Operations Research in Constraint Programming. *Annals of Operations Research*, vol. 175 issue 1, 2010, 37-76.
12. Achterberg T., Berthold T., Koch T., Wolter K. Constraint Integer Programming. A New Approach to Integrate CP and MIP, *Lecture Notes in Computer Science,* Volume 5015, 2008, 6-20.
13. Bockmayr A., Kasper T. Branch-and-Infer, A Framework for Combining CP and IP. *Constraint and Integer Programming Operations Research/Computer Science Interfaces Series*, Volume 27, 2004, 59-87.
14. Vlahavas I. MACLP: multi agent constraint logic programming, *Information Sciences* 144, 2002, 127-142.
15. 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, 164–172.
16. Relich, M. A declarative approach to new product development in the automotive industry. *In Environmental Issues in Automotive Industry*. Berlin Heidelberg: Springer; 2014, 23–45.
17. Sitek, P., Wikarek, J. A hybrid approach to modeling and optimization for supply chain management with multimodal transport. *IEEE Conference: 18th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2013, 777-782.