

Sitek, Pawel; Wikarek , Jaroslaw

Article

A hybrid method for the modelling and optimisation of constrained search problems

Foundations of Management

Provided in Cooperation with:

Faculty of Management, Warsaw University of Technology

Suggested Citation: Sitek, Pawel; Wikarek , Jaroslaw (2013) : A hybrid method for the modelling and optimisation of constrained search problems, Foundations of Management, ISSN 2300-5661, De Gruyter, Warsaw, Vol. 5, Iss. 3, pp. 7-22,
<https://doi.org/10.2478/fman-2014-0016>

This Version is available at:

<https://hdl.handle.net/10419/184558>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by-nc-nd/3.0>

A HYBRID METHOD FOR THE MODELLING AND OPTIMISATION OF CONSTRAINED SEARCH PROBLEMS

Paweł SITEK*, Jarosław WIKAREK**

Faculty of Electrical Engineering, Automatics and Computer Science
Kielce University of Technology, Kielce, Poland

*e-mail: sitek@tu.kielce.pl

**e-mail: j.wikarek@tu.kielce.pl

Abstract: The paper presents a concept and the outline of the implementation of a hybrid approach to modelling and solving constrained problems. Two environments of mathematical programming (in particular, integer programming) and declarative programming (in particular, constraint logic programming) were integrated. The strengths of integer programming and constraint logic programming, in which constraints are treated in a different way and different methods are implemented, were combined to use the strengths of both. The hybrid method is not worse than either of its components used independently. The proposed approach is particularly important for the decision models with an objective function and many discrete decision variables added up in multiple constraints. To validate the proposed approach, two illustrative examples are presented and solved. The first example is the authors' original model of cost optimisation in the supply chain with multimodal transportation. The second one is the two-echelon variant of the well-known capacitated vehicle routing problem.

Keywords: hybrid methods, constrained search problems, discrete optimisation, decision support, logistics.

1 Introduction

The vast majority of models [1-4] of decision support and/or optimisation in manufacturing, distribution, supply chain management and so on have been formulated as the mixed integer linear programming (MILP) or integer programming (IP) problems, and solved using the operations research (OR) methods. Their structures are similar and proceed from the principles and requirements of mathematical programming. The constraint-based environments have an advantage over traditional methods of mathematical modelling in that they work with a much broader variety of interrelated constraints (resource, time, technological and financial) and allow producing "natural" solutions for highly combinatorial problems.

A Constraint-based environments

We strongly believe that the constraint-based environment [5-7, 27] offers a very good framework for representing the knowledge and information needed for the decision support. The central issue for a constraint-based environment is a constraint satisfaction problem (CSP). CSPs are mathematical problems defined as a set of elements whose state must satisfy a number of constraints. CSPs represent the entities in a problem as a homogeneous collection of finite constraints over variables, which are solved using constraint satisfaction methods. CSPs are the subject of intense study in both

artificial intelligence and OR, since the regularity in their formulation provides a common basis for analysing and solving the problems of many unrelated families [5]. Formally, a CSP 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 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.

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 [26].

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

Constraint logic programming (CLP) is a form of 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. These environments are declarative.

The declarative approach and the use of logic programming provide incomparably greater possibilities for decision problems modelling than the pervasive approach based on mathematical programming.

2 Motivation and state of the art

Based on [1-4], and our previous work [6, 8-12], we observed some advantages and disadvantages of these environments.

An integrated approach of CP and mixed integer programming (MIP) can help to solve optimisation problems that are intractable with either of the two methods alone [13-16]. Although OR and CP have different roots, the links between the two environments have grown stronger in recent years.

Both MIP/MILP/IP and finite domain CP/CLP involve variables and constraints. However, the types of the variables and constraints that are used, and the way the constraints are solved, are different in the two approaches [16].

MILP relies completely on linear equations and inequalities in integer variables, i.e., there are only two types of constraints: linear arithmetic (linear equations or inequalities) and integrity (stating that the variables have to take their values in the integer numbers). In finite domain CP/CLP, the constraint language is richer. In addition to linear equations and inequalities, there are various other constraints: disequalities, non-linear and symbolic (*alldifferent*, *disjunctive*, *cumulative*, etc).

In both MILP/MIP and CP/CLP, there is a group of constraints that can be solved with ease and a group of constraints that are difficult to solve. The easily solved constraints in MILP/MIP are linear equations and inequalities over rational numbers.

Integrity constraints are difficult to solve using mathematical programming methods and often the real problems of MIP/MILP make them NP hard problems.

In CP/CLP, domain constraints with integers and equations between two variables are easy to solve. The system of such constraints can be solved over integer variables in polynomial time. The inequalities between two variables, general linear constraints (more than two variables) and symbolic constraints are difficult

to solve, which makes real problems in CP/CLP NP-hard. This type of constraints reduces the strength of constraint propagation. As a result, CP/CLP is incapable of finding even the first feasible solution.

Both approaches use various layers of the problem (methods, the structure of the problem, data) in different ways. The approach based on OR focuses mainly on the methods of optimisation and, to a lesser degree, on the structure of the problem (Fig. 1). However, the data are completely outside the model. The same model without any changes can be solved for multiple instances of data. In the approach based on CP, due to its declarative nature, the methods are already built-in. The data and structure of the problem are used for its modelling (Fig. 1).

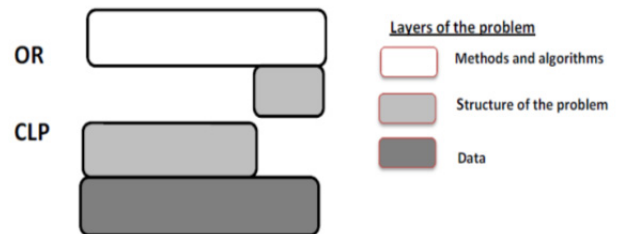


Figure 1. Layers used in the solution of the problem (OR and CP/CLP)

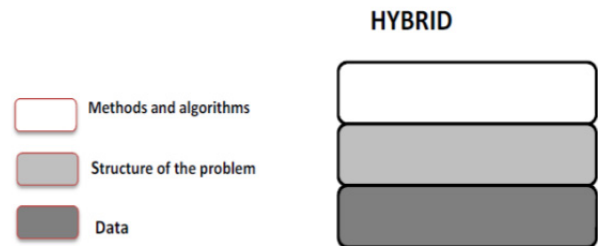


Figure 2. Layers used in the solution of the problem (HYBRID)

The motivation and contribution behind this work was to create a hybrid method for constrained decision problems modelling and optimisation instead of using mathematical programming or CP separately.

It follows from the above that what is difficult to solve in one environment can be easy to solve in the other.

Moreover, such a hybrid approach allows the use of all layers of the problem to solve it (Fig. 2).

The hybrid method is not inferior to its component elements applied separately. This is due to the fact that the number of decision variables and the search area are reduced. The extent of the reduction directly affects the effectiveness of the method.

As mentioned in Section 1, the vast majority of decision-making models for constrained problems in production, logistics and supply chain are formulated in the form of mathematical programming (MIP, MILP, IP).

Due to the structure of these models (summing of 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 disadvantage is that only linear constraints can be used. In practice, the issues related to the production, distribution and supply chain constraints are often logical, non-linear and so on. For these reasons, the problem was formulated in a new way.

In our approach to modelling and optimisation of these problems we proposed the solution environment, where:

- knowledge related to the supply chain can be expressed as linear and logical constraints (implementing all types of constraints of the previous MILP/MIP/IP models [8-11] and introducing new types of constraints (logical, non-linear, symbolic, etc.),
- the decision models solved using the proposed framework can be formulated as a pure model of MILP/MIP/IP or of CP/CLP, or it can also be a hybrid model,
- the problem is modelled in CP/CLP, which is far more flexible than MIP/MILP/IP,
- the novel method of constraint propagation is introduced (obtained by transforming the decision model to explore its structure),
- constrained domains of decision variables, new constraints and values for some variables are transferred from CP/CLP into MILP/MIP/IP,
- the efficiency of finding solutions to larger size problems is increased.

As a result, we obtained a more effective solution environment for a certain class of decision and optimisation problems.

3 Hybrid solution environment

Both environments have advantages and disadvantages. Environments based on the constraints such as CLPs are declarative and ensure a very simple modelling of decision problems, even those with poor structures if any. The problem is described by a set of logical predicates. The constraints can be of different types (linear, non-linear, logical, binary, etc.). The CLP does not require any search algorithms. This feature is characteristic of all declarative backgrounds in which modelling of the problem is also a solution, just as it is in Prolog, SQL and so on. The CLP seems perfect for modelling any decision problem.

Numerous OR models of decision-making have been developed and tested, particularly in the area of decision optimisation. Constantly improved methods and mathematical programming algorithms, such as the simplex algorithm, branch and bound, and branch-and-cost [20] have become classics now.

The proposed method's strength lies in high efficiency of optimisation algorithms and a substantial number of tested models.

Traditional methods when used alone to solve complex problems provide unsatisfactory results. This is related directly to different treatment of variables and constraints in those approaches [2]. The proposed hybrid approach, a composition of methods as described in Section 3, offers the optimal system for specific contexts.

A Architecture and implementation of Hybrid Solution Environment

The hybrid solution environment (HSE) consists of IP/CLP/hybrid models and a hybrid solution framework (HSF) for solving them (Fig. 3). The concept of this framework with its phases (P1 .. P5, G1 .. G3) is presented in Fig. 4.

A detailed description of the phases in the order of execution is shown in Table 1.

From a variety of tools for the implementation of the CP/CLP in HSE, ECLiPSe software [21] was selected. ECLiPSe is an open-source software system for the cost-effective development and deployment of CP applications. Environment for the implementation of MILP/MIP/IP in HSE was LINGO by LINDO Systems [22]. LINGO Optimization Modeling Software is a powerful tool for building and solving mathematical optimisation models [22].

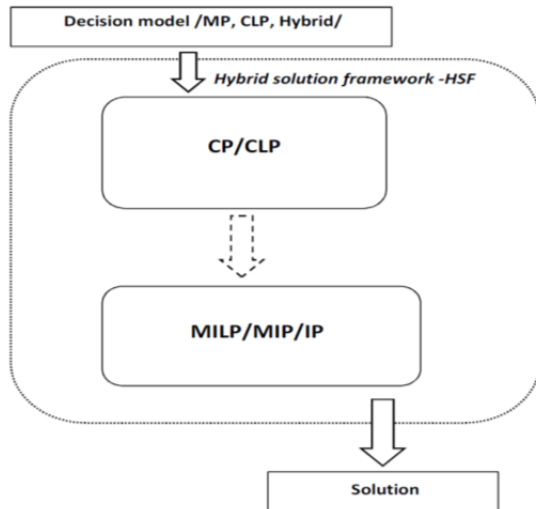


Figure 3. Scheme of the hybrid solution environment (HSE)

(CLP - Constraint Logic Programming, CP - Constraint Programming, IP - Integer Programming, MILP - Mixed Integer Linear Programming, MIP - Mixed Integer Programming)

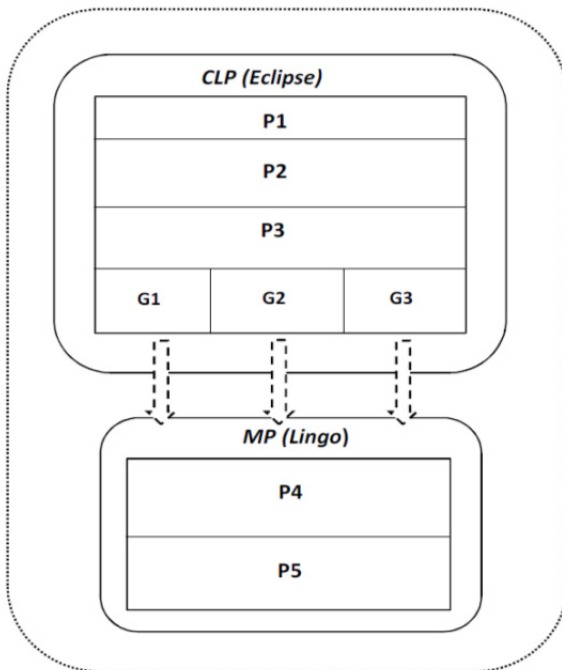


Figure 4. Detailed scheme of the hybrid solution framework (HSF)

(CLP - Constraint Logic Programming)

ECLiPSe software is the environmental leader in HSE. ECLiPSe was used to implement the following phases of the framework: P1, P2, P3, G1, G2 and G3 (Fig. 4, Table 1). The transformed files of the model were transferred from ECLiPSe to LINGO where they were

Table 1. Description of phases

Phase	P1
Name	Implementation of decision model
Description	The implementation of the model in CLP, the term representation of the problem in the form of predicates.
Phase	P2
Name	Transformation of implemented model for better constraint propagation (optional)
Description	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 and so on.
Phase	P3
Name	Constraint propagation
Description	Constraint propagation for the model, which 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.
Phase	G1
Name	Generation of MILP/MIP/IP model
Description	Generation of the model for mathematical programming. Generation performed automatically using CLP predicate. The resulting model is in a format accepted by the system LINGO.
Phase	G2
Name	Generation of additional constraints (optional)
Description	Generation of additional constraints on the basis of the results obtained in step P3.
Phase	G3
Name	Generation domains of decision variables and other values
Description	Generation of 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.
Phase	P4
Name	Merging MILP/MIP/IP model
Description	Merging files generated during the phases G1, G2 and G3 into one file. It is a model file format in LINGO system.
Phase	P5
Name	Solving MILP/MIP/IP model
Description	The solution model from the previous stage by LINGO. Generation of the report with the results and parameters of the solution.

merged (P4). Then the complete model was solved using LINGO efficient solvers (P5). Constraint propagation (phase P3) greatly affected the efficiency of the solution. Therefore, phase P2 was introduced.

During this phase, the transformation was performed using the structure and properties of the model. This is an optional phase that depends on the modelled problem. The details of this phase will be presented in one of the illustrative examples in Section 4 (cost optimisation of supply chain).

CLP, constraint logic programming; IP, integer programming; MILP, mixed integer linear programming; MIP, mixed integer programming.

4 Illustrative examples

The proposed HSE environment was verified and tested for two illustrative examples. The first example is the authors' original model of cost optimisation of supply chain with multimodal transport (section A). The second is a two-echelon capacitated vehicle routing problem (2E-CVRP) model (section B). It is the known benchmark of a very large number of sets/instances of data and their solutions.

A Cost optimisation of supply chain with multimodal transport

During the first stage, the model was formulated as a MILP problem [9, 10, 17] in order to test the proposed environment (Figs 1, 2) against the classical IP environment [22]. The next step involved the implementation and solving of the hybrid model. Indices, parameters and decision variables in the models together with their descriptions are provided in Table 2. The simplified structure of the supply chain network for this model, composed of producers, distributors and customers is presented in Fig. 5.

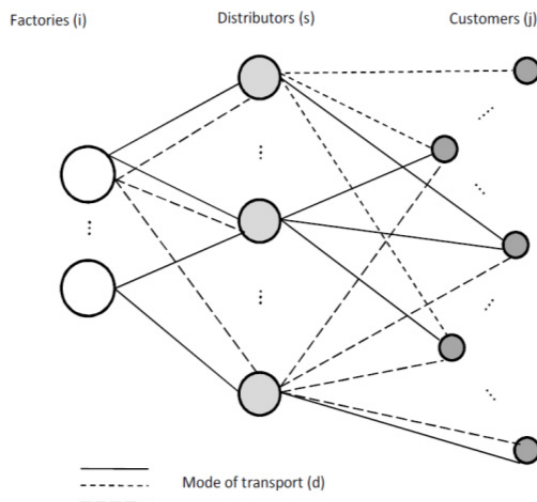


Figure 5. The simplified structure of the supply chain network

Table 2. Summary indices, parameters and decision variables

Symbol	Description
Indices	
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
k	product type ($k = 1, \dots, O$)
j	delivery point/customer/city ($j = 1, \dots, M$)
i	manufacturer/factory ($i = 1, \dots, N$)
s	distributor/distribution centre ($s = 1, \dots, E$)
d	mode of transport ($d = 1, \dots, L$)
Input parameters	
F_s	the fixed cost of distributor/distribution centre 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 $R_{s,k} = 1$ $R_{s,k} = 1$, 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
$Ka_{i,s,k,d}$	the variable cost of delivery of product k from manufacturer i or distributor s using mode of transport d
$Ra_{i,s,d}$	if manufacturer i can deliver to distributor s using mode of transport d then $Ra_{i,s,d} = 1$, otherwise $Ra_{i,s,d} = 0$
$A_{i,s,d}$	the fixed cost of delivery from manufacturer i to distributor s using mode of transport d
$Tm_{s,j,d}$	the time of delivery from distributor s to customer j using mode of transport d
$Kb_{s,j,k,d}$	the variable cost of delivery of product k from distributor s to customer j using mode of transport d
$Rb_{s,j,d}$	if distributor s can deliver to customer j using mode of transport d then $Rb_{s,j,d} = 1$, otherwise $Rb_{s,j,d} = 0$
$G_{s,j,d}$	the fixed 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
$Y_{s,j,k,d}$	delivery quantity of product k from distributor s to customer j using mode of transport d

Table 2 (cont.). Summary indices, parameters and decision variables

Symbol	Description
Decision variables	
$Xa_{i,s,d}$	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
$Ya_{s,j,d}$	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 $Tc_s = 0$
Values calculated	
$Koa_{i,s,d}$	the total 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
Cw	arbitrarily large constant (for example, the sum of all orders)

The proposed models are the cost 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. Multimodality in this example is understood as the possibility of using different modes of transport: railway, commercial vehicles, heavy trucks and so on.

The main assumptions made for 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 and so on,
- a part of the supply chain has the structure as in Fig. 5,
- 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 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),
- models have linear or linear and logical (hybrid model) constraints,
- logical constraints of hybrid model allow the distribution of exclusively one of two selected products in the distribution centre and allow the production of exclusively one of two selected products in the factory.

Objective function

The objective function (1) 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). 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 optimisation of various aspects of supply chain management. Each subset of the objective function with the same constraints provides a subset of the optimisation area and makes it much easier to search for a solution.

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

Constraints

The model was based on 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 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) and (8) guarantee deliveries with available transport taken into account. Constraints (9), (10) and (11) set values of decision variables based on binary variables Tc_s , $Xa_{i,s,d}$ and $Ya_{s,j,d}$. Dependencies (12) and (13) represent the relationship based on which the 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 distribution of exclusively one of the two selected products in the distribution centre s . Similarly, constraint (25) allows the production of exclusively one of the two selected products in the factory i .

Those constraints result from technological, marketing, sales or safety reasons. Therefore, some products cannot be distributed and/or produced 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 CSP make it possible to implement constraints such as (24) and (25).

Adding this type of constraints changes the model class. It is a hybrid model.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$\text{ExclusionP}(X_{i,s,k,d}, X_{i,s,l,d}, i) \text{ for } k \neq l, i=1..N \quad (25)$$

Model transformation

Due to the nature of the decision problem (adding up decision variables and constraints involving a lot of variables), the constraint propagation efficiency de-

creases dramatically. Constraint propagation is one of the most important methods in CLP affecting the efficiency and effectiveness of the CLP and hybrid optimisation environment (Fig. 3, Table 1). For this 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 modelling in the CLP environment consists in building a set of predicates with parameters. Each CLP predicate has a corresponding multi-dimensional vector representation. While modelling both problems, quantities i , s , k , d and decision variable $X_{i,s,k,d}$ were vector parameters (Fig. 6a). As shown in Fig. 6b, for each vector instance, there are five values to be determined. They define the size of the delivery, factories, distributors involved in the delivery and the mode of transport.

$$[Z_n, P, M, D, F, Tu, Tu, Oq, X, T]$$

Fig. 6a Representation of the problem in the classical approach – vector definition

$$\begin{aligned} &[[z_1, p1, m1, \dots, 10, \dots, 8], \\ &[z_2, p1, m2, \dots, 20, \dots, 6], \dots] \end{aligned}$$

Fig. 6b Representation of the problem in the classical approach – a set of vector instances in the process of finding a solution

The process of finding the solution may consist in using the constraint propagation methods, variable labelling and the backtracking mechanism. The number of parameters that must be specified/labelled in the given predicate/vector critically affect the quality of constraint propagation and the number of backtracks. 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 was complex and time-consuming. In addition, the above representation (Fig. 6a, 6b) arising from the structure of the problem is the cause of many backtracks.

Our idea involved the transformation of the problem by changing its representation without changing the very problem. All permissible routes were first generated based on the fixed data and a set of orders, and then the specific values of parameters i , s , k , d were assigned to each of the routes. In this way, only decision variables $X_{i,s,k,d}$ (deliveries) had to be specified (Fig. 7). 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.

$$\begin{aligned} &[[name_1, f1, p1, c1, m1, s1, s1, 5, 12, 100, \dots], \\ &[name_2, f1, p1, c1, m1, s1, s2, 6, 14, 100, \dots], \\ &[name_3, f1, p1, c1, m1, s2, s1, 6, 22, 100, \dots]] \end{aligned}$$

Figure 7. Representation of the problem in the novel approach – a set of feasible routes

Symbols necessary to understand both the representation of the problem and their descriptions are presented in Table 3.

Table 3. Symbols used in the representation of the problem

Symbol	Description
Z_n	order number
P	products, $P\{p_1, p_2, \dots, p_o\}$
M	customers, $M\{m_1, m_2, \dots, m_m\}$
D	distributors, $D\{c_1, c_2, \dots, c_e\}$
F	factories, $F\{f_1, f_2, \dots, f_n\}$
Tu	transport unit, $Tu\{s_1, s_2, \dots, s_l\}$
T	delivery time/period
Oq	order quantity
X	delivery quantity
Name_	routes name-number

Decision-making support

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

- the optimisation of total cost of the supply chain (objective function, decision variables – Appendix A2),
- 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 approach to both modelling and solving.

B Two-Echelon Capacitated Vehicle Routing Problem

The 2E-CVRP is proposed as a benchmark verifying the presented approach. The 2E-CVRP is an extension

of the classical capacitated vehicle routing problem (CVRP) where the delivery depot-customers pass through intermediate depots (called satellites). As in CVRP, the goal is to deliver goods to customers with known demands, minimising the total delivery cost with respect to vehicle capacity constraints. Multi-echelon systems presented in the literature usually explicitly consider the routing problem at the last level of the transportation system, while a simplified routing problem is considered at higher levels [18, 19, 23].

In 2E-CVRP, the freight delivery from the depot to the customers is managed by shipping the freight through intermediate depots. Thus, the transportation network is decomposed into two levels (Fig. 8): the first level connecting the depot (d) to intermediate depots (s) and the second one connecting the intermediate depots (s) to the customers (c). The objective is to minimise the total transportation cost of the vehicles involved in both levels. Constraints on the maximum capacity of the vehicles and the intermediate depots are considered, while the timing of the deliveries is ignored.

From a practical point of view, a 2E-CVRP system operates as follows (Fig. 8):

- freight arrives at an external zone, the depot, where it is consolidated into the first-level vehicles, unless it is already carried into a fully-loaded first-level vehicles,
- each first-level vehicle travels to a subset of satellites that will be determined by the model and then it will return to the depot,
- at a satellite, freight is transferred from first-level vehicles to second-level vehicles.

The mathematical model (MILP) was taken from [18]. Table 4 shows the parameters and decision variables of 2E-CVRP. Figure 8 shows an example of the 2E-CVRP transportation network.

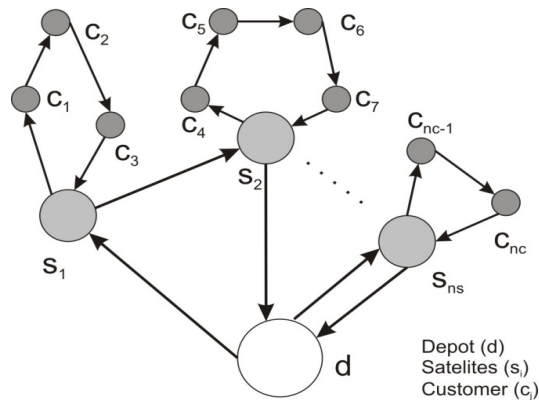


Figure 8. An example of two-echelon capacitated vehicle routing problem transportation network

The transformation of this model in the hybrid approach focused on the resizing of $Y_{k,i,j}$ decision variable by introducing additional imaginary volume of freight shipped from the satellite and re-delivered to it. Such transformation resulted in two facts. First of all, it forced the vehicle to return to the satellite from which it started its trip. Second, it reduced decision variable $Y_{k,i,j}$ to variable $Y_{i,j}$, which decreased the size of the combinatorial problem.

Table 4. Summary indices, parameters and decision variables

Symbol	Description
Indices	
n_s	number of satellites
n_c	number of customers
$V_0 = \{v_o\}$	depot
$V_s = \{v_{s1}, v_{s2}, \dots, v_{sn_s}\}$	set of satellites
$V_c = \{v_{c1}, v_{c2}, \dots, v_{cn_c}\}$	set of customers
Input parameters	
M_1	number of the first-level satellites
M_2	number of the second-level satellites
K_1	capacity of the vehicles for the first level
K_2	capacity of the vehicles for the second level
d_i	demand required by customer i
$c_{i,j}$	cost of the arc (i, j)
s_k	cost of loading/unloading operations of a unit of freight in satellite k
Decision variables	
$X_{i,j}$	an integer variable of the first-level routing is equal to the number of first-level vehicles using arc (i, j)
$Y_{k,i,j}$	a binary variable of the second-level routing is equal to 1 if a second-level vehicle makes a route starting from satellite k and goes from node i to node j and 0 otherwise
$Q_{i,j}^1$	the freight flow arc (i, j) for the first-level
$Q_{k,i,j}^2$	the freight arc (i, j) where k represents the satellite where the freight is passing through
$Z_{k,j}$	a binary variable that is equal to 1 if the freight to be delivered to customer j is consolidated in satellite k and 0 otherwise

$$\min \sum_{i,j \in V_0 \cup V_s} (c_{i,j} \cdot X_{i,j}) + \sum_{k \in V_s} \sum_{i,j \in V_s \cup V_c} (c_{i,j} \cdot Y_{k,i,j}) + \sum_{k \in V_s} (s_k \cdot Ds_k) \quad (1)$$

$$\sum_{i \in V_s} X_{0,i} \leq M_1 \quad (2)$$

$$\sum_{j \in V_s \cup V_0, j \neq k} X_{j,k} = \sum_{i \in V_s \cup V_0, i \neq k} X_{k,i} \text{ for } k \in V_s \cup V_0 \quad (3)$$

$$\sum_{k \in V_s} \sum_{j \in V_c} Y_{k,k,j} \leq M_2 \quad (4)$$

$$\sum_{i \in V_c, j \in V_c} Y_{k,i,j} = \sum_{i \in V_c, j \in V_c} Y_{k,j,i} \text{ for } k \in V_s \quad (5)$$

$$\begin{cases} Ds_j & j \text{ is not the deport} \\ \sum_{i \in V_c} -d_i & \text{otherwise} \end{cases} \text{ for } j \in V_s \cup V_0 \quad (6)$$

$$Q_{i,j}^1 \leq k_1 \cdot X_{i,j} \text{ for } i, j \in V_s \cup V_0, i \neq j \quad (7)$$

$$\begin{cases} \sum_{i \in V_0 \cup V_s, i \neq j} Q_{k,i,j}^2 - \sum_{i \in V_c, i \neq j} Q_{k,j,i}^2 = \\ Z_{k,j} d_j & j \text{ is not a satellite} \\ -D_j & \text{otherwise} \end{cases} \text{ for } j \in V_c \cup V_s, k \in V_s \quad (8)$$

$$Q_{k,i,j}^2 \leq k_2 \cdot Y_{k,i,j} \text{ for } i, j \in V_s \cup V_c, i \neq j, k \in V_s \quad (9)$$

$$\sum_{i \in V_s} Q_{i,V_0}^1 = 0 \quad (10)$$

$$\sum_{j \in V_c} Q_{k,j,k}^2 = 0 \text{ for } k \in V_s \quad (11)$$

$$Y_{k,i,j} \leq Z_{k,j} \text{ for } i \in V_s \cup V_c, j \in V_c, k \in V_s \quad (12)$$

$$Y_{k,j,i} \leq Z_{k,j} \text{ for } i \in V_s, j \in V_c, k \in V_s \quad (13)$$

$$\sum_{i \in V_s \cup V_c} Y_{k,i,j} = Z_{k,j} \text{ for } k \in V_s, j \in V_c, i \neq k \quad (14)$$

$$\sum_{i \in V_s} Y_{k,j,k} = Z_{k,j} \text{ for } k \in V_s, j \in V_c, i \neq k \quad (15)$$

$$\sum_{i \in V_s} Z_{i,j} = 1 \text{ for } j \in V_c \quad (16)$$

$$Y_{k,i,j} \leq \sum_{l \in V_s \cup V_0} X_{k,l} \text{ for } k \in V_s, i, j \in V_c \quad (17)$$

$$Y_{k,i,j} \in \{0,1\}, Z_{k,l} \in \{0,1\} \text{ for } k \in V_s, i, j \in V_s \cup V_c, l \in V_c \quad (18)$$

$$X_{k,j} \in Z^+ \text{ for } k, j \in V_s \cup V_0 \quad (19)$$

$$Q_{i,j}^1 \geq 0 \text{ for } i, j \in V_s \cup V_0; Q_{k,i,j}^2 \geq 0 \text{ for } i, j \in V_s \cup V_c, k \in V_s \quad (20)$$

$$Ds_k = \sum_{l \in V_c} (d_j \cdot Z_{k,j}) \text{ for } k \in V_s$$

$$\sum_{i,j \in S_c} Y_{k,i,j} \leq |S_c| - 1 \text{ for } S_c \subset V_c, 2 \leq |S_c| \leq |V_c| - 2 \quad (21)$$

$$Q_{k,i,j}^2 \leq (k_2 - d_j) \cdot Y_{k,i,j} \text{ for } i, j \in V_c, k \in V_s \quad (22)$$

$$Q_{k,i,j}^2 - \sum_{l \in V_s} Q_{k,j,l}^2 \leq (k_2 - d_j) \cdot Y_{k,i,j} \text{ for } i, j \in V_c, k \in V_s \quad (23)$$

5 Numerical experiments

In order to verify and evaluate the proposed approach, many numerical experiments were performed for both illustrative examples.

A Cost optimisation of supply chain with multimodal transport

All the examples relate to the supply chain with two manufacturers ($i = 1..2$), three distributors ($s = 1..3$), five customers ($j = 1..5$), three modes of transport ($d = 1..3$) and 10 types of products ($k = 1..10$). Other parameter values are shown in Appendix A1. The first series of experiments were designed to show the advantages of the hybrid approach used.

The experiments began with six examples: E1, E2, E3, E4, E5 and E6 for the problem formulated in MILP (Section 4) [17]. Two approaches were used to implement the proposed model: mathematical programming (LINGO) and the hybrid approach (LINGO, Eclipse, transformation). The examples E1 .. E6 varied in terms of the number of orders (No). The set of all orders for calculation examples is given in Appendix A1. The experiments were conducted to optimise examples E7 and E8, which are implementations of the hybrid model (with logical constraints) in the hybrid approach.

The implementation of logic constraints for the hybrid model was as follows: product $k = 5$ cannot be distributed with product $k = 6$; product $k = 2$ cannot be distributed with product $k = 8$; and these products cannot be produced together. The results in the form of the objective function, computation time, number of discrete decision variables and constraints are shown in Table 5 and Appendix A2.

Table 5. The results of numerical examples for both approaches

E(No)	MILP-LINGO				MILP-Hybrid			
	Fc	T	V	C	Fc	T	V	C
E1(6)	7764	4	1389	1405	7764	1	155	174
E2(9)	17043*	600**	1389	1567	17039	235	182	177
E3(12)	24106	600**	1389	1729	24106	5	215	177
E4(18)	35772	600**	1389	2053	35772	3	305	178
E5(24)	46481	600**	1389	2377	46481	11	370	178
E6(30)	48946*	600**	1389	2701	48006	8	450	178
P(No)	Hybrid-Hybrid							
	Fc	T	V	C				
E7(12)	24359	45	235	207				
E8(18)	35792	136	325	208				
Fc	The optimal value of the objective function							
T	Solution finding time							
V/C	The number of integer variables/constraints							
*	The feasible value of the objective function after the time T							
**	Calculation was stopped after T = 600s							

The analysis of the outcome indicates that the hybrid approach provided better results in terms of the time needed to find the solution in each case, and to obtain the optimal solution in some cases, which was impossible to do within the acceptable time limits using the traditional approaches.

B Two-Echelon Capacitated Vehicle Routing Problem

For the final validation of the proposed hybrid approach, the benchmark (2E-CVRP) was selected. 2E-CVRP, a well described and widely discussed problem, corresponded with the issues to which our hybrid approach was applied.

The instances for computational examples were built from the existing instances for CVRP [24] denoted as E-n13-k4. All the instance sets can be downloaded from the website [25]. The instance set was composed of five small-sized instances with one depot, 12 customers and two satellites. The full instance consisted of 66 small-sized instances because the two satellites were placed over 12 customers in all 66 possible ways (number of combinations: 2 out of 12). All the instances had the same position for depot and customers, whose coordinates were the same as those of instance E-n13-k4. Small-sized instances differed in the choice of two customers who were also satellites (En13-k4-2 (1,3), En13-k4-6 (1,6), En13-k4-61 (9,10), etc.).

The analysis of the results for the benchmark instances demonstrates that the hybrid approach may be a superior approach to the classical mathematical programming. For all examples, the solutions were found 2–16 times faster than they are in the classical approach. As the presented benchmark was formulated as an MILP problem, the HSF was tested for the solution efficiency. Owing to the hybrid approach, the 2E-CVRP models can be extended over logical, non-linear and other constraints.

Table 6. The results of numerical examples for both approaches

E-n13-k4	MILP-LINGO				MILP-Hybrid			
	Fc	T	V	C	Fc	T	V	C
En13-k4-13	288	600*	368	1262	288	342	186	1024
En13-k4-6	230	125	368	1262	230	55	186	1024
En13-k4-9	244	153	368	1262	244	44	186	1024
En13-k4-20	276	535	368	1262	276	32	186	1024
En13-k4-61	338	6648	368	1262	338	407	186	1024
Fc	The optimal value of the objective function							
T	Time of finding solution							
V/C	The number of integer variables/constraints							
*	The feasible value of the objective function after the time T							

6 Conclusion and discussion on possible extension

The efficiency of the proposed approach is based on the reduction of the combinatorial problem and using the best properties of both environments. The hybrid approach (Tables 5, 6) makes it possible to find better solutions in the shorter time.

In addition to solving larger problems faster, the proposed approach provides virtually unlimited modelling options.

Therefore, the proposed solution is recommended for decision-making problems that have a structure similar to the presented models (Section 4). This structure is characterised by the constraints and objective function in which the decision variables are added together. Further work will focus on running the optimisation models with non-linear and logical constraints, multi-objective, uncertainty and so on in the hybrid optimisation framework.

7 References

- [1] 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, pp. 4431–4454.
- [2] Perea-Lopez E., Ydstie B.E., Grossmann I.E. - *A model predictive control strategy for supply chain optimization*. Computers and Chemical Engineering 27, 2003, pp. 1201–1218.
- [3] Christian Lang J. - *Production and Operations Management: Models and Algorithms*. Production and Inventory Management with Substitutions, Lecture Notes in Economics and Mathematical Systems, Volume 636, 2010, pp 9–79.
- [4] Dang Q., Nielsen I.A., Steger-Jensen K., Madsen O. - *Scheduling a Single Mobile Robot for Part-Feeding Tasks of Production Lines*. Journal of Intelligent Manufacturing, 2013, DOI: 10.1007/s10845-013-0729-y.
- [5] Apt K., Wallace M. - *Constraint Logic Programming using Eclipse*. Cambridge University Press, 2006.
- [6] 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, pp. 728-737.

[7] Bocewicz G., Wójcik R., Banaszak Z. - *AGVs distributed control subject to imprecise operation times*, In: Agent and Multi-Agent Systems: Technologies and Applications, Lecture Notes in Artificial Intelligence, LNAI, Springer-Verlag, Vol. 4953, 2008, pp. 421-430.

[8] Sitek P., Zaborowski M. - *Grouping products in a follow-up production control system for parallel partitioned flow production lines*. Intelligent manufacturing systems IMS 2001: 6th IFAC Workshop, Pergamon, 2001, New York, pp.122-126.

[9] Sitek P., Wikarek J. - *The concept of decision support system structures for the distribution center*. MPER (Management and Production Engineering Review), Vol. 1, No. 3, 2010, pp.63-69.

[10] Sitek P., Wikarek J. - *Cost optimization of supply chain with multimodal transport*. Federated Conference on Computer Science and Information Systems (FedCSIS), 2012, pp. 1111-1118.

[11] Sitek P., Wikarek J. - *Supply chain optimization based on a MILP model from the perspective of a logistics provider*. Management and Production Engineering Review, 2012, pp. 49-61.

[12] Sitek P., Wikarek J. - *The Declarative Framework Approach to Decision Support for Constrained Search Problems*. INTECH, 2011, pp 163-182.

[13] Jain V., Grossmann I.E. - *Algorithms for hybrid MILP/CP models for a class of optimization problems*. INFORMS Journal on Computing 13(4), 2001, pp. 258-276.

[14] Milano M., Wallace M. - *Integrating Operations Research in Constraint Programming*. Annals of Operations Research, Vol. 175, Issue 1, 2010, pp. 37 - 76.

[15] 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, pp. 6-20.

[16] 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, pp. 59-87.

[17] Sitek P., Wikarek J. - *A hybrid approach to supply chain modeling and optimization*. Federated Conference on Computer Science and Information Systems (FedCSIS), 2013, pp. 1223-1230.

[18] Perboli G., Tadei R., Vigo D. - *The Two-Echelon Capacitated Vehicle Routing Problem: Models and Math-Based Heuristics*. Transportation Science, Vol. 45, 2011, pp. 364-380.

[19] Crainic T., Ricciardi, N., Storchi, G. - *Advanced freight transportation systems for congested urban areas*. Transportation Research part C 12, 2004, pp. 119-137.

[20] Schrijver A. - *Theory of Linear and Integer Programming*. ISBN 0-471-98232-6, John Wiley & Sons, 1998.

[21] www.eclipse.org

[22] www.lindo.com

[23] Ricciardi N., Tadei R., Grosso A. - *Optimal facility location with random throughput costs*. Computers and Operations Research 29 (6), 2002, pp. 593-607.

[24] Christofides N., Elion S. - *An algorithm for the vehicle dispatching problem*. Operational Research Quarterly 20, 1969, pp. 309-318.

[25] <http://www.orgroup.polito.it/>

[26] Rossi F., Van Beek P., Walsh T. - *Handbook of Constraint Programming (Foundations of Artificial Intelligence)*, Elsevier Science Inc. New York, NY, USA © 2006.

[27] Relich M. - *Project prototyping with application of CP-based approach*. Management, 2011, Vol. 15, No. 2, pp. 364-377.

APPENDIX A1

Table 7. Data for computational examples E1, E2, E3, E4, E5, E6, E7 and E8

k	V _k	j	s	F _s	V _s		
					E ₁ -E ₃ , E ₇	E ₄ -E ₆ , E ₈	
P1	1	1	C1	600	200	800	
P2	1	2	C2	700	200	800	
P3	3	3	C3	900	200	1000	
P4	2	4					
P5	3	5					
P6	1						
P7	1	i	d	Pt _s	Zt _s		Od _d
P8	3	F1	S1	10	E ₁ -E ₃ , E ₇	E ₄ - E ₆ , E ₈	
P9	2	F2	S2	20	30	60	125
P10	3		S3	40	20	20	180
					10	20	240

i	s	d	K _{i,s,d}	T _{i,s,d}
F1	C1	S2	2	3
F1	C1	S3	4	4
F1	C2	S2	4	2
F1	C2	S3	8	3
F1	C3	S2	6	2
F1	C3	S3	8	3

i	k	W _{i,k}	C _{i,k}
F1	P1	350	10
F1	P2	300	40
F1	P3	500	30
F1	P4	600	40
F1	P5	400	50
F1	P6	300	60

F2	C1	S2	5	4
F2	C1	S3	7	4
F2	C2	S2	2	6
F2	C2	S3	4	7
F2	C3	S2	2	6
F2	C3	S3	3	6

F2	P5	400	50
F2	P6	300	60
F2	P7	400	70
F2	P8	500	80
F2	P9	600	90
F2	P10	650	90

APPENDIX A2

Table 8. Results of optimisation for computational examples E1, E2, E3, E4, E5, E6, E7 and E8

Example E1 $Fc^{opt} = 7764$

Name	i	k	s	j	d1	d2	X_{iskd1}	Y_{sjkd2}
Z_01	F1	P1	C1	M1	S3	S2	10.00	10.00
Z_02	F1	P2	C1	M2	S3	S1	10.00	10.00
Z_03	F1	P3	C1	M3	S3	S2	25.00	25.00
Z_04	F1	P4	C1	M4	S3	S2	30.00	30.00
Z_05	F1	P5	C1	M5	S3	S2	10.00	10.00
Z_06	F1	P6	C1	M1	S3	S1	15.00	10.00
Z_06	F1	P6	C1	M1	S3	S2		5.00

i	s	d	$Xb_{i,s,d}$
F1	C1	S3	5

s	J	d	$Yb_{s,j,d}$	s	j	d	$Yb_{s,j,d}$
C1	M1	S1	1	C1	M3	S2	4
C1	M1	S2	1	C1	M4	S2	3
C1	M2	S1	1	C1	M5	S2	2

s	j	d	$K_{s,j,d}$	$T_{s,j,d}$	s	j	d	$K_{s,j,d}$	$T_{s,j,d}$
C1	M1	S1	2	1	C2	M3	S2	6	2
C1	M1	S2	4	2	C2	M4	S1	3	1
C1	M2	S1	2	1	C2	M4	S2	6	2
C1	M2	S2	4	2	C2	M5	S1	3	1
C1	M3	S1	2	1	C2	M5	S2	6	2
C1	M3	S2	4	2	C3	M1	S1	4	1
C1	M4	S1	2	1	C3	M1	S2	8	2
C1	M4	S2	4	2	C3	M2	S1	4	1
C1	M5	S1	2	1	C3	M2	S2	8	2
C1	M5	S2	4	2	C3	M3	S1	4	1
C2	M1	S1	3	1	C3	M3	S2	8	2
C2	M1	S2	6	2	C3	M4	S1	4	1
C2	M2	S1	3	1	C3	M4	S2	8	2
C2	M2	S2	6	2	C3	M5	S1	4	1
C2	M3	S1	3	1	C3	M5	S2	8	2

Example E2 $Fc^{opt} = 17039$

Name	i	k	s	j	d1	d2	X_{iskd1}	Y_{sjkd2}
Z_01	F1	P1	C1	M1	S3	S2	10.00	10.00
Z_02	F1	P2	C2	M2	S3	S1	10.00	10.00
Z_03	F1	P3	C1	M3	S3	S2	16.00	16.00
Z_03	F1	P3	C2	M3	S3	S2	9.00	9.00
Z_04	F1	P4	C1	M4	S3	S2	30.00	30.00
Z_05	F1	P5	C2	M5	S3	S2	1.00	10.00
Z_05	F2	P5	C2	M5	S3	S2	9.00	
Z_06	F1	P6	C1	M1	S3	S1	2.00	5.00
Z_06	F2	P6	C1	M1	S3	S1	13.00	
Z_06	F2	P6	C1	M1	S3	S2	10.00	10.00
Z_07	F2	P7	C2	M2	S3	S2	20.00	20.00
Z_08	F2	P8	C1	M3	S3	S2	2.00	2.00
Z_08	F2	P8	C2	M3	S3	S2	23.00	23.00
Z_09	F2	P9	C1	M4	S3	S2	30.00	30.00

i	s	d	$Xb_{i,s,d}$	i	s	d	$Xb_{i,s,d}$
F1	C1	S3	3	F2	C1	S3	2
F1	C2	S3	1	F2	C2	S3	3

k	i	k
P5	F1	P6
P5	F2	P6
P2	F1	P8
P2	F2	P8

k	s	k
P5	C1	P6
P5	C2	P6
P5	C3	P6
P2	C1	P8
P2	C2	P8
P2	C3	P8

s	k	$T_{s,k}$	s	k	$T_{s,k}$	s	k	$T_{s,k}$
C1	P1	2	C2	P1	1	C3	P1	3
C1	P2	2	C2	P2	1	C3	P2	3
C1	P3	2	C2	P3	1	C3	P3	3
C1	P4	2	C2	P4	1	C3	P4	3
C1	P5	2	C2	P5	1	C3	P5	3
C1	P6	2	C2	P6	1	C3	P6	3
C1	P7	2	C2	P7	1	C3	P7	3
C1	P8	2	C2	P8	1	C3	P8	3
C1	P9	2	C2	P9	1	C3	P9	3
C1	P10	2	C2	P10	1	C3	P10	3

s	j	d	$Yb_{s,j,d}$	s	j	d	$Yb_{s,j,d}$
C1	M1	S1	1	C2	M2	S1	1
C1	M1	S2	1	C2	M2	S2	1
C1	M3	S2	3	C2	M3	S2	5
C1	M4	S2	6	C2	M5	S2	2

Name	k	j	$T_{i,k}$	$Z_{i,k}$	Name	k	j	$T_{i,k}$	$Z_{i,k}$
Z_01	p1	m1	8	10	Z_11	p1	m3	8	15
Z_02	p2	m2	12	10	Z_12	p2	m4	12	20
Z_03	p3	m3	10	25	Z_13	p3	m5	10	25
Z_04	p4	m4	8	30	Z_14	p4	m1	8	30
Z_05	p5	m5	12	10	Z_15	p5	m2	12	30
Z_06	p6	m1	8	15	Z_16	p6	m3	8	15
Z_07	p7	m2	12	20	Z_17	p7	m4	12	20
Z_08	p8	m3	10	25	Z_18	p8	m5	10	25
Z_09	p9	m4	8	30	Z_19	p9	m1	8	30
Z_10	p10	m5	12	30	Z_20	p10	m2	12	35

Example E3 $Fc^{opt} = 24106$

Name	i	k	s	j	d1	d2	X_{iskd1}	Y_{sjkd2}
Z_01	F1	P1	C1	M1	S3	S1	25.00	10.00
Z_11	F1	P1	C1	M3	S3	S1		15.00
Z_11	F1	P1	C1	M3	S3	S2		
Z_02	F1	P2	C1	M2	S3	S1	30.00	10.00
Z_12	F1	P2	C1	M4	S3	S2		20.00
Z_03	F1	P3	C1	M3	S3	S1	25.00	3.00
Z_03	F1	P3	C1	M3	S3	S2		22.00
Z_04	F1	P4	C1	M4	S3	S1	30.00	20.00
Z_04	F1	P4	C1	M4	S3	S2		10.00
Z_06	F1	P6	C1	M1	S3	S1	10.00	10.00
Z_06	F2	P6	C2	M1	S2	S1	5.00	5.00
Z_05	F2	P5	C3	M5	S3	S2	10.00	10.00
Z_07	F2	P7	C3	M2	S3	S2	20.00	20.00
Z_08	F2	P8	C2	M3	S2	S1	5.00	5.00
Z_08	F2	P8	C2	M3	S3	S2	20.00	20.00
Z_09	F2	P9	C2	M4	S2	S1	30.00	30.00
Z_10	F2	P10	C2	M5	S3	S2	20.00	20.00
Z_10	F2	P10	C3	M5	S3	S2	10.00	10.00

i	s	d	$Xb_{i,s,d}$	i	s	d	$Xb_{i,s,d}$
F1	C1	S3	5	F2	C2	S3	3
F2	C2	S2	4	F2	C3	S3	2

s	j	d	$Yb_{s,j,d}$	s	j	d	$Yb_{s,j,d}$
C1	M1	S1	2.000000	C2	M3	S1	2.000000
C1	M2	S1	1.000000	C2	M3	S2	3.000000
C1	M3	S1	1.000000	C2	M4	S1	6.000000
C1	M3	S2	4.000000	C2	M5	S2	3.000000
C1	M4	S1	4.000000	C3	M2	S2	1.000000
C1	M4	S2	2.000000	C3	M5	S2	3.000000
C2	M1	S1	1.000000				

Example E4 $Fc^{opt} = 35772$

Name	i	k	s	j	d1	d2	X_{iskd1}	Y_{sjkd2}
Z_01	F1	P1	C1	M1	S3	S1	25.00	7.00
Z_01	F1	P1	C1	M1	S3	S2		3.00
Z_11	F1	P1	C1	M3	S3	S2		15.00
Z_02	F1	P2	C1	M2	S3	S2	30.00	10.00
Z_12	F1	P2	C1	M4	S3	S2		20.00
Z_03	F1	P3	C1	M3	S3	S2	50.00	25.00
Z_13	F1	P3	C1	M5	S3	S2		25.00
Z_14	F1	P4	C1	M1	S3	S1	60.00	4.00
Z_14	F1	P4	C1	M1	S3	S2		26.00
Z_04	F1	P4	C1	M4	S3	S2		30.00
Z_15	F1	P5	C1	M2	S3	S2	40.00	30.00
Z_05	F1	P5	C1	M5	S3	S2		10.00
Z_06	F1	P6	C1	M1	S3	S1	30.00	15.00
Z_16	F1	P6	C1	M3	S3	S2		15.00
Z_17	F2	P7	C1	M4	S3	S1	20.00	20.00
Z_07	F2	P7	C2	M2	S2	S2	20.00	20.00
Z_08	F2	P8	C1	M3	S3	S2	50.00	25.00
Z_18	F2	P8	C1	M5	S3	S2		25.00
Z_09	F2	P9	C1	M4	S3	S2	30.00	30.00
Z_10	F2	P10	C1	M5	S3	S1	30.00	30.00

i	s	d	$Xb_{i,s,d}$	i	s	d	$Xb_{i,s,d}$
F1	C1	S3	12.00	F2	C2	S2	1.00
F2	C1	S3	8.00				

s	j	d	$Yb_{s,j,d}$	s	j	d	$Yb_{s,j,d}$
C1	M1	S1	3.00	C1	M4	S2	7.00
C1	M1	S2	3.00	C1	M5	S1	9.00
C1	M2	S2	5.00	C1	M5	S2	9.00
C1	M3	S2	9.00	C2	M2	S2	1.00
C1	M4	S1	2.00				

Example E5 $Fc^{opt} = 46481$

Name	i	k	s	j	d1	d2	X_{iskd1}	Y_{sjkd2}
Z_01	F1	P1	C1	M1	S3	S2	27.00	10.00
Z_11	F1	P1	C1	M3	S3	S1		2.00
Z_11	F1	P1	C1	M3	S3	S2		13.00
Z_21	F1	P1	C1	M5	S3	S1	31.00	2.00
Z_20	F2	P10	C1	M2	S3	S2		1.00
Z_10	F2	P10	C1	M5	S3	S1	21.00	30.00
Z_20	F2	P10	C2	M2	S2	S1		1.00
Z_20	F2	P10	C2	M2	S3	S1	13.00	34.00
Z_02	F1	P2	C1	M1	S3	S2	31.00	1.00
Z_02	F1	P2	C1	M2	S3	S2		10.00
Z_12	F1	P2	C1	M4	S3	S1		20.00
Z_23	F1	P3	C1	M4	S2	S1	2.00	1.00
Z_03	F1	P3	C1	M3	S2	S2		25.00
Z_03	F1	P3	C1	M3	S3	S2	50	1.00
Z_23	F1	P3	C1	M4	S3	S1		2.00
Z_13	F1	P3	C1	M5	S3	S1		23.00
Z_13	F1	P3	C1	M5	S3	S2	6.00	12.00
Z_04	F1	P4	C1	M4	S2	S1		1.00
Z_04	F1	P4	C1	M4	S3	S1	55.00	30.00
Z_14	F1	P4	C1	M1	S3	S2		18.00
Z_04	F1	P4	C1	M4	S3	S2		1.00
Z_24	F1	P4	C1	M5	S3	S1	40.00	10.00
Z_15	F1	P5	C1	M2	S3	S1		20.00
Z_15	F2	P5	C1	M2	S3	S1	10.00	10.00
Z_15	F1	P5	C1	M2	S3	S2		15.00
Z_06	F1	P6	C1	M1	S2	S2	17.00	3.00
Z_06	F1	P6	C1	M1	S3	S2		4.00
Z_06	F1	P6	C1	M1	S3	S1	5.00	4.00
Z_06	F2	P6	C2	M1	S2	S1		8.00
Z_07	F2	P7	C1	M2	S3	S1	12.00	10.00
Z_07	F2	P7	C1	M2	S3	S2		2.00
Z_17	F2	P7	C2	M4	S2	S2	27.00	20.00
Z_07	F2	P7	C2	M2	S2	S1		8.00
Z_07	F2	P7	C2	M2	S3	S1	1.00	21.00
Z_08	F2	P8	C1	M3	S3	S1		4.00
Z_08	F2	P8	C1	M3	S3	S2	40.00	15.00
Z_08	F2	P8	C1	M5	S3	S2		10.00
Z_18	F2	P8	C2	M5	S2	S1	10.00	10.00
Z_01	F2	P9	C1	M1	S3	S1	46.00	13.00
Z_19	F2	P9	C1	M1	S3	S2		3.00
Z_09	F2	P9	C1	M4	S3	S2	14.00	30.00
Z_19	F2	P9	C2	M1	S2	S1		14.00

i	s	d	Xb _{i,s,d}	i	s	d	Xb _{i,s,d}
F1	C1	S2	2.00	F2	C2	S2	8.00
F1	C1	S3	11.00	F2	C2	S3	1.00
F2	C1	S3	8.00				

Z_29	F2	P9	C2	M2	S2	S1	2.00	2.00
Z_20	F2	P10	C1	M2	S3	S1	34.00	34.00
Z_30	F2	P10	C2	M1	S2	S1	7.00	2.00
Z_20	F2	P10	C2	M2	S2	S1		1.00
Z_10	F2	P10	C2	M5	S2	S1		2.00
Z_10	F2	P10	C2	M5	S2	S2		28.00
Z_10	F2	P10	C2	M5	S3	S2	26.00	

s	j	d	Yb _{s,i,d}	s	j	d	Yb _{s,i,d}
C1	M1	S1	3.000000	C1	M4	S2	5.000000
C1	M1	S2	4.000000	C1	M5	S1	13.000000
C1	M2	S1	4.000000	C1	M5	S2	6.000000
C1	M2	S2	4.000000	C2	M1	S1	4.000000
C1	M3	S1	8.000000	C2	M2	S1	11.000000
C1	M3	S2	5.000000	C2	M4	S2	1.000000
C1	M4	S1	5.000000	C2	M5	S1	3.000000

i	s	d	Xb _{i,s,d}	i	s	d	Xb _{i,s,d}
F1	C1	S2	6.00	F2	C2	S2	5.00
F1	C1	S3	9.00	F2	C2	S3	3.00
F2	C1	S3	8.00				

Example E6 Fc^{opt} = 48006

Name	i	k	s	j	d1	d2	X _{iskd1}	Y _{sikd2}
Z_01	F1	P1	C1	M1	S2	S2	10.00	10.00
Z_11	F1	P1	C1	M3	S3	S1	17.00	15.00
Z_21	F1	P1	C1	M5	S3	S1		2.00
Z_02	F1	P2	C1	M2	S2	S1	10.00	10.00
Z_22	F1	P2	C1	M1	S3	S2	21.00	1.00
Z_12	F1	P2	C1	M4	S3	S1		20.00
Z_13	F1	P3	C1	M5	S2	S1	25.00	25.00
Z_03	F1	P3	C1	M3	S3	S1	27.00	25.00
Z_23	F1	P3	C1	M4	S3	S1		2.00
Z_04	F1	P4	C1	M4	S2	S1	4.00	1.00
Z_24	F1	P4	C1	M5	S2	S1		1.00
Z_04	F1	P4	C1	M4	S2	S2		29.00
Z_04	F1	P4	C1	M4	S3	S2		57.00
Z_14	F1	P4	C1	M1	S3	S2	30.00	30.00
Z_25	F1	P5	C1	M3	S2	S1		1.00
Z_05	F1	P5	C1	M5	S2	S1	2.00	8.00
Z_05	F1	P5	C1	M5	S3	S1		37.00
Z_15	F1	P5	C1	M2	S3	S1	30.00	30.00
Z_05	F2	P5	C2	M5	S2	S2		2.00
Z_26	F1	P6	C1	M5	S2	S1	13.00	1.00
Z_06	F1	P6	C1	M1	S2	S2		1.00
Z_06	F1	P6	C1	M1	S2	S1		10.00
Z_06	F1	P6	C1	M1	S3	S1	16.00	15.00
Z_16	F1	P6	C1	M3	S3	S2		15.00
Z_06	F2	P6	C2	M1	S2	S1	6.00	4.00
Z_26	F2	P6	C2	M5	S2	S1		2.00
Z_07	F2	P7	C1	M2	S3	S1	35.00	18.00
Z_17	F2	P7	C1	M4	S3	S1		2.00
Z_17	F2	P7	C1	M4	S3	S2		15.00
Z_07	F2	P7	C2	M2	S2	S1	2.00	2.00
Z_27	F2	P7	C2	M3	S3	S2	5.00	2.00
Z_17	F2	P7	C2	M4	S3	S1		3.00
Z_08	F2	P8	C1	M3	S3	S1	21.00	19.00
Z_18	F2	P8	C1	M5	S3	S1		2.00
Z_08	F2	P8	C2	M3	S2	S2	19.00	6.00
Z_28	F2	P8	C2	M4	S2	S1		2.00
Z_18	F2	P8	C2	M5	S2	S2	23.00	23.00
Z_18	F2	P8	C2	M5	S3	S2		12.00
Z_19	F2	P9	C1	M1	S3	S2	60.00	30.00
Z_09	F2	P9	C1	M4	S3	S2		30.00

s	j	d	Yb _{s,i,d}	s	j	d	Yb _{s,i,d}
C1	M1	S1	1.00	C1	M5	S1	11.00
C1	M1	S2	7.00	C2	M1	S1	1.00
C1	M2	S1	22.00	C2	M2	S1	1.00
C1	M3	S1	15.00	C2	M3	S2	1.00
C1	M3	S2	1.00	C2	M4	S1	1.00
C1	M4	S1	3.00	C2	M5	S1	1.00
C1	M4	S2	7.00	C2	M5	S2	8.00

Example E7 Fc^{opt} = 24359

Name	i	k	s	j	d1	d2	X _{iskd1}	Y _{sikd2}
Z_01	F1	P1	C1	M1	S3	S1	5.000	5.000
Z_01	F1	P1	C2	M1	S2	S1	5.00	5.00
Z_11	F1	P1	C2	M3	S2	S2	15.00	15.00
Z_02	F1	P2	C1	M2	S3	S1	10.00	10.00
Z_12	F1	P2	C1	M4	S3	S2	20.00	20.00
Z_03	F1	P3	C1	M3	S3	S2	25.00	25.00
Z_04	F1	P4	C1	M4	S3	S2	30.00	30.00
Z_05	F1	P5	C1	M5	S3	S1	10.00	10.00
Z_06	F2	P6	C2	M1	S2	S1	15.00	15.00
Z_07	F2	P7	C3	M2	S3	S2	20.00	20.00
Z_08	F2	P8	C2	M3	S3	S2	15.00	15.00
Z_08	F2	P8	C3	M3	S3	S1	10.00	10.00
Z_09	F2	P9	C2	M4	S2	S1	30.00	30.00
Z_10	F2	P10	C2	M5	S2	S2	1.00	20.00
Z_10	F2	P10	C2	M5	S3	S2	19.00	
Z_10	F2	P10	C3	M5	S3	S1	10.00	10.00

i	s	d	Xb _{i,s,d}	i	s	d	Xb _{i,s,d}
F1	C1	S3	5	F2	C2	S3	33
F1	C2	S2	1	F2	C2	S3	2
F2	C2	S2	4				

s	j	d	Yb _{s,i,d}	s	j	d	Yb _{s,i,d}
C1	M1	S1	1.00	C2	M3	S2	3.00
C1	M2	S1	1.00	C2	M4	S1	6.00
C1	M3	S2	4.00	C2	M5	S2	3.00
C1	M4	S2	4.00	C3	M2	S2	1.00
C1	M5	S1	3.00	C3	M3	S1	3.00
	M1	S1	2.00	C3	M5	S1	3.00

Example E8 $Fc^{opt} = 24359$

Name	i	k	s	j	d1	d2	X_{iskd1}	Y_{sjkd2}
Z_01	F1	P1	C1	M1	S3	S2	19.00	4.00
Z_11	F1	P1	C1	M3	S3	S1		1.00
Z_11	F1	P1	C1	M3	S3	S2		14.00
Z_01	F1	P1	C2	M1	S3	S1	6.00	6.00
Z_10	F2	P10	C1	M5	S3	S2	30.00	30.00
Z_02	F1	P2	C2	M2	S3	S2	30.00	10.00
Z_12	F1	P2	C2	M4	S3	S2		20.00
Z_03	F1	P3	C1	M3	S3	S1	50.00	3.00
Z_03	F1	P3	C1	M3	S3	S2		22.00
Z_13	F1	P3	C1	M5	S3	S2		25.00
Z_14	F1	P4	C1	M1	S3	S2	58.00	28.00
Z_04	F1	P4	C1	M4	S3	S2		30.00
Z_14	F1	P4	C2	M1	S3	S1	2.00	2.00
Z_15	F2	P5	C2	M2	S2	S2	6.00	30.00
Z_15	F2	P5	C2	M2	S3	S2	34.00	
Z_05	F2	P5	C2	M5	S3	S1	30.00	10.00
Z_06	F1	P6	C1	M1	S3	S2		15.00
Z_16	F1	P6	C1	M3	S3	S1	30.00	15.00
Z_17	F2	P7	C1	M4	S3	S2		20.00

Z_07	F2	P7	C2	M2	S2	S2	2.00	20.00
Z_07	F2	P7	C2	M2	S3	S2	18.00	
Z_08	F2	P8	C1	M3	S3	S1	50.00	25.00
Z_18	F2	P8	C1	M5	S3	S2		25.00
Z_09	F2	P9	C1	M4	S3	S2	30.00	30.00

i	s	d	$X_{b_{i,s,d}}$	i	s	d	$X_{b_{i,s,d}}$
F1	C1	S3	8.00	F2	C2	S2	1.00
F1	C2	S3	1.00	F2	C2	S3	3.00
F2	C1	S3	8.00				

s	j	d	$Y_{b_{s,j,d}}$	s	j	d	$Y_{b_{s,j,d}}$
C1	M1	S2	4.00	C2	M1	S1	1.00
C1	M3	S1	10.00	C2	M2	S2	6.00
C1	M3	S2	4.00	C2	M4	S2	1.00
C1	M4	S2	7.00	C2	M5	S1	3.00
C1	M5	S2	12.00				