Uncertain Data Dependency Constraints in Matrix Models

Carmen Gervet, S Galichet

▶ **To cite this version:**

Carmen Gervet, S Galichet. Uncertain Data Dependency Constraints in Matrix Models. CPAIOR, 2015, Barcelona, Spain. pp.173-181. hal-01742390

HAL Id: hal-01742390

https://hal.umontpellier.fr/hal-01742390v1

Submitted on 24 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Uncertain Data Dependency Constraints in Matrix Models*

C. Gervet and S. Galichet

LISTIC, Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance - Université de Savoie, BP 80439
74944 Annecy-Le-Vieux Cedex France
gervetec,sylvie.galichet@univ-savoie.fr

Abstract. Uncertain data due to imprecise measurements is commonly specified as bounded intervals in a constraint decision or optimization problem. Dependencies do exist among such data, e.g. upper bound on the sum of uncertain production rates per machine, sum of traffic distribution ratios from a router over several links. For tractability reasons existing approaches in constraint programming or robust optimization frameworks assume independence of the data. This assumption is safe, but can lead to large solution spaces, and a loss of problem structure. Thus it cannot be overlooked. In this paper we identify the context of matrix models and show how data dependency constraints over the columns of such matrices can be modeled and handled efficiently in relationship with the decision variables. Matrix models are linear models whereby the matrix cells specify for instance, the duration of production per item, the production rates, or the wage costs, in applications such as production planning, economics, inventory management. Data imprecision applies to the cells of the matrix and the output vector. Our approach contributes the following results: 1) the identification of the context of matrix models with data constraints, 2) an efficient modeling approach of such constraints that suits solvers from multiple paradigms. An illustration of the approach and its benefits are shown on a production planning problem.

Key words: Data uncertainty, data constraints, Interval reasoning, Interval Linear Programs

1 Introduction

Data uncertainty due to imprecise measurements or incomplete knowledge is ubiquitous in many real world applications, such as network design, renewable energy economics, investment and production planning (e.g. [13, 18]). Formalisms such as linear programming or constraint programming have been extended and successfully used to tackle certain forms of data uncertainty. Bounded intervals are commonly used to specify such imprecise parameters, which take the form

* This research was partly support by the Marie Curie CIG grant, FP7-332683

of coefficients in a given constraint relation. In such problems, uncertain data dependencies do exist, such as an upper bound on the sum of uncertain production rates per machine, the sum of traffic distribution ratios from a router over several links. To our knowledge, existing approaches in Operations Research assume independence of the data when tackling real world problems essentially to maintain computational tractability. This assumption is safe in the sense that no potential solution to the uncertain problem is removed. However, the solution set can be very large even if no solution actually holds once the data dependencies are checked, since the problem structure is lost. Thus accounting for possible data dependencies cannot be overlooked.

In this paper we tackle these issues, by identifying the context of matrix models, where we show how constraints over uncertain data can be handled efficiently. Matrix models are of high practical relevance in many combinatorial optimization problems where the uncertain data corresponds to coefficients of the decision variables. Clearly, the overall problem does not need to be itself a matrix model. With the imprecise data specifying cells of an input matrix, the data constraints correspond to restrictions over the data in each column of the matrix. For instance in a production planning problem, the rows would denote the products to be manufactured and the columns the machines available. A data constraint such as an upper bound on the sum of uncertain production rates per machine, applies to each column of the matrix. In this context, we observe that there is a dynamic relationship between the constraints over uncertain data and the decisions variables that quantify the usage of such data. Uncertain data are not meant to be pruned and instantiated by the decision maker. However, decision variables are, and the solver controls their possible values. This leads us to define a notion of relative consistency of uncertain data constraints, in relationship with the decision variables involved, in order to check and infer consistency of such constraints. For instance, if an uncertain input does not satisfy a dependency constraint, this does not imply that the problem has no solution! It tells us that the associated decision variable should be 0, to reflect the fact that the given machine cannot produce this input.

Our main contribution lies in identifying the context of matrix models, to study the efficient handling of uncertain data constraints. To our knowledge, this is a first efficient handling of uncertain data constraints in combinatorial problems. Our approach contributes the following within this context: 1) identify the role of uncertain data constraints and their impact on the decision variables, 2) propose a new consistency notion of the uncertain data constraints and a model that implements it efficiently. We illustrate the benefits and impacts of our approach on a classical production planning problem with data constraints.

The paper is structured as follows. Section 2 summarizes the related work, while Section 3 gives the intuition of our approach. Section 4 defines our approach, and Section 5 illustrates it. A conclusion is given in Section 6.

2 Related work

In the past 15 years, the generic CSP formalism has been extended to account for forms of uncertainty: e.g. numerical, mixed, quantified, fuzzy, uncertain CSP and CDF-interval CSPs [6]. The *numerical, uncertain, or CDF-interval* CSPs, extend the classical CSP to approximate and reason with continuous uncertain data represented by intervals; see the real constant type in Numerica [19] or the bounded real type in ECLiPSe [7]. The solution sets produced can be very large. This led to some research to extract the relationship between uncertain data that satisfy dependency constraints and possible solutions by applying regression analysis techniques [11]. The fuzzy and mixed CSP [9] coined the concept of uncontrollable variables, that can take a set of values but their domain is not meant to be pruned during problem solving (unlike decision variables). Some constraints over uncontrollable variables can be expressed and thus some limited form of data dependency modeled, mainly in a discrete environment.

The general QCSP formalism introduces universal quantifiers where the domain of a universally quantified variable (UQV) is not meant to be pruned, and its actual value is unknown a priori. There has been work on QCSP with continuous domains, using one or more UQV and dedicated algorithms [2, 5, 15]. Discrete QCSP algorithms cannot be used to reason about uncertain data since they apply a preprocessing step enforced by the solver `QCSPsolve` [10], which essentially determines whether constraints of the form $\forall X, \forall Y, C(X, Y)$, and $\exists Z, \forall Y, C(Z, Y)$, are either always true or false for all values of a UQV. This is a too strong statement, that does not reflect the fact that the data will be refined later on and might satisfy the constraint.

Closer to our approach are the fields of Interval Linear Programming [14, 8] and Robust Optimization [3, 4], whereby in the former we seek the solution set that encloses all possible solutions whatever the data might be, and in the latter the solution that holds in the larger set of possible data realization. They do offer a sensitivity analysis to study the solution variations as the data changes. However, to our knowledge, uncertain data constraints have been ignored for computational tractability reasons.

3 Intuition

The main novel idea behind our work is based on the study of a problem structure. We identify the context of matrix models where uncertain data correspond to coefficients of the decision variables, and the constraints over these apply to the columns of the input matrix. Such data constraints state restrictions *on the possible usage of the data*, and we show how their satisfaction can be handled efficiently in relationship with the corresponding decision variables.

In this context, the role and handling of uncertain data constraints is to determine "which data can be used, to build a solution to the problem". This is in contrast with standard constraints over decision variables, which role is to determine "what value can a variable take to derive a solution that holds". We illustrate the context and our new notion of uncertain data constraint satisfaction on a production planning problem inspired from [12].

Example 1. Three types of products are manufactured, P_1, P_2, P_3 on two different machines M_1, M_2. The production rate of each product per machine is imprecise and specified by intervals. Each machine is available 9 hrs per day, and an expected demand per day is specified by experts as intervals. Furthermore we know that the total production rate of each machine cannot exceed 7 pieces per hour. We are looking for the number of hours per machine for each product, to satisfy the expected demand. An instance data model is given below.

Product	Machine M1	Machine M2	Expected demand
P_1	$[2,3]$	$[5,7]$	$[28,32]$
P_2	$[2,3]$	$[1,3]$	$[25,30]$
P_3	$[4,6]$	$[2,3]$	$[31,37]$

The uncertain CSP model is specified as follows:

$$[2,3] * X_{11} + [5,7] * X_{12} = [28,32] \tag{1}$$

$$[2,3] * X_{21} + [1,3] * X_{22} = [25,30] \tag{2}$$

$$[4,6] * X_{31} + [2,3] * X_{32} = [31,37] \tag{3}$$

$$\forall j \in \{1,2\} : X_{1j} + X_{2j} + X_{3j} \leq 9 \tag{4}$$

$$\forall i \in \{1,2,3\}, \forall j \in \{1,2\} : X_{ij} \geq 0 \tag{5}$$

Uncertain data constraints:

$$a_{11} \in [2,3], a_{21} \in [2,3], a_{31} \in [4,6], \ \ a_{11} + a_{21} + a_{31} \leq 7 \tag{6}$$

$$a_{12} \in [5,7], a_{22} \in [1,3], a_{32} \in [2,3], \ \ a_{12} + a_{22} + a_{32} \leq 7 \tag{7}$$

Consider a state of the uncertain CSP such that $X_{11} = 0$. The production rate of machine M_1 for product P_1 becomes irrelevant since $X_{11} = 0$ means that machine M_1 does not produce P_1 at all in this solution. The maximum production rate of M_1 does not change but now applies to P_2 and P_3. Thus $X_{11} = 0$ infers $a_{11} = 0$. Constraint (6) becomes:

$$a_{21} \in [2,3], a_{31} \in [4,6], \ \ a_{21} + a_{31} \leq 7 \tag{8}$$

Assume now that we have a different production rate for P_3 on M_1:

$$a_{11} \in [2,3], a_{21} \in [2,3], a_{31} \in [8,10], \ \ a_{11} + a_{21} + a_{31} \leq 7 \tag{9}$$

P_3 cannot be produced by M_1 since $a_{31} \in [8,10] \nleq 7$, the total production rate of M_1 is too little. This does not imply that the problem is unsatisfiable, but that P_3 cannot be produced by M_1. Thus $a_{31} \nleq 7$ yields $X_{31} = 0$ and $a_{31} = 0$. □

4 Our approach

We now formalize our approach: define the context of matrix models we identified and the handling of uncertain data constraints within it.

4.1 Problem definition

Definition 1 (interval data). *An interval data, is an uncertain data, specified by an interval* $[\underline{a}, \overline{a}]$*, where* $\underline{a}$ *(lower bound) and* $\overline{a}$ *(upper bound) are positive real numbers, such that* $\underline{a} \leq \overline{a}$*.*

Definition 2 (Matrix model with column constraints). *A matrix model with uncertain data constraints is a constraint problem or a component of a larger constraint problem that consists of:*

1. *A matrix* (A_{ij}) *of input data, such that each row* i *denotes a given product* P_i*, each column* j *denotes the source of production and each cell* a_{ij} *the quantity of product* i *manufactured by the source* j*. If the input is bounded, we have an interval input matrix, where each cell is specified by* $[\underline{a_{ij}}, \overline{a_{ij}}]$*.*
2. *A set of decision variables* $X_{ij} \in \mathbb{R}^+$ *denoting how many instances of the corresponding input shall be manufactured*
3. *A set of column constraints, such that for each column* j*:* $\Sigma_i[\underline{a_{ij}}, \overline{a_{ij}}]$ @ c_j*, where* @ $\in \{=, \leq\}$*, and* c_j *can be a crisp value or a bounded interval.*

To reason about uncertain matrix models we make use of the robust counterpart transformation of interval linear models into linear ones. We recall it, and define the notion of relative consistency of column constraints.

4.2 Linear transformation

An Interval Linear Program is a Linear constraint model where the coefficients are bounded real intervals [8, 3]. The handling of such models transforms each interval linear constraint into an equivalent set of atmost 2 standard linear constraints. Equivalence means that both models denote the same solution space. We recall the transformations of an ILP into its equivalent LP counterpart.

Property 1 (Interval linear constraint and equivalence). Let all decision variables $X_{il} \in \mathbb{R}^+$, and all interval coefficients be positive as well. The interval linear constraint $C = \Sigma_i[\underline{a_{il}}, \overline{a_{il}}] * X_{il}$ @ $[\underline{c_l}, \overline{c_l}]$ with @ $\in \{\leq, =\}$, is equivalent to the following set of constraints depending on the nature of @. We have:

1. $C = \Sigma_i[\underline{a_{il}}, \overline{a_{il}}] * X_{il} \leq [\underline{c_l}, \overline{c_l}]$ is transformed into: $C = \Sigma_i \underline{a_{il}} * X_{il} \leq \overline{c_l}$
2. $C = \Sigma_i[\underline{a_{il}}, \overline{a_{il}}] * X_{il} = [\underline{c_l}, \overline{c_l}]$ is transformed into:

$$C = \{\Sigma_i \underline{a_{il}} \times X_{il} \leq \overline{c_l} \;\; \wedge \;\; \Sigma_i \overline{a_{il}} * X_{il} \geq \underline{c_l}\}$$

Note that case 1 can take a different form depending on the decision maker risk adversity. If he assumes the highest production rate for the smallest demand (pessimistic case), the transformation would be: $C = \Sigma_i \overline{a_{il}} * X_{il} \leq \underline{c_l}$. The solution set of the robust counterpart contains that of the pessimistic model.

Example 2. Consider the following constraint $a_1 * X + a_2 * Y = [120, 150]$ (case 2), with $a_1 \in [0.2, 0.7], a_2 \in [0.1, 0.35], X, Y \in [0, 1000]$. It is rewritten into the system of constraints: $l_1 : 0.7 * X + 0.35 * Y \geq 120 \wedge l_2 : 0.2 * X + 0.1 * Y \leq 150$.

The transformation procedure also applies to the column constraints, and is denoted `transf`. It evaluates to true or false since there is no variable involved.

4.3 Relative consistency

We now define in our context, the relative consistency of column constraints with respect to the decision variables. At the unary level this means that if $(X_{ij} = 0)$ then $(a_{ij} = 0)$, if $\neg$ $\texttt{transf}(a_{ij}@c_j)$ then $(X_{ij} = 0)$ and if $X_{ij} > 0$ then $\texttt{transf}(a_{ij}@\ c_j)$ is true.

Definition 3 (Relative consistency). *A column constraint $\Sigma_i a_{il}@\ c_l$ over the column l of a matrix $I * J$, is relative consistent w.r.t. the decision variables X_{il} if and only if the following conditions hold (C4. and C5. being recursive):*

C1. $\forall i \in I$ *such that* $X_{il} > 0$, *we have* $\texttt{transf}(\Sigma_i a_{il}\ @\ c_l)$ *is true*
C2. $\forall k \in I$ *such that* $\{\neg\texttt{transf}(\Sigma_{i\neq k} a_{il}@\ c_l)$ *and* $\texttt{transf}(\Sigma_i a_{il}@\ c_l)\}$ *is true, we have* $X_{kl} > 0$
C3. $\forall i \in I$ *such that* X_{il} *we have* $i, \texttt{transf}(a_{il}\ @\ c_l)$ *is true*
C4. $\forall k \in I$, *such that* $\neg\texttt{transf}(a_{kl}\ @\ c_l)$, *we have* $X_{kl} = 0$ *and* $\Sigma_{i\neq k} a_{il}@\ c_l$ *is relative consistent*
C5. $\forall k \in I$, *such that* $X_{kl} = 0$, *we have* $\Sigma_{i\neq k} a_{il}@\ c_l$ *is relative consistent*

Example 3. Consider the Example 1. It illustrates C4. and C5, leading to the recursive call to C3. Let us assume now that the X_{i1} are free, and the column constraint $[2,3]+[2,3]+[4,6] = [7,9]$. Rewritten into $2+2+4 \leq 9, 3+3+6 \geq 7$, we have $X_{31} > 0$, since $3+3 \not\geq 7$ and $3+3+6 \geq 7$. It is relative consistent with $X_{31} > 0$ (C2.).

4.4 Column constraint model

Our intent is to model column constraints and infer relative consistency while preserving the computational tractability of the model. We do so by proposing a Mixed Integer Interval model of a column constraint. We show how it allows us to check and infer relative consistency efficiently. This model can be embedded in a larger constraint model. The consistency of the whole constraint system is inferred from the local and relative consistency of each constraint.

Modeling column constraints Consider the column constraint over column l: $\Sigma_i[\underline{a_{il}}, \overline{a_{il}}]\ @\ c_l$. It needs to be linked with the decision variables X_{il}. Logical implications could be used, but they would not make an active use of consistency and propagation techniques. We propose an alternative MIP model.

To each data we associate a Boolean variable. Each indicates whether: 1) the data must be accounted for to render the column constraint consistent, 2) the data violates the column constraint and needs to be discarded, 3) the decision variable imposes a selection or removal of the data. Thus the column constraint in transformed state is specified as a scalar product of the data and Boolean variables. The link between the decision variables and their corresponding Booleans is specified using a standard mathematical programming technique that introduces a big enough positive constant K, and a small enough constant λ.

Theorem 1 (column constraint model). *Let $X_{il} \in \mathbb{R}^+$ be decision variables of the matrix model for column l. Let $\mathbf{B_{il}}$ be Boolean variables. Let K be a large positive number, and λ a small enough positive number. A column constraint*

$$\Sigma_i[\underline{a_{il}}, \overline{a_{il}}] \ @ \ c_l$$

is relative consistent if the following system of constraints is bounds consistent

$$\texttt{transf}(\Sigma_i[\underline{a_{il}}, \overline{a_{il}}] \times \mathbf{B_{il}} \ @ \ c_l) \tag{10}$$

$$\forall i, 0 \leq X_{il} \leq K \times \mathbf{B_{il}} \tag{11}$$

$$\forall i, \lambda \times \mathbf{B_{il}} \leq X_{il} \tag{12}$$

The proof is omitted for space reasons.

5 Illustration of the approach

We illustrate the approach on the production planning problem. The robust model is specified below. Each interval linear core constraint is transformed into a system of two linear constraints, and each column constraint into its robust counterpart.

For the core constraints we have:
$2 * X_{11} + 5 * X_{12} \leq 32,\ \ 3 * X_{11} + 7 * X_{12} \geq 28,$
$2 * X_{21} + \quad X_{22} \leq 30,\ \ 3 * X_{21} + 3 * X_{22} \geq 25,$
$4 * X_{31} + 2 * X_{32} \leq 37,\ \ 6 * X_{31} + 3 * X_{32} \geq 31,$
$\forall j \in \{1, 2\}, X_{1j} + X_{2j} + X_{3j} \leq 9,$
$\forall i \in \{1, 2, 3\}, \forall j \in \{1, 2\} : X_{ij} \geq 0,$
$\forall i, j, X_{ij} \geq 0, B_{ij} \in \{0, 1\},$

And for the column constraints:
$a_{11} \in [2, 3], a_{21} \in [2, 3], a_{31} \in [4, 6],\ \ a_{11} + a_{21} + a_{31} \leq 7$ and
$a_{12} \in [5, 7], a_{22} \in [1, 3], a_{32} \in [2, 3],\ \ a_{12} + a_{22} + a_{32} \leq 7$ transformed into:

$2 * B_{11} + 2 * B_{21} + 4 * B_{31} \leq 7,$
$5 * B_{12} + \quad B_{22} + 2 * B_{32} \leq 7,$
$\forall i \in \{1, 2, 3\}, j \in \{1, 2\}\ \ 0 \leq X_{ij} \leq K * B_{ij},$
$\forall i \in \{1, 2, 3\}, j \in \{1, 2\}\ \ \lambda * B_{ij} \leq X_{ij}$

We consider three different models: 1) the robust approach that seeks the largest solution set, 2) the pessimistic approach, and 3) the model without column data constraints. They were implemented using the ECLiPSe `ic` interval solver [7]. We used the constants K=100 and $\lambda = 1$. The column constraints in the tightest model take the form: $3 * B_{11} + 3 * B_{21} + 6 * B_{31} \leq 7$ and $7 * B_{12} + 3 * B_{22} + 3 * B_{32} \leq 7$.

The solution set results are summarized in the following table with real values rounded up to hundredth for clarity. The tightest model, where the decision maker assumes the highest production rates has no solution.

Variables	With column constraints			Without column constraints
	Robust model		Tightest model	
	Booleans	Solution bounds	Solution bounds	Solution bounds
X_{11}	0	0.0..0.0	–	0.0 .. 7.00
X_{12}	1	4.0..4.5	–	0.99 .. 6.4
X_{21}	1	3.33..3.84	–	0.33 .. 7.34
X_{22}	1	4.49..5.0	–	0.99 .. 8.0
X_{31}	1	5.16..5.67	–	1.66 ..8.67
X_{32}	0	0.0..0.0	–	0.0 .. 7.0

Results From the table of results we can clearly see that:

1. Enforcing Bounds Consistency (BC) on the constraint system without the column constraints, is safe since the bounds obtained enclose the ones of the robust model with column constraints. However, they are large, and the impact of accounting for the column constraints, both in the much reduced bounds obtained, and to detect infeasibility is shown.
2. The difference between the column and non column constraint models is also interesting. The solutions show that only X_{11} and X_{32} can possibly take a zero value from enforcing BC on the model without column constraints. Thus all the other decision variables require the usage of the input data resources. Once the column constraints are enforced, the input data a_{11} and a_{32} must be discarded since otherwise the column constraints would fail. This illustrates the benefits of relative consistency over column constraints.
3. The tightest model fails, because we can see from the solution without column constraints that a_{21} and a_{31} must be used since their respective X_{ij} are strictly positive in the solution to the model without column constraints. However from the tight column constraint they can not both be used at full production rate at the same time. The same holds for a_{12} and a_{22}.

All computations were performed in constant time given the size of the problem. This approach can easily scale up, since if we have n uncertain data (thus n related decision variables) in the matrix model, our model generates n Boolean variables and $O(2n+2) = O(n)$ constraints. This number does not depend on the size or bounds of the uncertain data domain, and the whole problem models a standard CP or MIP problem, making powerful use of existing techniques.

6 Conclusion and future work

In this paper we have identified the context of matrix models to account for uncertain data constraints efficiently. Such models are common in many applications ranging from production planning, economics, or inventory management to name a few. In this context, we defined the notion of relative consistency, and a model of uncertain data constraints that implements it effectively. An interesting challenge to our eyes, would be to investigate how the notion of relative consistency can be generalized and applied to certain classes of global constraints in a CP environment, whereby the uncertain data appears as coefficients to the decision variables.

References

1. Benhamou, F. *Inteval constraint logic programming.* In Constraint Programming: Basics and Trends. LNCS, Vol. 910, 1-21, Springer, 1995.
2. Benhamou, F. and Goualard, F. *Universally quantified interval constraints.* In Proc. of CP-2000, LNCS 1894, Singapore, 2000.
3. Ben-Tal, A; and Nemirovski, A. *Robust solutions of uncertain liner programs.* Operations Research Letters, 25, 1-13, 1999.
4. Bertsimas, D. and Brown, D. *Constructing uncertainty sets for robust linear optimization.* Operations Research, 2009.
5. Bordeaux L., and Monfroy, E. *Beyond NP: Arc-consistency for quantified constraints.* In Proc. of CP 2002.
6. Brown K. and Miguel I. *Chapter 21: Uncertainty and Change* Handbook of Constraint Programming Elsevier, 2006.
7. Cheadle A.M., Harvey W., Sadler A.J., Schimpf J., Shen K. and Wallace M.G. *ECLiPSe: An Introduction.* Tech. Rep. IC-Parc-03-1, Imperial College London, London, UK.
8. Chinneck J.W. and Ramadan K. *Linear programming with interval coefficients.* J. Operational Research Society, 51(2):209–220, 2000.
9. Fargier H., Lang J. and Schiex T. *Mixed constraint satisfaction: A framework for decision problems under incomplete knowledge.* In Proc. of AAAI-96, 1996.
10. Gent, I., and Nightingale, P., and Stergiou, K. *QCSP-Solve: A Solver for Quantified Constraint Satisfaction Problems.* In Proc. of IJCAI 2005.
11. Gervet C. and Galichet S. *On combining regression analysis and constraint programming.* Proceedings of IPMU, 2014.
12. Inuiguchi, M. and Kume, Y. *Goal programming problems with interval coefficients and target intervals.* European Journl of Oper. Res. 52, 1991.
13. Medina A., Taft N., Salamatian K., Bhattacharyya S. and Diot C. *Traffic Matrix Estimation: Existing Techniques and New Directions.* Proceedings of ACM SIGCOMM02, 2002.
14. Oettli W. *On the solution set of a linear system with inaccurate coefficients.* J. SIAM: Series B, Numerical Analysis, 2, 1, 115-118, 1965.
15. Ratschan, S. *Efficient solving of quantified inequality constraints over the real numbers.* ACM Trans. Computat. Logic, 7, 4, 723-748, 2006.
16. Rossi F., van Beek P., and Walsh T. *Handbook of Constraint Programming.* Elsevier, 2006.
17. Saad A., Gervet C. and Abdennadher S. *Constraint Reasoning with Uncertain Data usingCDF-Intervals* Proceedings of CP'AI-OR, Springer,2010.
18. Tarim, S. and Kingsman, B. *The stochastic dynamic production/inventory lot-sizing problem with service-level constraints.* International Journal of Production Economics 88, 105119,2004.
19. Van Hentenryck P., Michel L. and Deville Y. *Numerica: a Modeling Language for Global Optimization* The MIT Press, Cambridge Mass, 1997.
20. Yorke-Smith N. and Gervet C. *Certainty Closure: Reliable Constraint Reasoning with Uncertain Data* ACM Transactions on Computational Logic 10(1), 2009.