Data Uncertainty in Constraint Programming
A Non-Probabilistic Approach

Neil Yorke-Smith* and Carmen Gervet

IC–Parc, Imperial College, London, SW7 2AZ, U.K., {nys,cg6}@icparc.ic.ac.uk

Abstract The constraint programming paradigm has proved to have the flexibility and efficiency necessary to treat well-defined large-scale optimisation (LSCO) problems. Many real world problems, however, are ill-defined, incomplete, or have uncertain data. Research on ill-defined LSCO problems has centred on modelling the uncertainties by approximating the state of the real world, with no guarantee as a result that the actual problem is being solved. We focus here on ill-defined data, motivated by problems from energy trading and computer network optimisation, where no probability distribution is known or can be usefully obtained. We suggest a non-probabilistic *certainty closure* approach to model the data uncertainty, discuss the formalism and semantics required to build a constraint solving system based on such a computation domain, and give examples in the case of linear systems.

1 Introduction

Uncertainty can arise from many sources. In the real world, we find dynamic environments, over-constrained problems, and partial or incomplete data. The difficulties of a large-scale combinatorial optimisation (LSCO) problem are correspondingly greater when the problem is ill-defined.

Four factors characterise an optimisation problem and uncertainty may feature in any of them: the input (or data), the constraints, the decision criteria, and consequently, the output (or solution). Some aspects have been examined in the literature:

- dynamic environments [1, 2]: new/changing constraints, anticipated change
- over-constrained problems [3, 4]: hard and soft constraints, decision criteria
- probabilistic models and data [5, 6]: incomplete or inconsistent information

The constraint programming (CP) paradigm has been extended to address the first two of these areas, whereas the third has been more traditionally the realm of operational research (OR). Interest in uncertainty within CP has been growing recently [7, 8], but there has remained little work on data uncertainty.

Gervet et al. [9] describe a speculative constraint optimisation project in energy trading. Due to pending market deregulation and in conjunction with other

* Corresponding author

ill-defined problem elements, there was no reasonable probability distribution that could be forecast for the data parameters. Instead, an iterative prototyping approach using simulation was employed.

In network management, by contrast, all the data may be available in theory but may be too costly and voluminous to collect in practice. Moreover, the data (such as traffic flow through a router) fluctuates constantly, and to sample at all points at one instant would be prohibitive; the data may be expected to be inconsistent by the time it is collected. It is of commercial interest to make decisions to optimise the network [10][1], but we must do so based on incomplete and inconsistent information for which no stochastic description is apparent.

These motivating problems from the real world exemplify that we potentially face dynamic, non-deterministic data where, harder still, no stochastic characterisation is available. It is clearly inadequate to address this situation in CP either by ignoring the uncertain data or by assuming the deterministic case.

The experience of traditional OR helps little, on the whole. The approach to uncertain data [11] has been largely to reduce to the deterministic case and perform simulation, or to attach a probability distribution, perhaps derived by forecasting from past data, and consider risk and utility factors. In *robust optimisation*, for instance, variance measures or expected utilities are used; in *stochastic optimisation*, penalties based on expected feasibility of scenarios. In *fuzzy programming* [12], constraints (and hence data) and goals are modelled as fuzzy sets, but now the set membership functions are assumed known.

All of these approaches rely on *a priori* knowledge, or empirical estimation, of a probability distribution for the data or of some other stochastic description of it. Further, rationality of the decision maker is assumed. Not in every situation, as we have seen, are these requirements meaningful or even feasible.

The need for a robust, non-probabilistic alternative has been recognised in OR [13]. As Hoffman writes, "We hope that future research will also address [in addition to incorporating risk] the issue of how to incorporate ... data for which even the mean value is not known and for which one only has range estimates of its value." [11]

In the following sections, we shall speak of uncertainty to mean data uncertainty. We introduce the concept of the certainty closure and apply the methodology to a case study arising from our motivational problems. We then place the work in a wider context and anticipate future steps.

2 The Certainty Closure

We give a definition of the certainty closure and introduce the semantic and algorithmic concerns that follow for data uncertainty in constraint programming. The idea is to enclose the uncertainty and so be able to solve an ill-defined problem without approximation via some mathematical model. Thus we guarantee that the true problem is solved and that the solutions found are robust.

[1] This research has been applied to a commercial product, under development at IC–Parc and Parc Technologies: `www.parc-technologies.com`.

Data uncertainty is reflected as uncertainty in the coefficients of the constraints: what were constant coefficients have become uncertain in value. We suppose that all the constraints are hard and static; such issues have been considered elsewhere (e.g., [3]). An introduction to CP may be found in [14].

Definition 1. *An* uncertain constraint satisfaction problem $\langle \mathcal{V}, \mathcal{D}, \mathcal{C}; \mathcal{U} \rangle$ *is a standard CSP in which the coefficients in the constraints $\mathcal{C}$ may range, not necessarily independently, over an uncertainty set $\mathcal{U}$.*

It is unknown which values in the uncertainty set the data might take, but it is necessary in every case that no solution to the resulting constraints be excluded. This motivates the definition of the certainty closure:

Definition 2. *The* certainty closure *of an uncertain constraint satisfaction problem $\langle \mathcal{V}, \mathcal{D}, \mathcal{C} \rangle$ is the CSP $\langle \mathcal{V}, \mathcal{D}, \mathcal{C}' \rangle$, where $\mathcal{C}'$ is derived from $\mathcal{C}$ as follows. For each constraint $C \in \mathcal{C}$, let $C^{(i)}$ be the $0 < k \leq \infty$ possible constraints resulting as the coefficients in C vary over the uncertainty set of their possible values. Let $\overline{C} = \bigvee_{i=1}^{k} C^{(i)}$ be the disjunction of the $C^{(i)}$. Then let $\mathcal{C}'$ be the conjunction of the constraints $\overline{C}$, wlog removing repeated and trivially redundant constraints.*

That is, the constraints in $\mathcal{C}'$ are the generalisation of those in $\mathcal{C}$ to hold under all possible realisations of the data. We write ρ to denote the certainty closure mapping: $\langle \mathcal{V}, \mathcal{D}, \mathcal{C} \rangle \xrightarrow{\rho} \langle \mathcal{V}, \mathcal{D}, \mathcal{C}' \rangle$. Any solution to the true problem T is guaranteed to be a solution to $\rho(T)$, and no solution of T is excluded.

The idea can be seen in the transformation of the simple uncertain constraint $X \geq a$, where a ranges in $[20, 30]$ ($[\underline{X}, \overline{X}]$ denoting a real interval), under ρ to $X \geq 20$. Satisfaction of the former constraint guarantees satisfaction of the latter, no matter what the true value of the data. A case study is given in Sect. 3.

It might be objected that this is the worst case and over-emphasises pessimistic combinations of data values. In reply, we assume (motivated by Sect. 1) only that all elements of $\mathcal{U}$ are possible; we have no information as to likelihood. Further, random values within $\mathcal{U}$ can be almost as bad as the most difficult values [15].

We do not assume that $\mathcal{U}$ is the direct product of the coefficients of $\mathcal{C}$: the values for the data may be dependent.[2] This is in contrast to probabilistic models, where independence of data parameters is nearly always required.

The certainty closure allows robust inference on the original problem in the sense that the domains resulting for the variables are *sure*: valid whatever the realisation of the data within the uncertainty set. The new CSP will be satisfiable provided the original is satisfiable for some element of $\mathcal{U}$: the system $X \geq [20, 30] \wedge X \leq [20, 25]$ need not be satisfiable, for example, whereas its certainty closure will be.

The solution of an uncertain CSP is derived from its certainty closure such that all possible solutions to the original CSP are contained within. This means

[2] If the dependence is between data in different constraints, the above formulation must be amended slightly.

the decision variables will take a set of possible values, in general, rather than be instantiated to one value, but the domains that result are sure and correct.

The distinction must be made between the *natural* domains of the variables $\mathcal{V}$ and the *calculus* domain which is reached by applying the certainty closure. The former is the domain with which the user interacts and over which constraints are applied; the latter is how we represent the inferences and calculate with surety, and is hidden from the user.

In enhancing the CP paradigm to cope with uncertainty, we do not at the calculus level assume the rationality of the decision maker. Rather than proposing a single instantiation, precluding all others, we return all possible solutions. This contrasts with the fundamental assumption of rationality often made in probabilistic approaches.

The type of constraints found in the energy trading and network optimisation problems of Sect. 1 are principally linear. Motivated by this, consider the case of data uncertainty in a linear problem, formalised as an interval linear system (ILS), as defined below. We follow the notation of Neumaier [16], except to use bold font to denote an interval quantity.

Definition 3. *Let $\mathcal{V}$ be a set of n variables over $\mathbb{R}$, and $\mathcal{C}$ be a set of m linear constraints (equalities or inequalities) for $\mathcal{V}$ in normal form. Let $A \in \mathbb{R}^{m \times n}$ be the matrix of left-hand sides and $b \in \mathbb{R}^m$ be the vector of right-hand sides. Let R be the list of m relations (or constraint symbols), one for each constraint; $R_i \in \{<, \leq, =, \geq, >\}$, $\forall i = 1, \ldots, m$. Then an* interval linear system *induced by $\mathcal{C}$ on $\mathcal{V}$ is a tuple $\langle \mathbf{A}, R, \mathbf{b} \rangle$, where $\mathbf{A} \in \mathbb{IR}^{m \times n}$ is an interval matrix $[\underline{A}, \overline{A}]$ with $A \in \mathbf{A}$, and $\mathbf{b} \in \mathbb{IR}^m$ is an interval vector $[\underline{b}, \overline{b}]$ with $b \in \mathbf{b}$.*

For example, the ILS given by

$$\mathbf{A} = \begin{pmatrix} [-2, 2] & [1, 2] \\ [-2, -1] & -1 \\ 6 & [\frac{3}{2}, 3] \end{pmatrix} \text{ and } \mathbf{b} = \begin{pmatrix} [3, 4] \\ [-5, 5] \\ [4, 15] \end{pmatrix}$$

and $R = (\leq, =, =)^\top$ has the *solution set* $\Sigma(\mathbf{A}, \mathbf{b})$, shown in Fig. 1, which is given by the rays $2X+Y = 3$, $6X+3Y = 15$, $2X-Y = -5$ and $12X+3Y = 8$, and the points $(X, Y) \in \left\{ (0, 4), (\frac{1}{3}, \frac{14}{3}), (\frac{5}{3}, \frac{10}{3}), (\frac{5}{2}, 0), (\frac{23}{6}, -\frac{38}{3}), (\frac{2}{3}, 0) \right\}$. Hence the solution set is non-convex and the *interval hull* $\square\Sigma(\mathbf{A}, \mathbf{b})$, the smallest hyperbox enclosing the solution set, is unbounded.

The above formulation is very general. Following [13], we do not impose what form the uncertainty set might take. Two common choices, from the field of robust computation, are *intervals* and *ellipsoids*. An interval (strictly, a closed non-empty bounded interval) given by lower and upper bounds is the simplest description of an uncertain value, and the properties and uses of interval computation are well-known. Ellipsoids, while more complicated than intervals, arise naturally in problems in engineering [17].

We will make use of intervals for three reasons. First, they describe well the uncertainty in the motivational LSCO problems of Sect. 1: uncoupled data

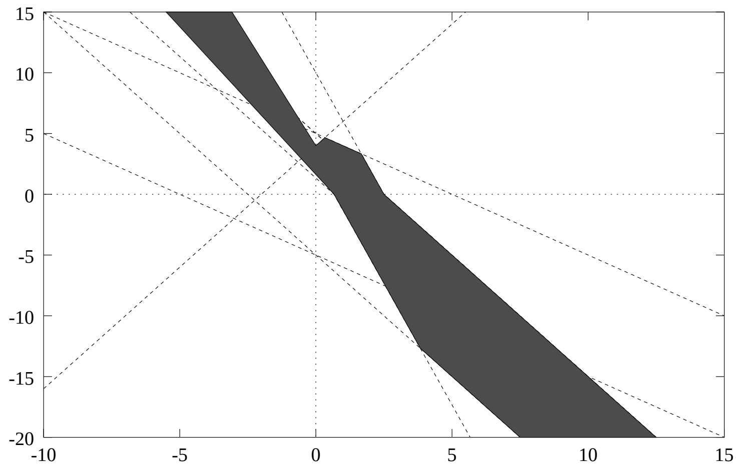

Figure 1. Solution set

known only to be between lower and upper bounds. Second, libraries for interval computation are available in CLP systems. Third, intervals give ease of intuition.

3 Case Study: Positive Orthant Interval Linear Systems

Linear constraints, and systems which can be readily linearised, are sufficient to model many (although far from all) important real-world problems. A case in point is that of network optimisation. The difficulty here is not in the constraints, which largely form a linear flow model, but in the data, which is incomplete. The uncertain model variables are non-negative reals which may be assumed independent, thus motivating the sequel.

Definition 4. *A* positive orthant interval linear system *is an interval linear system in which the natural domain of each variable, $dom(V), V \in \mathcal{V}$, is non-negative. Thus, the solution set lies within the positive orthant of* $\mathbb{R}^n$.

An example is the system shown earlier in Fig. 1, if we impose $X, Y \geq 0$. Such a system $\langle \boldsymbol{A}, R, \boldsymbol{b}\rangle$ is tractable, unlike the general case, because the solution set in the positive orthant, denoted $\Sigma\left(\mathbf{A}, \mathbf{b}\right)_+ = \Sigma\left(\mathbf{A}, \mathbf{b}\right) \cap \mathbb{R}^n_+$ is convex. However, to avoid an exponential growth in the number of faces of the closure to a single constraint, we impose a restriction on equality constraints:

$$\begin{gathered}\forall i = 1, \ldots, m, \text{ if } R_i = (=) \text{ and } 0 \in \boldsymbol{A}_{i,n}, \text{then} \\ \neg\left(\hat{\boldsymbol{A}}_i > 0 \wedge \boldsymbol{b}_i > 0\right) \ \wedge\ \neg\left(\hat{\boldsymbol{A}}_i < 0 \wedge \boldsymbol{b}_i < 0\right)\end{gathered} \qquad (\bigstar)$$

where $\hat{\boldsymbol{A}}_i$ denotes the vector $(\boldsymbol{A}_{i,1}, \ldots, \boldsymbol{A}_{i,n-1})$ of length $n-1$.

The intuition behind the method is that each inequality gives rise to one halfspace (a line, in 2D), and the points which correspond to feasible solutions of

INPUT:	ILS $\langle \boldsymbol{A}, R, \boldsymbol{b} \rangle$ as part of a UCSP $\langle \mathcal{V}, \mathcal{D}, \mathcal{C}; \mathcal{U} \rangle$
OUTPUT:	Bounds for the variables in $\mathcal{V}$
METHOD:	1. Check for suitable form
	2. Rewrite equalities
	3. Transform to certainty closure
	4. Find convex hull
	5. Derive interval hull

Figure 2. Overview of algorithm

all the constraints are those which lie within the intersection of these halfspaces. This intersection is guaranteed (in the positive orthant) to be convex, and (★) permits equalities to be rewritten as a pair of inequalities.

Our motivational problems have no uncertainty in the objective function, and to simplify the presentation we postpone a discussion of optimisation under uncertainty. However, observe that for an ILS with linear objective max $\sum_i \mathbf{c_i} X_i$ (without loss of generality), uncertainty in the objective is easily removed by adding the additional constraint $\sum_i \mathbf{c_i} X_i \geq Z$ for an auxiliary variable Z and optimising max Z. This reduces the problem to the main case.

We will transform the system $\langle \boldsymbol{A}, R, \boldsymbol{b} \rangle$ to its certainty closure, the system $\langle \boldsymbol{A}', R', \boldsymbol{b}' \rangle$. Figure 2 gives the algorithm in outline.

Steps 1 and 2. We require linear constraints with interval uncertainty, obeying (★). Each equality constraint can be replaced by a pair of inequalities, taking care to choose the relational operators correctly.

Step 3. The transformation ρ is given by replacing the matrix $\boldsymbol{A}$ and vector $\boldsymbol{b}$ by non-interval versions as follows.

Proposition 5. *The certainty closure of a positive orthant interval linear system $L = \langle \boldsymbol{A}, R, \boldsymbol{b} \rangle$ satisfying (★) is the numeric linear inequality system $A' x \leq b'$, where $A' \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n$, and $b' \in \mathbb{R}^m$ are given by the following.*

If $0 \in \boldsymbol{A}_{i,n}$, then $((A')_i, b'_i)$ is

$$\begin{cases} \left(\underline{\hat{\boldsymbol{A}}_i}, \overline{\boldsymbol{A}_{i,n}}, \underline{\boldsymbol{b}_i}\right) & \text{if } \{<, \leq\} \in R_i, \\ -\left(\overline{\hat{\boldsymbol{A}}_i}, \underline{\boldsymbol{A}_{i,n}}, \underline{\boldsymbol{b}_i}\right) & \text{if } \{>, \geq\} \in R_i, \end{cases}$$

while if $0 \notin \boldsymbol{A}_{i,n}$, then $((A')_i, b'_i)$ is

$$\begin{cases} \left(\underline{(\boldsymbol{A})_i}, \overline{\boldsymbol{b}_i}\right) & \text{if } \{<, \leq\} \in R_i \text{ and } \underline{\boldsymbol{A}_{i,n}} \geq 0, \\ \left(\overline{(\boldsymbol{A})_i}, \underline{\boldsymbol{b}_i}\right) & \text{if } \{<, \leq\} \in R_i \text{ and } \overline{\boldsymbol{A}_{i,n}} < 0, \\ -\left(\overline{(\boldsymbol{A})_i}, \underline{\boldsymbol{b}_i}\right) & \text{if } \{>, \geq\} \in R_i \text{ and } \underline{\boldsymbol{A}_{i,n}} \geq 0, \\ -\left(\underline{(\boldsymbol{A})_i}, \overline{\boldsymbol{b}_i}\right) & \text{if } \{>, \geq\} \in R_i \text{ and } \overline{\boldsymbol{A}_{i,n}} < 0. \end{cases}$$

Proof. Omitted.

Consider again the example of Sect. 2. From the initial system

$$\mathbf{A} = \begin{pmatrix} [-2,2] & [1,2] \\ [-2,-1] & -1 \\ 6 & [\frac{3}{2},3] \end{pmatrix} \text{ and } \mathbf{b} = \begin{pmatrix} [3,4] \\ [-5,5] \\ [4,15] \end{pmatrix}$$

and $R = (\leq, =, =)^\top$, the transformation to the certainty closure yields

$$A' = \begin{pmatrix} -2 & -2 & 1 & -6 & 6 \\ 1 & -1 & 1 & -3 & \frac{3}{2} \end{pmatrix}^\top \text{ and } b' = \begin{pmatrix} 4 & 5 & 5 & -4 & 15 \end{pmatrix}^\top \ .$$

Hence the positive orthant interval hull is $\square\Sigma\,(\mathbf{A},\mathbf{b})_+ = \left([0,\frac{5}{2}],[0,\frac{14}{3}]\right)$.

By this means, from a constraint problem with interval uncertain data, we obtain a certain constraint problem. By inference on the transformed problem (that is, the certainty closure), sure domains can be found for the variables — domains which include the values that arise under every data realisation — and for an ILS, we give the inference explicitly.

Steps 4 and 5. The certainty closure of an ILS is a halfspace description of a polytope. The action of the constraints on the variable domains is equivalent to the projection of the convex hull of the polytope onto each axis [18]. Geometrically, we seek bounds on the dual representation of the polytope, that is, the bounds of its vertex form. In practice, we can use linear programming to find the interval hull directly.

For each variable X_j, $j=1,\ldots,n$, solve two linear programs which differ only in their objectives, $\min X_j$ and $\max X_j$, subject to the constraints given by the certainty closure. With $2n$ applications of simplex we have tight, sure bounds:

Proposition 6. *Let $\rho(L)$ be the certainty closure of L defined above. The bounds obtained on the domains $\mathcal{D}$ by the method given are the tightest certain bounds possible, and can be computed in expected time $\mathcal{O}(mn^3)$.*

Proof. The transformation from L to $\rho(L)$ can be done in $\mathcal{O}(2mn)$ operations, and its correctness follows from Proposition 5. The constraints in $\rho(L)$ are linear and the solution space of their conjunction intersected with the positive orthant is a convex polytope, possibly unbounded [19]. This polytope is the convex hull $\Sigma\,(\mathbf{A},\mathbf{b})_+$ by construction, and projecting onto each normal e_i immediately gives the interval hull. The bounds so obtained are tight to the solution set since the maximum and minimum possible value of each variable are found by the simplex iterations. The projection can be done with $2n$ iterations, at expected cost $\mathcal{O}(3mn^2)$ each. □

This provides bounds for the natural domains of the variables. It may well be that not every value in the product space of the variable domains is feasible; other constraint inference techniques would be needed to determine this, since no further tightening will be obtained by reasoning solely on their bounds.

The certainty closure for interval linear systems, and the hull inference described above, have been implemented on the ECL^{i}PS^e CLP system [20] using the interval computation library ic and the commercial LP solver XPRESS-MP. For problems up to several hundred variables and constraints, even a straightforward implementation gives the bounds in a few seconds.

We are investigating the application of the certainty closure to the network optimisation problem, and initial findings show promise. The current approach of data correction to the uncertain LSCO has been complemented in:

- proving the correctness or otherwise of current results
- identifying deep inconsistencies, beyond data correction, in the raw data
- understanding the source of bottlenecks in the network

Thus insight is gained of the relationship between the network topology and traffic flow, leading to robust quantitative results and improved global understanding.

4 Related Work

A problem with uncertain data is, in general, much harder than the same problem with certain data, as we might expect. With a linear system, for instance, even checking a possible enclosure for interval uncertain data is NP-hard [17], and all methods for the general problem are exponential in the worst case.

Data uncertainty in linear programming has been addressed by Ben-Tal and Nemirovski [13]. They define a 'robust counterpart' to an LP problem, and seek the solution with the best objective that satisfies all realisations of the constraints. There are close parallels between the ideas of the robust counterpart and of the certainty closure of a linear system. The uncertainty in the constraints must be independent.

The most general method for 'interval linear programming' is that of Chinneck and Ramadan [21]. Applying a transformation, they find the best and worst optimum for a linear system in which intervals may appear as coefficients in the constraints or the objective. Where equality constraints occur, they must resort to an exponential enumeration of cases.

Various authors have proposed analytic solutions to restricted cases of the ILS problem (see [16] among many others). The iterative methods that usually result require the matrix $\boldsymbol{A}$ to be square and have certain properties, and the solution set to be bounded.

A different approach, first suggested by Oettli [18] in an early paper, is to find some characterisation of the solution set as a polytope, then apply the simplex algorithm $2n$ times to find the extreme points with respect to each axis normal. This is the method we use in Sect. 3; later authors have suggested extensions.

We should also note the contrast with methods developed for interval constraint logic programming (see, e.g., [22]). Here, domains of the variables are treated as (usually real) intervals, and interval-based local consistency operators applied. However, the tight solution set is not guaranteed and, further, interval narrowing methods perform best in answering global questions arising from non-linear constraint systems [22, 23].

5 Perspectives and Future Directions

Work on data uncertainty in constraint programming is fairly new, despite the reality of uncertainty in many real world situations. Existing CP models do not address this aspect of the problem when a probabilistic framework is not applicable. The aim of the certainty closure approach is to provide robust solutions to uncertain data in ill-defined LSCO problems, and our first results are promising in terms of robustness and efficiency.

The case study points us to three issues in integration with the paradigm. First, the importance of incremental execution. Subsequent to the initial system, we may receive updated or new information regarding the uncertain data, but it is inefficient to recompute the entire inference on each occasion. For the case of ILS, we can think in terms of the addition of a new halfspace equation, or correspondingly a potential 'slicing-off' of some part of the convex hull.

One of the strengths of CP is the ability to handle many types of constraints (non-linear, scheduling, disjunctive, and so on), and problems involving heterogeneous constraints. The second area to explore in future work is to consider a wider class of constraints in uncertain LSCO problems, and consider how the certainty closure should be formulated.

The third area is optimisation. If the decision criterion is maximisation of a function $f : \mathcal{V} \to \mathbb{R}$, say, then solving the certainty closure as a *constraint optimisation problem* (COP) gives a hard upper bound. The problem is more difficult if f contains uncertainty; here the definition of ρ will need to be extended.

There has been little previous work on uncertain COPs. Sengupta et al. [12] model vagueness and uncertainty in linear inequalities by using intervals, and present an interpretation of the constraints based on the preference of the decision maker. Chinneck and Ramadan [21] find the best and worst optimum and the data values which lead to these. The best optimum supposes the most favourable values of the data; the worst, like the certainty closure, the least favourable. Again, rationality of the decision maker is assumed.

Acknowledgement. The authors thank Warwick Harvey for constructive suggestions and much helpful discussion. This work was partially supported by the EPSRC under grant GR/N64373/01.

References

[1] Wallace, R.J., Freuder, E.C.: Representing and coping with recurrent change in dynamic constraint satisfaction problems. In: CP'99 Workshop on Modelling and Solving Soft Constraints. (1999)

[2] Fowler, D.W., Brown, K.N.: Branching constraint satisfaction problems for solutions robust under likely changes. In: Proc. of CP2000, Singapore (2000) 500–504
[3] Schiex, T., Fargier, H., Verfaillie, G.: Valued constraint satisfaction problems: Hard and easy problems. In: Proc. of IJCAI'95. (1995) 631–637
[4] Dubois, D., Fargier, H., Prade, H.: Possibility theory in constraint satisfaction problems: Handling priority, preference and uncertainty. Applied Intelligence **6** (1996) 287–309
[5] Taha, H.: Operations Research: An Introduction. 5th edn. Prentice Hall (1997)
[6] Fargier, H., Lang, J.: Uncertainty in constraint satisfaction problems: A probabilistic approach. In Clarke, M., ed.: Symbolic and Qualitative Approaches to Reasoning and Uncertainty. Volume 747 of LNCS., Springer–Verlag (1993) 97–104
[7] Shazeer, N.M., Littman, M.L., Keim, G.A.: Constraint satisfaction with probabilistic preferences on variable values. In: Proc. of the Sixteenth National Conf. on Artificial Intelligence. (1999)
[8] Walsh, T.: Stochastic constraint programming. In: Proc. of AAAI'01 Fall Symposium on Using Uncertainty within Computation. (2001)
[9] Gervet, C., Caseau, Y., Montaut, D.: On refining ill-defined constraint problems: A case study in iterative prototyping. In: Proc. of PACLP'99, London (1999) 255–275
[10] Gervet, C., Rodošek, R.: RiskWise-2 problem definition. IC–Parc Internal Report (2000)
[11] Hoffman, K.L.: Combinatorial optimization: Current successes and directions for the future. J. Computational and Applied Mathematics **124** (2000) 341–360
[12] Sengupta, A., Pal, T.K., Chakraborty, D.: Interpretation of inequality constraints involving interval coefficients and a solution to interval linear programming. Fuzzy Sets and Systems **119** (2001) 129–138
[13] Ben-Tal, A., Nemirovski, A.: Robust solutions of uncertain linear programs. Operations Research Letters **25** (1999) 1–13
[14] Marriott, K., Stuckey, P.J.: Programming with Constraints: An Introduction. MIT Press, Cambridge, MA (1998)
[15] Ben-Tal, A., Nemirovski, A.: Robust solutions of linear programming problems contaminated with uncertain data. Mathmatical Programming Series A (2000)
[16] Neumaier, A.: Interval Methods for Systems of Equations. Cambridge University Press (1990)
[17] Kreinovich, V., Lakeyev, A., Rohn, J., Kahl, P.: Computational Complexity and Feasibility of Data Processing and Interval Computations. Kluwer (1998)
[18] Oettli, W.: On the solution set of a linear system with inaccurate coefficients. Journal of SIAM: Series B, Numerical Analysis **2** (1965) 115–118
[19] Aberth, O.: The solution of linear interval equations by a linear programming method. Linear Algebra and its Applications **259** (1997) 271–279
[20] IC–Parc: ECL^{i}PS^e User Manual Version 5.2. (2001)
[21] Chinneck, J.W., Ramadan, K.: Linear programming with interval coefficients. Journal of the Operational Research Society **51** (2000) 209–220
[22] Benhamou, F.: Interval constraints. In Floudas, C.A., Pardalos, P.M., eds.: Encyclopedia of Optimization. Kluwer Academic Publishers, New York (2001)
[23] Chiu, C.K., Lee, J.H.M.: Towards practical interval constraint solving in logic programming. In: Proc. of ILPS-94, Ithaca, NY (1994) 109–123