Constraint Reasoning with Uncertain Data using CDF-Intervals

Aya Saad, Carmen Gervet and Slim Abdennadher

German University in Cairo
New Cairo City, Egypt
{aya.saad, carmen.gervet, slim.abdennadher}@guc.edu.eg

Abstract. Interval coefficients have been introduced in OR and CP to specify uncertain data in order to provide reliable solutions to convex models. The output is generally a solution set, guaranteed to contain all solutions possible under any realization of the data. This set can be too large to be meaningful. Furthermore, each solution has equal uncertainty weight, thus does not reflect any possible degree of knowledge about the data. To overcome these problems we propose to extend the notion of interval coefficient by introducing a second dimension to each interval bound. Each bound is now specified by its data value and its degree of knowledge. This is formalized using the cumulative distribution function of the data set. We define the formal framework of constraint reasoning over this cdf-intervals. The main contribution of this paper concerns the formal definition of a new interval arithmetic and its implementation. Promising results on problem instances demonstrate the approach.

1 Introduction

Interval coefficients have been introduced in Operations Research and Constraint Programming to specify uncertain data in order to provide reliable solutions to convex models. They are at the heart of paradigms such as robust optimization [3, 12] in Operations Research as well as mixed CSP [8], reliable constraint reasoning [15, 16], and quantified CSP [17] in Constraint Programming. These paradigms specify erroneous and incomplete data using uncertainty sets that denote a deterministic and bounded formulation of an ill-defined data. To remain computationally tractable, the uncertainty sets are approximated by convex structures such as intervals (extreme values within the uncertainty set) and interval reasoning can be applied ensuring effective computations.

The concept of convex modeling was coined to formalize the idea of enclosing uncertainty sets and yield reliable solutions; i.e guaranteed to contain any solution produced by any possible realization of the data [2, 5, 16]. As a result, the outcome of such systems is a solution set that can be refined when more knowledge is acquired about the data, and does not exclude any potential solution. The benefits of these approaches are that they deal with real data measurements, produce robust/reliable solutions, and do so in a computationally tractable manner. However, the solution set can sometimes be too large to be meaningful since it encloses all solutions that can be constructed using the data intervals. Furthermore each solution derived has equal uncertainty weight, thus does not reflect any possible degree of knowledge about the data. For instance,

consider a collected set of data measurements in traffic flow analysis [9], or in image recognition [7], the data is generally ill-defined but some data values can occur more than others or have a darker shade of grey (hence greater degree of knowledge or certainty). This quantitative information is available during data collection, but lost during the reasoning because not accounted for in the representation of the uncertain data. As a consequence it is not available in the solution set produced. Mainly reliable models offer reliability and robustness, tractable models, but do not account for quantitative information about the data.

This paper addresses this problem. Basically we extend the interval data models with a second dimension: a quantitative dimension added to the measured input data. The main idea introduced in this paper is to show that we can preserve the tractability of convex modeling while enriching the uncertain data sets with a representation of the degree of knowledge available. Our methodology consists of building data intervals employing two dimensional points as extreme values. We assume that with each uncertain data value comes its frequency of occurrence or density function. We then compute the cumulative distribution function (cdf) over this function. The cdf is an aggregated quantitative measurement indicating for a given uncertain value, the probability that the actual data value lies before it. It has been used in different models under uncertainty to analyze the distribution of data sets (e.g. [14] and [10]). It enjoys three main properties that fit an interval representation of data uncertainty: i) the cdf is a monotone, non decreasing function like arithmetic ordering, suitable for interval computations and pruning, ii) it *directly* represents the aggregated probability that a quantity lies within bounds, thus showing the confidence interval of this uncertain data, iii) it brings flexibility to the problem modeling assumptions (e.g. by choosing the data value bounds based on the cdf values, or its sought confidence interval). We introduce the concept of cdf-intervals to represent such convex sets, following the concept of interval coefficients. This requires the decision variables to range over cdf-intervals as well. Basically, in our framework, the elements of a variable's domain are points in a 2D-space, the first dimension for its data value, the second for its aggregated cdf value. It is defined as a cdf-interval specified by its lower and upper bounds. A new domain ordering is defined within the 2D space. This raises the question of performing arithmetic computations over such variables to infer bound consistency. We define the constraint domain over which the calculus in this new domain structure can be performed, including the inference rules.

This paper contains the following contributions: (1) a new representation of uncertain data, (2) a formal framework for solving systems of arithmetic constraints over cdf-intervals, (3) a practical framework including the inference rules within the usual fixed point semantics, (4) an application to interval linear systems. The paper is structured along the contributions listed hereabove.

2 Basic Concepts

This section recalls basic concepts we use to characterize the degree of knowledge, and introduces our notations. These definitions can be found in [10].

2.1 Cumulative Distribution Function

Throughout this paper we assume independence of data.

Definition 1 (density function). Given a data set $\{N_1, \dots, N_n\}$, with available information regarding the occurrence of each data item. The density function $f(x)$ of a data item x , is defined over the population size, m , by:

$$f(x) = \frac{\text{number of occurrences of value } N_x}{m} \quad (1)$$

The cumulative distribution function cdf of a given value, defines its accumulated density so far. It does not assume any specific distribution function, but follows that of the density function.

Definition 2 (cumulative distribution function). Given an item value x , with density function $f(x)$, and an unknown variable (commonly referred to as the real-valued random variable) X , the cdf of x is the function:

$$F_X(x) = P(X \leq x) = \sum_{X \leq x} f(x) \quad (2)$$

The cdf ranges over the interval $[0,1]$.

Property 1. Every cdf is monotone, and right continuous.

In other words, the cdf value of a point is the density value of that point in addition to the sum of frequencies of all preceding points, with smaller item value. As illustrated in fig. 1 the cdf associated with a density function is always increasing until it reaches the point of stability '1' where the curve remains constant. All the data population resides between two points having cdf values '0' and '1' with an average step height equivalent to (m/n) , where n is the number of distinct values in the data set and m is the size of its population. The cdf expresses the distribution of the data in an aggregated manner.

2.2 Joint Cumulative Distribution Function

Operations can be performed on $cdfs$ but carry a different interpretation than operations over standard arithmetic calculus since they relate to probabilities. The joint operation is essential to our solver and is recalled below. The *joint cdf* is the cdf that results from superimposing two variables with a relation, each exerting a cdf on its own.

Definition 3 (Joint CDF). Given two random variables X and Y ¹

$$F_{XY}(x, y) = P(X \leq x, Y \leq y) \quad (3)$$

For independent variables $P(X \leq x, Y \leq y) = F_X(x) \times F_Y(y)$.

Definition 4 (Joint CDF over bounded intervals). When X and Y are bounded intervals; i.e. $X \in [a, b]$ and $Y \in [c, d]$, the joint CDF is defined by:

$$P(a \leq X \leq b, c \leq Y \leq d) = F_{XY}(b, d) - F_{XY}(a, d) - F_{XY}(b, c) + F_{XY}(a, c) \quad (4)$$

The joint cdf F_{XY} of two variables X and Y is used extensively in the computation of the cdf resulting from any operation between the variables.

¹ Recall that a random variable maps an event to values, e.g. event that $X \leq x$.

Fig. 1. constructing the data interval bounds

2.3 Notations

We assume that a data takes its value in the set of real numbers $\mathbb{R}$, denoted by a, b, c . A cdf value $F_X(x)$ is associated to the uncertainty curve of a given point p . For simplicity, $F_X(x)$ is noted F_x^p , i.e. the cdf value of an uncertain data p at value x . We have $p_x \in \mathbb{R} \times [0, 1]$ with coordinates (x, F_x^p) . Data points are denoted by p, q, r possibly subscripted by a data value. Variables are denoted by X, Y, Z and take their value in $\mathcal{U} = \mathbb{R} \times [0, 1]$. Intervals of elements from $\mathcal{U}$ are denoted I, J, K . We denote F^I the approximated linear curve relative to the cdf curve inbetween the bounds of I , and F_a^I the cdf value of the data value a plotted on the 2D-interval I .

3 Uncertain Data Representation

Given a measured (also possibly randomly generated) data set denoting the population of an uncertain data, we construct our cdf-intervals as detailed in algorithm 1.

```

procedure ConstructIntervalBounds( $m, Arr[n], Freq[n]$ )
1:  $cdf[1] \leftarrow Freq[1]/m$ 
2: for  $i = 2$  to  $n$  do
3:    $cdf[i] \leftarrow (Freq[i]/m) + cdf[i - 1]$ 
4:  $i = 1$ ,
5: while  $(Freq[i] \leq m/n)$  do {  $m/n$  is the average step value }
6:    $i \leftarrow i + 1$ 
7:  $lowerbound \leftarrow (Arr[i], cdf[i])$ 
8:  $upperbound \leftarrow (cdf^{-1}[0.98], 0.98)$ 

```

Algorithm 1: data interval bounds construction

The algorithm runs in $O(n)$ where n is the number of distinct values in the data set. It receives three parameters: the size of the data population, m , a sorted list (ascending

order) of the distinct measured data, and a list of their corresponding frequencies. Both lists are of the same size n . The algorithm first computes the *cdf* in a cumulative manner. The turning points are then extracted by recording the data value having a density less than the average step value (m/n) and the value with *cdf* equal to 98%.

Fig.1 illustrates an example of an interval data construction. For a data set size $n = 11$, and a population size $m = 30$, $Arr[n] = [12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42]$, and its corresponding frequencies $Freq[n] = [1, 3, 6, 5, 3, 4, 2, 2, 1, 1]$, the computed *cdf*-interval has the following bounds $[(15, 0.13), (39.6, 0.98)]$.

3.1 Interpretation of the confidence interval $[p_a, p_b]$

Consider the practical meaning of the interval $[p_a, p_b]$ that we have sought to obtain. This interval is built according to two main sources of information: 1) the monotony and non-decreasing properties of the *cdf* curve to account for degree of knowledge, 2) the extreme turning points over such a curve. Recall that the *cdf* curve indicates the aggregated distribution function of a data set. Plotting a point on this curve tells us what are the chances that the actual data value lies on or before this point. The extreme turning points we have considered are such that the lower bound indicates when the slope (thus frequency of occurrence over the population) increases more than the average; and the upper bound that of the *cdf* reaching a plateau. The measure of this upper bound has been associated with the *cdf* value of 98%. This corresponds to the distance of $avr + 3\sigma$ when the distribution follows a normal distribution. Such interpretation is a conservative view that can be revised by the decision maker.

It is also important to note the effectiveness of using the *cdf* as an indicator of degree of knowledge. Given a measurement of the data p such that (x, F_x^p) is any point, we have the following due to F^p monotone non-decreasing property:

$$a \leq x \leq b, \quad F_a^p \leq F_x^p \leq F_b^p$$

This implies that we can order (partially) points in this 2D-space $\mathcal{U} = \mathbb{R} \times [0, 1]$. Thus we can construct an algebra over variables taking their value in this space. In particular, we can approximate the *cdf* curve associated with a data population by the linear (increasing or constant) slope between the two turning points.

3.2 Linear Approximation within $\mathbf{I} = [p_a, p_b]$

Definition 5. For a given interval $\mathbf{I} = [p_a, p_b]$, F_x^I is the projected approximated *cdf* value of p_x onto F^I (the *cdf* associated to the interval), we will denote $p_x \in \mathbf{I}$ as $p_x = (x, F_x^I)$ for any point lying within the $\mathbf{I}$ interval bounds such that:

$$a < x < b, \quad F_x^I = \frac{F_b^I - F_a^I}{b - a} \cdot (x - a) + F_a^I \quad (5)$$

Property 2. $F_a^p = F_a^I$ and $F_b^q = F_b^I$

Example 1. Fig. 2 illustrates the computation of F_x^I . We have $\mathbf{I} = [(18.6, 0.44), (27.8, 0.78)]$. Given a data value $x = 24$ we compute its *cdf* $F_x^I = 0.64$, and obtain the point $p_x = (24, 0.64)$.

Fig. 2. F_x^I linear projection

4 cdf-Intervals

Our approach follows real interval arithmetic. It adds a second dimension to each uncertain value, requiring us to define a new ordering among points in a two dimensional space, together with new inference rules.

Consider a data population with its *cdf* curve. Create the set of points such that each points p_x is specified by $(x, F_x) \in \mathbb{R} \times [0, 1]$. The set $\mathcal{U} = \mathbb{R} \times [0, 1]$ is a set of tuples partially ordered, and constitutes a poset with a unique greatest lower bound and least upper bound. Similarly to reliable computing methods the constraint system will produce a solution set as opposed to a solution point. The variables thus denote intervals within the cdf-interval structure and constraint processing needs to be extended to perform arithmetic operations within this algebraic structure.

4.1 cdf-Interval ordering $\leq$

Definition 6 (Ordering over $\mathcal{U}$, $\leq$). Let $p_x = (x, F_x^p)$, $q_y = (y, F_y^q) \in \mathcal{U}$, the ordering $\leq$ is a partial order defined by:

$$p_x \leq q_y \Leftrightarrow x \leq y \text{ and } F_x^p \leq F_y^q$$

Example 2. Consider the three points $p_x = (1, 0.3)$, $q_y = (2, 0.5)$ and $r_z = (2, 0.1)$. We have $p_x \leq q_y$ as well as $r_z \leq q_y$, but p_x and r_z are not comparable.

A *cdf*-interval delimited by two points p_x and q_y is specified by the syntax $[p_x, q_y]$ such that $p_x \leq q_y$. One important task in interval reasoning is the computation of a new interval from arbitrary points or previous intervals, such that it describes the smallest interval containing a collection of elements. This is based on the meet and join operators.

Definition 7 (meet and join). Given the arithmetic ordering and meet and join operations over the reals $(\mathbb{R}, \leq, \min, \max)$ and the ordering of *cdf* values within $([0, 1], \leq$

, $\min, \max$), the meet glb and join lub operators of two points p_x and q_y in $\mathcal{U}$ are defined by:

$$\begin{aligned} \text{glb}(p_x, q_y) &= (\min(x, y), \min(F_{lb}^p, F_{lb}^q)) \\ \text{lub}(p_x, q_y) &= (\max(x, y), \max(F_{ub}^p, F_{ub}^q)) \end{aligned} \quad (6)$$

where $lb = \min(x, y)$ and $ub = \max(x, y)$

The following property establishes the link between the partial ordering $\leq$ and the pair (glb, lub) as actual meet and join.

Property 3 (Consistency property).

$$\begin{aligned} p_x \leq q_y &\Leftrightarrow p_x = \text{glb}(p_x, q_y) \\ p_x \leq q_y &\Leftrightarrow q_y = \text{lub}(p_x, q_y) \end{aligned} \quad (7)$$

Fig. 3. glb and lub computation

Fig. 3 illustrates an example which computes glb and lub of two points $p_x = (1, 0.3)$ and $q_y = (2, 0.1)$. $\text{glb}(p_x, q_y) = (1, 0.05)$ and $\text{lub}(p_x, q_y) = (2, 0.6)$.

4.2 cdf Domains

The key element to a cdf-Interval domain is the approximated cdf curve it lies on. However, to remain computationally tractable we do not maintain a full domain representation of the points defining the curve. Instead, we approximate the curve by a linear curve whose extreme points are the bounds of the interval. Elements of the interval domain lie on the linear curve. This leads to the following concept of cdf-Interval domain.

Definition 8 (cdf-Domain). A cdf-Domain is a pair $[p_a, p_b]$ satisfying $p_a \leq p_b$ and denoting the set:

$$\{p_x = (x, F_x^p) \mid p_a \leq p_x \leq p_b, \text{ and } F_x^p = \frac{F_b^p - F_a^p}{b - a} \cdot (x - a) + F_a^p\} \quad (8)$$

Example 3. Consider a cdf-variable X with domain $I = [(18.6, 0.44), (27.8, 0.78)]$ as illustrated in fig. 2. X can take any point value (x, F_x^I) such that $18.6 \leq x \leq 27.8$ and $F_x^I = \frac{0.78-0.44}{27.8-18.6} \cdot (x - 18.6) + 0.44$.

5 Core Operations on cdf-Intervals

Clearly this work follows the real interval arithmetic introduced in [4]. In particular when the degree of knowledge provides equal weight to each data value the computed intervals are identical. Thus the novelty here lies in the calculus presented with respect to the second dimension, i.e. the degree of knowledge based on the *cdf* values. We consider the standard arithmetic operations interpreted over the set of cdf-intervals. For $\odot \in \{+_{\mathcal{U}}, -_{\mathcal{U}}, \times_{\mathcal{U}}, \div_{\mathcal{U}}\}$ a binary interval arithmetic over two dimensions we seek:

$$[p_a, p_b] \odot [q_c, q_d] = \{p_X \odot q_Y \mid p_X \in [p_a, p_b], q_Y \in [q_c, q_d]\} \quad (9)$$

Fig. 4. *cdf* distribution resulting from superimposing two intervals for x and y with a relation: (a) addition (b) multiplication (c) subtraction. $x \in [a, b]$ with a cdf F_x^I and $y \in [c, d]$ with a cdf F_y^I

Any two intervals, each shaping a different distribution *cdf*, can be involved in a relation given by a function. This relation in turn shapes a *cdf* that is based on a double integration of their joint *cdf* over the set of values per interval under the curve of the function [10]. From this generic methodology we derive *cdf* lower and upper bound equations for each binary arithmetic operation. Derived equations are shown by the dark shaded area under the curve of the relation depicted in fig. 4. Proofs are omitted for space reason.

5.1 Addition '+_U'

Consider two arbitrary intervals $\mathbf{I} = [p_a, p_b]$ and $\mathbf{J} = [q_c, q_d]$, their arithmetic addition is a result of adding every two points p_x and q_y from both intervals. The resulting *cdf*, F^{I+J} is obtained by superimposing both *cdf* distributions F_x^I and F_y^J as depicted by the dark shaded regions in fig.4(a): it represents the area under the line that describes the relation

between x and y , first arguments of points p_x and q_y respectively. The computation of F^{I+J} is based on the joint *cdf* previously discussed in section 2.2; we assume in our calculations that probabilities outside the interval are negligible ' $= 0$ ' since we are working on the majority of the population inside an interval. The resulting interval is specified by

$$[(lb_+, F_{lb_+}^{I+J}), (ub_+, F_{ub_+}^{I+J})]$$

such that the real interval arithmetic addition is applied to compute the 1st-dimension lower and upper bounds respectively denoted lb_+ and ub_+ :

$$lb_+ = \min(a + c, a + d, b + c, c + d) \text{ and } ub_+ = \max(a + c, a + d, b + c, c + d)$$

$$F_{lb_+}^{I+J} = 0.5[F_{lb_+}^J F_c^J + F_a^I F_{lb_+}^I]$$

$$F_{ub_+}^{I+J} = 0.5[F_b^I + F_d^J]$$

with the *cdf* value of lb_+ on **J** and **I** respectively defined as:

$$F_{lb_+}^J = \begin{cases} F_c^J + a \left[\frac{F_d^J - F_c^J}{d - c} \right] & \text{if } d < a + c \\ 1 & \text{otherwise} \end{cases}$$

$$F_{lb_+}^I = \begin{cases} F_a^I + c \left[\frac{F_b^I - F_a^I}{b - a} \right] & \text{if } b < a + c \\ 1 & \text{otherwise} \end{cases} \quad (10)$$

Example 4. Given two data populations with associated *cdf* curves, approximated by the *cdf*-intervals **I** = [(1, 0.3), (7, 0.65)] and **J** = [(2, 0.46), (9, 0.6)]; the addition of two uncertain data from **I** and **J** respectively is specified by the *cdf*-interval $[r_{a+c}, r_{b+d}] = (3, 0.38), (16, 0.625)]$ as shown in fig.5. Note that in the absence of second dimension we obtain a regular interval arithmetic addition.

Fig. 5. Example: *cdf*-Interval addition

5.2 Multiplication '×_u'

Multiplying each pair of points lying in two intervals results in computing the cdf $F^{I \times J}$ distribution illustrated by the shaded dark region under the curve in fig.4(b).

$\mathbf{I} = [p_a, p_b] \times \mathbf{J} = [q_c, q_d]$ produces the cdf-interval

$$[(lb_{\times}, F_{lb_{\times}}^{I \times J}), (ub_{\times}, F_{ub_{\times}}^{I \times J})]$$

such that the first dimensions, follows the conventional real interval arithmetic multiplication. The lower and upper bounds are defined by $lb_{\times}$ and $ub_{\times}$. Recall that data values can be negative:

$$lb_{\times} = \min(a \times c, a \times d, b \times c, b \times d) \text{ and } ub_{\times} = \max(a \times c, a \times d, b \times c, b \times d)$$

The 2^{nd} -dimension cdf for the resulting interval bounds, are computed as follows:

$$\begin{aligned} F_{lb_{\times}}^{I \times J} &= 0.5(F_b^I F_c^J + F_a^I F_d^J) \\ F_{ub_{\times}}^{I \times J} &= \max(F_d^J, F_b^I) \end{aligned} \quad (11)$$

Example 5. Given two variables ranging over the respective cdf-intervals: $\mathbf{I} = [(1, 0.3), (4, 0.65)]$ and $\mathbf{J} = [(1.5, 0.46), (5, 0.6)]$ as illustrated in fig.6. The result of the multiplication is a cdf-interval specified by $[r_{a \times c}, r_{b \times d}] = (1.5, 0.24), (20, 0.65)]$.

Fig. 6. Example: cdf-Interval multiplication

5.3 Subtraction '-_u'

Given two cdf-intervals $\mathbf{I} = [p_a, p_b]$ and $\mathbf{J} = [q_c, q_d]$, the subtraction derives the cdf-interval

$$[(lb_{-}, F_{lb_{-}}^{J-I}), (ub_{-}, F_{ub_{-}}^{J-I})]$$

such that

$$lb_{-} = \min(c - a, d - a, c - b, d - b) \text{ and } ub_{-} = \max(c - a, d - a, c - b, d - b)$$

and F^{J-I} lower and upper bounds are:

$$\begin{aligned} F_{lb_-}^{J-I} &= 0.5(F_b^I F_c^J + F_a^I F_d^J) \\ F_{ub_-}^{J-I} &= 0.5F_b^I [1 + F_{ub_-}^J] \end{aligned} \quad (12)$$

5.4 Division ' $\div_{\mathcal{U}}$ '

Given two cdf-intervals $\mathbf{I} = [p_a, p_b]$ and $\mathbf{J} = [q_c, q_d]$, the division operation derives the resulting cdf-interval

$$[(lb_{\div}, F_{lb_{\div}}^{J \div I}), (ub_{\div}, F_{ub_{\div}}^{J \div I})]$$

such that:

$$lb_{\div} = \min(c \div a, d \div a, c \div b, d \div b) \text{ and } ub_{\div} = \max(c \div a, d \div a, c \div b, d \div b)$$

$F^{J \div I}$ lower and upper bounds are:

$$\begin{aligned} F_{lb_{\div}}^{J \div I} &= 0.5(F_b^I F_c^J) \\ F_{ub_{\div}}^{J \div I} &= F_d^J [F_b^I - 0.5F_a^I] \end{aligned} \quad (13)$$

6 Implementation

The constraint system behaves like a solver over real intervals, based on the relational arithmetic of real intervals where arithmetic expressions as interpreted as relations [6]. The relations are handled using the following transformation rules that extend the ones over real intervals with inferences over the cdf values. The handling of these rules is done by a relaxation algorithm which resembles the arc consistency algorithm AC-3 [13]. The solver converges to a fixed point or infers failure. we ensure termination of the generic constraint propagation algorithm because the cdf-domain ordering is reflexive, antisymmetric and transitive. Hereafter we present the main transformation rules for the basic arithmetic operations. For space reasons, when a domain remains unchanged we will use the following notation: $\mathbf{I} = [p_a, p_b]$, $\mathbf{J} = [q_c, q_d]$ and $\mathbf{K} = [r_e, r_f]$. The cdf-variables are denoted by X, Y and Z . Failure is detected if some domain bounds do not preserve the ordering $\leq$.

Ordering constraint $X \leq Y$

$$\frac{p_b' = glb(p_b, q_d), q_c' = lub(p_a, q_c)}{\{X \in \mathbf{I}, Y \in \mathbf{J}, X \leq Y\} \mapsto \{X \in [p_a, p_b'], Y \in [q_c', q_d], X \leq Y\}}$$

Equality constraint $X = Y$

$$\frac{p_b' = glb(p_b, q_d), p_a' = lub(p_a, q_c)}{\{X \in \mathbf{I}, Y \in \mathbf{J}, X = Y\} \mapsto \{X \in [p_a', p_b'], Y \in [p_a', p_b'], X = Y\}}$$

Ternary addition constraints $X +_{\mathcal{U}} Y = Z$

$$\frac{r_f' = (ub_+, F_{ub_+}^{I+J}), r_e' = (lb_+, F_{lb_+}^{I+J})}{\{X \in \mathbf{I}, Y \in \mathbf{J}, Z \in \mathbf{K}, Z = X +_{\mathcal{U}} Y\} \mapsto \{X \in \mathbf{I}, Y \in \mathbf{J}, Z \in [r_e', r_f'], Z = X +_{\mathcal{U}} Y\}}$$

The projection onto Y 's domain is symmetrical.

$$\frac{p_b' = (ub_-, F_{ub_-}^{K-J}), p_a' = (lb_-, F_{lb_-}^{K-J})}{\{X \in \mathbf{I}, Y \in \mathbf{J}, Z \in \mathbf{K}, X = Z -_{\cup} Y\} \mapsto \{X \in [p_a', p_b'], Y \in \mathbf{J}, Z \in \mathbf{K}, X = Z -_{\cup} Y\}}$$

Ternary multiplication constraint $X \times_{\cup} Y = Z$

$$\frac{r_f' = (ub_{\times}, F_{ub_{\times}}^{I \times J}), r_e' = (lb_{\times}, F_{lb_{\times}}^{I \times J})}{\{X \in \mathbf{I}, Y \in \mathbf{J}, Z \in \mathbf{K}, Z = X \times_{\cup} Y\} \mapsto \{X \in \mathbf{I}, Y \in \mathbf{J}, Z \in [r_e', r_f'], Z = X \times_{\cup} Y\}}$$

The projection onto Y 's domain is symmetrical.

$$\frac{p_b' = (ub_{\div}, F_{ub_{\div}}^{K \div J}), p_a' = (lb_{\div}, F_{lb_{\div}}^{K \div J})}{\{X \in \mathbf{I}, Y \in \mathbf{J}, Z \in \mathbf{K}, X = \mathbf{K} \div_{\cup} \mathbf{J}\} \mapsto \{X \in [p_a', p_b'], Y \in \mathbf{J}, Z \in \mathbf{K}, X = \mathbf{K} \div_{\cup} \mathbf{J}\}}$$

6.1 Empirical evaluation

A prototype implementation of the constraint system was done as a module on top of the CHR library in Sicstus Prolog. To evaluate the added value of the new constraint domain, we consider an example provided in [16], and attached a degree of knowledge: the cdf values to the interval bounds. Example 6 aims at solving a system of linear equations which has cdf-interval coefficients and unknown variables having no certainty degree defined, i.e. the lower bound points are '(0, 0)' whereas upper-bound of the variable interval is '(\infty, 1)' with a *cdf* value '1'. Shown below are pruned domains of the variables at fixed point using our inferences. The cdf-intervals attached to the data (here coefficients) were propagated onto the cdf-variables, X_1 and X_2 . We can see that inferences on the 1st-dimension yield similar pruning on the resulting variable domains and the additional information coming from the *cdf* component demonstrate the information gained on the density of occurrence for the resulting points within the cdf-domains.

Example 6. Consider the system of linear equations ($\mathbf{A}, \mathbf{R}, \mathbf{b}$) shown below:

$$\mathbf{A} = \begin{bmatrix} [(-2, 0.32), (2, 0.83)] & [(1, 0.41), (2, 0.95)] \\ [(-2, 0.35), (-1, 0.87)] & [(-1.001, 0.31), (-1, 0.75)] \\ [(5.999, 0.28), (6, 0.86)] & [(1.5, 0.34), (3, 0.96)] \end{bmatrix},$$

$$\mathbf{R} = \begin{bmatrix} \leq \\ = \\ = \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} [(3, 0.4), (4, 0.88)] \\ [(-5, 0.3), (5, 0.78)] \\ [(4, 0.29), (15, 0.85)] \end{bmatrix}$$

The output domains for the variables X_1 and X_2 after applying the propagation cdf-interval techniques are $[(0.00, 0.05), (2.50, 0.69)]$ and $[(0.00, 0.06), (5.00, 0.59)]$ respectively.

Remark If we remove any density value for the uncertain coefficients by assigning values '0' and '1' to the lower and upper bounds of the *cdf* dimension, this amounts to considering that the uncertain data is equally distributed along the given interval. The output in this case is $X_1 \in [(0.00, 0.00), (2.50, 1.00)]$ and $X_2 \in [(0.00, 0.00), (5.00, 1.00)]$. In this sense the cdf-interval constraint model generalizes the real interval arithmetic constraint reasoning.

6.2 cdf-Interval Linear Programming

In this section we also show how the cdf-interval constraint model does also generalize Interval Linear Systems [11]. Thus linear systems with cdf-interval coefficients and variables can also be solved by a polynomial transformation into a linear model that can then be sent to the Simplex method. The approach is very similar to the one used for ILS with positive coefficients (also called POLI systems) [1].

Definition 9. Let $\mathcal{V}$ be a set of n cdf-variables, and C a set of m linear constraints over $\mathcal{V}$ with cdf-interval coefficients. An cdf-interval Linear System UILS induced by C over $\mathcal{V}$ is a tuple $\langle \mathbf{A}, \mathfrak{R}, \mathbf{b} \rangle$, where $\mathbf{A} \in (\mathbb{R} \times [0..1])^{m \times n}$ is the matrix of the LHS intervals $[glb_A, lub_A]$, $\mathbf{b} \in (\mathbb{R} \times [0..1])^m$ is the vector of the RHS intervals $[glb_b, lub_b]$ and $\mathfrak{R}_i \in \{<, \leq, =, \geq, >\} \forall i = 1, \dots, m$ is a list of m relations.

Generated Linear Inequalities For space reasons we demonstrate the approach of transformation of a positive cdf-interval linear system with positive values into a linear system and omit the formal proof of equivalence of models.

Consider the following minimization problem over cdf-coefficients and variables:

$$\begin{aligned} \text{Minimize } Z &= \sum_{j=1}^n [p_{e_j}, p_{f_j}] p_{x_j} \\ \text{subject to } \sum_{j=1}^n [p_{a_{ij}}, p_{b_{ij}}] p_{x_j} &\geq [p_{c_i}, p_{d_i}] \quad \forall i = 1, 2, \dots, m \\ &\quad \forall j, \quad p_{x_j} \in \mathbb{R}^+ \times [0..1] \end{aligned} \quad (14)$$

The transformation of the above model yields $(m \times 2^{n+1})$ inequalities per dimension. The produced solution set is $S_{di} = \{S_{di}^k | k = 1, 2, \dots, 2^{n+1}\}$ where the upper-bound value range is $S_{di} = lub_{k=1}^{2^{n+1}} S_{di}^k$ and the lower-bound value range is $S_{di} = glb_{k=1}^{2^{n+1}} S_{di}^k$; d is the dimension order: 1 or 2.

The transformation is performed in two steps: first, each equality constraint is transformed into two cdf-interval inequalities; then the cdf-constraints are transformed into linear constraints. The output of the transformation applied to example 6 will be:

$$\mathbf{A} = \begin{bmatrix} (-2, 0.32) & (1, 0.41) \\ (-2, 0.35) & (-1.001, 0.31) \\ (1, 0.13) & (1, 0.25) \\ (5.999, 0.28) & (1.5, 0.34) \\ (-6, 0.14) & (-3, 0.04) \end{bmatrix}, \mathfrak{R} = \begin{bmatrix} \leq \\ \leq \\ \leq \\ \leq \\ \leq \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} (4, 0.88) \\ (5, 0.78) \\ (5, 0.7) \\ (15, 0.85) \\ (-4, 0.71) \end{bmatrix}$$

Accordingly, the 1st constraint row is the inequality constraint: $(-2, 0.32)p_{x_1} + (1, 0.41)p_{x_2} \leq (4, 0.88)$. This constraint in turn will be transformed into two components: $-2x_1 + x_2 \leq 4$ and $0.25[0.41F_{ub}^2 + 0.95F_{lb}^2 + 0.32F_{ub}^1 + 0.83F_{lb}^1] \leq 0.88$. The linear equations resulting from the transformation of the above output are presented here, where equations 15 and 16 are in the 1st and 2nd dimensions respectively:

$$\begin{aligned}
-2x_1 + 2x_2 &\leq 4 \\
-2x_1 - 1.001x_2 &\leq 5 \\
x_1 + x_2 &\leq 5 \\
5.999x_1 + 1.5x_2 &\leq 15 \\
-6x_1 - 3x_2 &\leq -4
\end{aligned} \tag{15}$$

$$\begin{aligned}
0.25[0.41F_{ub}^2 + 0.95F_{lb}^2 + 0.32F_{ub}^1 + 0.83F_{lb}^1] &\leq 0.88 \\
0.25[0.31F_{ub}^2 + 0.75F_{lb}^2 + 0.35F_{ub}^1 + 0.87F_{lb}^1] &\leq 0.78 \\
0.5[F_{ub}^1 + F_{ub}^2] &\leq 0.7 \\
0.25[0.34F_{ub}^2 + 0.96F_{lb}^2 + 0.28F_{ub}^1 + 0.86F_{lb}^1] &\leq 0.85 \\
0.5[F_{ub}^1 + F_{ub}^2] &\leq 0.71 \\
F_{lb}^1 &\leq F_{ub}^1 \\
F_{lb}^2 &\leq F_{ub}^2
\end{aligned} \tag{16}$$

for a point $p_{x_i}:x_i$ is the point value in the 1st-dimension and both F_{lb}^i and F_{ub}^i are the 2nd-dimension lower bound and upper bound values that the point can take.

An additional constraint has been added to the linear equations in the 2nd-dimension; this ensures that the lower bound value is less than its upper bound value. It is clear that resulting equations in both dimensions are linear and can be solved using linear programming techniques.

7 Conclusion

The framework of reliable computing offers robust and tractable approaches to reason with uncertain data by means of convex models of uncertainty sets (e.g using interval coefficients). It does not account however for any degree of knowledge about the data such as the density function. Thus all solutions in the solution set have equal uncertainty weight. This paper addressed this issue and showed how to embed a degree of knowledge in the form of the cumulative distribution function. The paper proposed the novel concept of cdf-interval, whereby an uncertainty set is specified by an interval of points with first coordinate the data uncertainty value, and second coordinate its cdf value. Since the uncertain data and consequently the decision variables are specified by their confidence interval, so is the solution set. We also presented the constraint system over this new domain by extending the real interval arithmetic to cdf-interval arithmetic using the monotone non-increasing property of the cdf function. Finally the application of cdf-intervals to extend the approach over Interval Linear Systems was showcased. We are currently applying this new approach to larger systems of constraints with application to vehicle routing, networking but also finance and image recognition, where the uncertain data is enriched with a degree of knowledge (i.e..the density function drawn from historic data trends or randomly generated).

References

1. O. Beaumont. Solving interval linear systems with linear programming techniques. *Linear Algebra and its Applications*, 281(1-3):293–309, 1998.
2. Y. Ben-Haim and I. Elishakoff. Discussion on: A non-probabilistic concept of reliability. *Structural Safety*, 17(3):195–199, 1995.
3. A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1):1–14, 1999.
4. F. Benhamou and R. de Vinci. Interval constraint logic programming. *Constraint programming: basics and trends: Châtillon Spring School, France*, 1994.
5. J. Chinneck and K. Ramadan. Linear programming with interval coefficients. *Journal of the Operational Research Society*, pages 209–220, 2000.
6. J. Cleary. Logical arithmetic. *Future Computing Systems*, 2(2):125–149, 1987.
7. A. Deruyver and Y. Hodé. Qualitative spatial relationships for image interpretation by using a conceptual graph. *Image and Vision Computing*, 27(7):876–886, 2009.
8. H. Fargier, J. Lang, and T. Schiex. Mixed constraint satisfaction: A framework for decision problems under incomplete knowledge. In *Proceedings of the National Conference on Artificial Intelligence*, pages 175–180. Citeseer, 1996.
9. M. Grossglauser and J. Rexford. Passive Traffic Measurement for Internet Protocol Operations. *The Internet as a Large-Scale Complex System*, page 91, 2005.
10. J. Gubner. *Probability and Random processes for electrical and computer Engineers*. Cambridge Univ Pr, 2006.
11. E. Hansen. Global optimization using interval analysis: the one-dimensional case. *Journal of Optimization Theory and Applications*, 29(3):331–344, 1979.
12. K. Hoffman. Combinatorial Optimization: Current successes and directions for the future. *Journal of computational and applied mathematics*, 124(1-2):341–360, 2000.
13. A. Mackworth. Consistency in networks of relations. *Artificial intelligence*, 8(1):99–118, 1977.
14. A. Tversky and D. Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of risk and uncertainty*, 5(4):297–323, 1992.
15. N. Yorke-Smith. *Reliable constraint reasoning with uncertain data*. PhD thesis, IC-Parc, Imperial College London, 2004.
16. N. Yorke-Smith and C. Gervet. Certainty closure: Reliable constraint reasoning with incomplete or erroneous data. *ACM Transactions on Computational Logic (TOCL)*, 10(1):3, 2009.
17. K. Zhou, J. C. Doyle, and K. Glover. *Robust and optimal control*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996.