Continuous First-Order Constraint Satisfaction with Equality and Disequality Constraints

Stefan Ratschan

Institut d'Informatica i Aplicacions, Universitat de Girona, Spain,
e-mail: stefan.ratschan@risc.uni-linz.ac.at

Abstract. In an earlier paper we have shown, how one can successfully use constraint satisfaction techniques for proving and solving formulae in the first-order predicate language over the reals (i.e., real first-order constraints). This approach was restricted to inputs that contain inequality symbols such as $\leq$, but no equality symbols ($=$) or disequality symbols ($\neq$). In this paper we lay the basis for extending this approach to inputs that contain (dis)equalities. This considerably widens the applicability of constraint programming in areas such as control engineering, and builds a strong bridge to areas such as quantifier elimination, automated theorem proving, and computer algebra.

1 Introduction

Let a real first-order constraint be a formula in the first-order predicate language with predicate symbols $=$, $\neq$, $\leq$, $<$, function symbols such as $+$, $\times$, sin, tan, and their usual interpretation over the real numbers. The problem of solving real first-order constraints is of fundamental importance—we have created a web-page [26] that lists more then 50 papers with applications. In an earlier paper [29] we have shown how one can extend constraint satisfaction techniques to solve real first-order constraints that do not contain equalities or disequalities. In this paper we set the basis for extending this approach to the case with equalities and disequalities.

The main difficulty for introducing equalities in this context is, that the solution set of equality constraints does not have volume, which can make it very difficult to find elements in such a solution set. For example, for proving a first-order constraint of the form $\exists x\ f(x) = 0$, one has to find an element of the solution set of the constraint $f(x) = 0$. However, in general, it may be the case that this solution set is non-empty, but no element is a natural number, rational number, floating point number, or even real algebraic number. So it is futile to try to prove the existential quantifier by finding the exact value of such an element. However, it is often quite easy to find a small interval that provably contains a solution. For example, the expression $x^2 - 2$ is negative for $x = 0$ and positive for $x = 4$. Hence, by elementary analysis (Boltzmann intermediate value theorem), the interval $[0, 4]$ contains a solution of $x^2 - 2 = 0$, and $\exists x\ x^2 - 2 = 0$ is true. Note that the approach fails if we use the larger interval $[-4, 4]$, where the function is positive also for $x = -4$.

By applying first-order constraint satisfaction [29] one can find intervals that enclose the solutions of the occurring equalities tightly. But, up to now it was unclear, how to extend the approach sketched above to general first-order constraint for which all the variables ranges over small closed intervals.

A naive approach for extending first-order constraint satisfaction to equalities would rewrite equalities of the form $f = 0$ to $f \geq 0 \ \wedge \ f \leq 0$. However, this usually creates first-order constraints that are *numerically ill-posed* [30]. Such constraints are exactly the ones, where a first-order constraint satisfaction based approach does not (and can not) succeed.

Another idea would be to rewrite constraints of the form $\exists x \ f(x) = 0$ to $\exists x \ f(x) \geq 0 \ \wedge \ \exists x \ f(x) \leq 0$. However, there seems to be no useful way of generalizing this approach.

One might also think that one can easily solve the problem by proving that independent of the values for the universally quantified variables we can apply an existence proof based on interval methods [25,16]. However, this would require a transformation of the constraint into prenex normal form (i.e., all quantifiers occur at the beginning), which destroys important structural information. Furthermore this approach cannot deal with alternations between universal and existential quantifiers. Moreover, it does not follow the recursive structure of first-order constraints and algorithms for solving them, making their cooperation impracticable.

Traditionally, symbolic quantifier elimination algorithms [31,6,11,32] have been used to solve real first-order constraints. Although these techniques have been successful for certain applications, they still suffer from problems such as unwieldy output expressions, and restriction to addition and multiplication, small number of variables and small polynomial degrees. By using interval/constraint satisfaction techniques one can at least relieve some of these problems [2,28,29,27].

Existence proofs for systems of equations based on interval arithmetic have been studied in detail by interval analysis [25,16]. The usual case studied has a 0-dimension solution set. Branch and bound algorithms are used to search the solution space, variants of Newton's methods are used to find approximate zeros, and then a box is constructed around the zero [21], within which one can prove the existence of a solution. Existence proofs are also used in the system Numerica [10,9], that uses constraint satisfaction techniques (box consistency [3]) to prune the search space.

The generalization to existence proofs for underdetermined/parametrized system of equations is an important sub-problem in constrained optimization [17,16,8]. The underdetermined case has also been studied on its own [23,7,33]. An important related question is interval-based sensitivity analysis, that is the computation of a provably correct bound on a certain solution of a parametrized system of equations under a given perturbation of its parameters [24,18].

The idea to apply interval techniques to real first-order constraints is according to Hong [12]. They also have been used for solving special cases occurring in engineering applications [13,20]. Constraint satisfaction techniques have been used for the special case with one universal quantifier [2], and for the general case [29].

The content of the paper is as follows: In Section 2 we introduce the necessary preliminaries; in Section 3 we describe the main idea in an informal style; in Section 4 we give the formal details; in Section 5 we show how to use the introduced techniques within continuous first-order constraint satisfaction [29]; in Section 6 we describe a provisional implementation, and in Section 7 we conclude the paper.

2 Preliminaries

Throughout the paper we use the term "constraint" as a short-cut for "real first-order constraint". We assume that all the occurring variables are elements of a totally ordered set $\mathcal{V}$. As a slight modification of the usual syntax we require that all quantified variables be bounded by closed intervals, which we denote as follows: $\exists x \in I\ \phi$, and $\forall x \in I\ \phi$. Here we call the intervals I *quantifier bounds*.

We use the terms free variable, sub-constraint, and atomic constraint as usual. Furthermore we let a *bounded constraint* be a pair (ϕ, B) where $B \subseteq \mathbb{R}^n$, where n is the number of free variables of ϕ.

Note that it is trivial to eliminate negations from constraints: Push them down to the atomic constraints by swapping universal and existential quantifiers and applying de Morgan's law. Then replace an atomic constraint of the form $\neg f = 0$ by $f \neq 0$, $\neg f \leq 0$ by $f > 0$ and so on. Thus we only work on constraints without negations in this paper. Whenever we speak of the negation of a constraint, we assume that this negation has immediately been eliminated according to the above procedure.

Given a first-order constraint ϕ in variables $v_1, \dots, v_n$, $\phi\frac{x_1}{v_1} \dots \frac{x_n}{v_n}$ denotes the truth-value of ϕ when the real numbers $x_1, \dots, x_n$ are assigned to the variables $v_1, \dots, v_n$, respectively.

For an interval I, we denote by $\underline{I}$ its left border, and by $\overline{I}$ its right border. An n-dimensional *box* is a Cartesian product of n closed intervals. Given a box $B = [\underline{x_1}, \overline{x_1}] \times \dots \times [\underline{x_{|\mathcal{V}|}}, \overline{x_{|\mathcal{V}|}}]$, and a variable v, which is i-th variable in $\mathcal{V}$, we denote by $B(i)$, and also by $B(v)$, the interval $[\underline{x_i}, \overline{x_i}]$. Moreover we denote by $B\frac{I}{v}$ the box $[\underline{x_1}, \overline{x_1}] \times \dots \times I \times \dots \times [\underline{x_n}, \overline{x_n}]$. For two sets B_1 and B_2 we denote by $B_1 \uplus B_2$ the smallest box containing both B_1 and B_2. For every set D we denote by $\widehat{D}$ the smallest closed set containing D.

We extend the notion of sub-constraint to bounded constraints by calling a bounded constraint (ϕ', B') is *sub-constraint* of a bounded constraint (ϕ, B) iff ϕ' a sub-constraint of ϕ, and for all variables v, $B'(v)$ is

- if v is quantified in ϕ along the path from the root to ϕ', then the corresponding quantifier bound, and
- $B(v)$, otherwise.

For example, the bounded constraint $(x^2 + y^2 = 1, [-2, 2] \times [-2, 2])$ is a sub-constraint of the bounded constraint $(\exists y\ x^2 + y^2 = 1, [-2, 2])$.

The main tool that we use in the proofs is the following theorem, which is a reformulation of Theorem 5.3.7 of A. Neumaier's book on Interval Methods [25]—a parametrized generalization of Miranda's Theorem [19, 22] that ensures the continuity of the parametrized solution space. Miranda's theorem is a generalization of the Boltzmann intermediate value theorem to systems of equations, and it is one of the main tools used in interval analysis [25, 16]. Readers who are not interested in proofs can savely skip this theorem.

Theorem 1. *Let $G : \mathbb{R}^p \times \mathbb{R}^n \to \mathbb{R}^n$ be a continuous function. Let $T \subseteq \mathbb{R}^p$ and $B \subseteq \mathbb{R}^n$ be boxes such that for all $i \in \{1, \dots, n\}$, for all $t \in T$, for all $(x_1, \dots, x_n) \in B$,*

- *$x_i = \underline{B(i)}$ implies $G_i(t, x_1, \dots, x_n) \leq 0$, and*
- *$x_i = \overline{B(i)}$ implies $G_i(t, x_1, \dots, x_n) \geq 0$.*

Then there is a continuous function $S : T \to B$ such that for all $t \in T$, $G(t, S(t)) = 0$.

Note that one can easily change the necessary signs for the first and second condition by simply multiplying G_i with -1. Note further that for applying the theorem in a concrete situation one might have to reorder the coordinates of G by a suitable bijection.

For illustrating the theorem, take the example of the function $g(t, x) \doteq t^2 x$, where $T = [-1, 1]$ and $B = [-1, 1]$. For all $t \in [-1, 1]$, $g(t, -1) = -t^2 \leq 0$, and $g(t, 1) = t^2 \geq 0$. Therefore there is a function $S : [-1, 1] \to [-1, 1]$ such that for all $t \in [-1, 1]$, $g(t, S(t)) = 0$. In our case this function is $S(t) = 0$.

3 Main Idea

In this section we demonstrate the main idea of how to (dis)prove first-order constraints with small quantifier bounds. For this we employ a semi-formal style. In the following section we will then describe the formal details.

We proceed recursively according to the structure of constraints. For example, for a constraint of the form $\exists x \in I_x\ \forall y \in I_y\ \exists z \in I_z\ x^2yz = 0\ \wedge\ xy^2z - 1 = 0$, we first compute some information for the atomic sub-constraints $x^2yz = 0$ and $xy^2z - 1 = 0$, then for $x^2yz = 0 \wedge xy^2z - 1 = 0$, $\exists z \in I_z\ x^2yz = 0 \wedge xy^2z - 1 = 0$, and so on. Along the way we try prove existential quantifiers or disprove universal quantifiers using this information.

Which information allows us to do this? Let us first consider a constraint of the form $\exists x \in I\ \phi$. In this case, knowing that ϕ has a solution in I suffices to prove the constraint—this solution could be a solution of an equality. However, information about one solution for all atomic sub-constraints is not enough: Consider a constraint of the form $\exists x \in I_x \exists y \in I_y\ \phi_1 \wedge \phi_2$. From knowing that both ϕ_1 and ϕ_2 have a solution within $I_x \times I_y$, we cannot infer that $\phi_1 \wedge \phi_2$ has a solution in $I_x \times I_y$, because the respective solutions might be different (see the left-hand side of Figure 1).

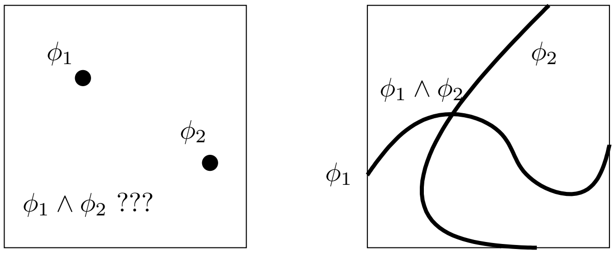


Fig. 1. Singular Solutions

Thus we have to propagate more information for proving existentially quantified conjunctions. In the example above, from knowing that *for all y in I_y there is an x in I_x such that* ϕ_1 and that *for all x in I_x there is a y in I_y such that* ϕ_2, and furthermore knowing that the solution for the existentially quantified variable depends continuously on the universally quantified one, we can prove $\exists x \in I_x \exists y \in I_y\ \phi_1 \wedge \phi_2$ (in other words, on the right-hand side of Figure 1, there is a line going from left to right, and a line going from top to bottom, and therefore there is a point on both lines).

One might think that information similar to the above is enough. But while it would be sufficient in the example above, in general, it does not propagate. For illustrating this, we assume that in the above example both ϕ_1 and ϕ_2 contain an additional parameter p, and assume that we have the information

forall p in I_p for all y in I_y there is an x in I_x such that ϕ_1

and

for all p in I_p for all x in I_x there is a y in I_y such that ϕ_2.

This information allows us to prove $\forall p \in I_p \exists x \in I_x \exists y \in I_y\ \phi_1 \wedge \phi_2$, but it does *not* allow us to propagate the according continuity information: In general, the x and y-coordinates of solutions of $\phi_1 \wedge \phi_2$ might not depend continuously on p! For example, the p- and x-coordinates of the solution might look like Figure 2, where for all p we can find an according x in the solution, but this solution can not be described by a continuous function.

In order to deal with this problem we use even more general information to describe such solutions: Given a constraint ϕ, a variable set $U = \{u_1, \ldots, u_k\}$ such that $U \subseteq \mathcal{V} = \{x_1, \ldots, x_{|\mathcal{V}|}\}$, and a box $B \subseteq \mathbb{R}^{|\mathcal{V}|}$, the information that we propagate is the predicate "there exist continuous $F_{x_1}, \ldots, F_{x_{|\mathcal{V}|}}$ (*witness functions*) in $\mathbb{R}^k \to \mathbb{R}$ such that for all $s_1, \ldots, s_k \in [0,1]^k$,

- for all $i \in \{1, \ldots, k\}$, $s_i = 0$ implies $F_{u_i}(s_1, \ldots, s_k) = \underline{\underline{B(u_i)}}$,
- for all $i \in \{1, \ldots, k\}$, $s_i = 1$ implies $F_{u_i}(s_1, \ldots, s_k) = \overline{\overline{B(u_i)}}$,
- for all $i \in \{1, \ldots, |\mathcal{V}|\}$, $F_{x_i}(s_1, \ldots, s_k) \in B(x_i)$,
- $\phi \frac{F_{x_1}(s_1, \ldots, s_k)}{x_1} \ldots \frac{F_{x_{|\mathcal{V}|}}(s_1, \ldots, s_k)}{x_{|\mathcal{V}|}}$."

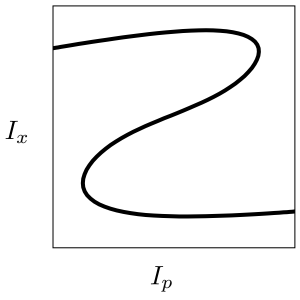


Fig. 2. Failed Propagation

The first item guarantees that there is a witness on the left box border (in Figure 2, the solution touches the box on the left-hand side). The second item does the same for the right box border (in Figure 2, the solution touches the box on the left-hand side). The third item requires the witness function to stay within the box, and the fourth item requires the witness function to describe a solution.

In the following we denote the above predicate by $M_\forall(\{u_1, \dots, u_k\}, \phi, B)$. We call such a predicate *witness predicate*, and k its *dimension*. As we have already seen above, this predicate implies the first-order predicate $\forall u_1 \in B_{u_1} \dots \forall u_k \in B_{u_k} \exists v_1 \in B_{v_1} \dots \exists v_l \in B_{v_l} \phi$. Moreover, for a closed constraint ϕ, a set of variables U and box B, $M_\forall(U, \phi, B)$ implies that ϕ holds. So this is the information we want to compute for proving the total constraint.

By propagating the above information directly, we can just prove existential quantifiers, but not disprove universal quantifiers. However, we can also use such witness predicates for the latter purpose: Just propagate this information also for the negation of the constraint. The next section will formalize the details.

Note that the above approach is not applicable, when the involved intervals are too large: First, the solution of the constraint might behave too wildly, making it impossible to describe it by witness functions (e.g., for the constraint $x^2+y^2 = 1$ within $[-2, 2] \times [-2, 2]$ no witness predicate with dimension higher than 0 holds, but one can easily find sub-boxes with witness predicates of dimension 1). And second, the larger a box, the higher is the effort to find such witness functions (e.g., if we use narrowing operators for box-consistency for computing witness functions—as described in Sections 5 and 6—then this will not succeed for too large boxes).

4 Computing and Propagating Witness Predicates

In this section we show formally how one can propagate witness predicates over the logical symbols $\wedge$, $\vee$, $\exists$, and $\forall$. We start with conjunctions:

Theorem 2. *If $M_\forall(U_1, \phi_1, B)$, $M_\forall(U_2, \phi_2, B)$, and $U_1 \cup U_2 = \mathcal{V}$, then $M_\forall(U_1 \cap U_2, \phi_1 \wedge \phi_2, B)$.*

Proof. The idea of the proof is that whenever the witness functions of ϕ_1 and of ϕ_2 have the same value, then this is a solution for $\phi_1 \wedge \phi_2$, and so the fourth property necessary for witness functions is fulfilled. We will prove that there exists such values and construct witness functions from them that fulfill the remaining properties.

Let $F^1_{x_1}, \ldots, F^1_{x_{|\mathcal{V}|}}$ be the witness functions of ϕ_1, and let $F^2_{x_1}, \ldots, F^2_{x_{|\mathcal{V}|}}$ be the witness functions of ϕ_2. For all $i \in \{1, \ldots, |\mathcal{V}|\}$, let $\mathcal{F}^1_i : [0,1]^{|U_1|+|U_2|} \to \mathbb{R}$ be such that $\mathcal{F}^1_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) := F^1_{x_i}(r_1, \ldots, r_{|U_1|})$ and let $\mathcal{F}^2_i : [0,1]^{|U_1|+|U_2|} \to \mathbb{R}$ be such that $\mathcal{F}^2_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) := F^2_{x_i}(r_{|U_1|+1}, \ldots, r_{|U_1|+|U_2|})$. We assume that in the order on $\mathcal{V}$ the elements of $U_1 \cap U_2$ appear first.

Let $\mathcal{G} : [0,1]^{|U_1|+|U_2|} \to \mathbb{R}^{|\mathcal{V}|}$ be the function for which the i-th component is $\mathcal{F}^1_{x_i} - \mathcal{F}^2_{x_i}$, where $i \in \{1, \ldots, |\mathcal{V}|\}$. For all elements $(r_1, \ldots, r_{|U_1|+|U_2|})$ of the solution set of $\mathcal{G} = 0$,

$$\phi_1 \frac{\mathcal{F}^1_{x_1}(r_1, \ldots, r_{|U_1|+|U_2|})}{x_1} \cdots \frac{\mathcal{F}^1_{x_{|\mathcal{V}|}}(r_1, \ldots, r_{|U_1|+|U_2|})}{x_{|\mathcal{V}|}}$$

and (note that we again use the witness functions of ϕ_1)

$$\phi_2 \frac{\mathcal{F}^1_{x_1}(r_1, \ldots, r_{|U_1|+|U_2|})}{x_1} \cdots \frac{\mathcal{F}^1_{x_{|\mathcal{V}|}}(r_1, \ldots, r_{|U_1|+|U_2|})}{x_{|\mathcal{V}|}}$$

and holds. So for these values $\phi_1 \wedge \phi_2$ holds, too. Now we show that such solutions of $\mathcal{G} = 0$ exist, and that they have the necessary properties to describe them by witness functions.

For proving existence we apply Theorem 1. For showing that $\mathcal{G}$ fulfills the necessary conditions observe that the dimension of its co-domain is $|\mathcal{V}| = |U_1 \cup U_2| = |U_1| + |U_2| - |U_1 \cap U_2|$. So we can interpret the first $|U_1 \cap U_2|$ variables of $\mathcal{G}$ as parameters. Then it suffices to prove that there is a bijection C between the variables $x_1, \ldots, x_{|\mathcal{V}|}$ and the coordinates of $\mathcal{G}$ that are no parameters, such that for an arbitrary, but fixed $i \in \{1, \ldots, |\mathcal{V}|\}$, for all $(r_1, \ldots, r_{|U_1|+|U_2|}) \in [0,1]^{|U_1|+|U_2|}$,

$$r_{C(i)} = 0 \text{ implies } \mathcal{F}^1_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) - \mathcal{F}^2_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) \leq 0, \quad (1)$$

and

$$r_{C(i)} = 1 \text{ implies } \mathcal{F}^1_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) - \mathcal{F}^2_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) \geq 0. \quad (2)$$

Here we have two cases:

- $x_i \in U_2$: In this case, let $C(i) := |U_1| + k$, where k is the number of x_i in U_2 (according to the order on $\mathcal{V}$). Therefore $C(i)$ points into the witness functions of ϕ_2, and according to the properties of these witness functions $r_{C(i)} = 0$ implies $\mathcal{F}^2_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) = \underline{B(i)}$, and $r_{C(i)} = 1$ implies $\mathcal{F}^2_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) = \overline{B(i)}$. As a consequence, since $\mathcal{F}^1_i(r_1, \ldots, r_{|U_1|+|U_2|}) \in B(i)$, the Inequalities 1 and 2 hold.

- Otherwise ($x_i \in U_1$ but not in U_2): In this case let $C(i)$ be the number of x_i in U_1. Since all variables in $U_1 \cap U_2$ appear before in the order, $C(i) > |U_1 \cap U_2|$, and it points to the coordinates of $\mathcal{G}$ that are no parameters. So according to the properties of these witness functions $r_{C(i)} = 0$ implies $\mathcal{F}^1_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) = \underline{B(i)}$, and $r_{C(i)} = 1$ implies $\mathcal{F}^1_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) = \overline{B(i)}$. As a consequence, since $\mathcal{F}^2_{x_i}(r_1, \ldots, r_{|U_1|+|U_2|}) \in B(i)$, the Inequalities 1 and 2 hold (up to a change of signs for which Theorem 1 is still valid).

So, by Theorem 1 there is a continuous function $S : [0,1]^{|U_1 \cap U_2|} \to [0,1]^{|\mathcal{V}|}$ such that for all $(p_1, \ldots, p_{|U_1 \cap U_2|}) \in [0,1]^{|U_1 \cap U_2|}$,

$$\mathcal{G}(p_1, \ldots, p_{|U_1 \cap U_2|}, S(p_1, \ldots, p_{|U_1 \cap U_2|})) = 0.$$

Therefore, the functions

$$\lambda p_1, \ldots, p_{|U_1 \cap U_2|} \,.\, \mathcal{F}^1_{x_i}(p_1, \ldots, p_{|U_1 \cap U_2|}, S(p_1, \ldots, p_{|U_1 \cap U_2|})),$$

where $i \in \{1, \ldots, |\mathcal{V}|\}$ provide the witness functions for $M_\forall(U_1 \cap U_2, \phi_1 \wedge \phi_2, B)$. It is easy to show that they fulfill the necessary properties. Note that one could construct the new witness functions from $\mathcal{F}^2_1, \ldots, \mathcal{F}^2_{|\mathcal{V}|}$ in a symmetric way. □

The disjunctive case is easier:

Theorem 3. *If $M_\forall(U_1, \phi_1, B)$ and $M_\forall(U_2, \phi_2, B)$ then $M_\forall(U_1, \phi_1 \vee \phi_2, B)$ and $M_\forall(U_2, \phi_1 \vee \phi_2, B)$.*

Proof. Both the witness functions of ϕ_1 and of ϕ_2 are also witness functions of $\phi_1 \vee \phi_2$. The necessary conditions are easy to check. □

For existential quantifiers propagation just removes the corresponding variable:

Theorem 4. *If $M_\forall(U, \phi, B)$ then $M_\forall(U \setminus \{x\}, \exists x \in B(x)\ \phi, B)$.*

Proof. Let $F_{x_1}, \ldots, F_{x_{|\mathcal{V}|}}$ be the witness functions corresponding to ϕ. We prove that they are also are witness functions for $M_\forall(U \setminus \{x_i\}, \exists x_i \in B(x_i)\ \phi, B)$. For this it is necessary to prove

$$\exists x_i \phi \frac{F_{x_1}(s_1, \ldots, s_k)}{x_1} \cdots \frac{F_{x_n}(s_1, \ldots, s_k)}{x_{|\mathcal{V}|}}$$

This clearly holds because F_{x_i} supplies the necessary value to satisfy the existential quantifier. □

Note that in the special case, where ϕ contains only the free variable x, the above theorem provides a witness predicate that proves the constraint $\exists x \in B(x)\ \phi$. The theorem corresponding to universal quantifiers is trivial to prove:

Theorem 5. *If $M_\forall(U, \phi, B)$ where U contains all the free variables of ϕ then $M_\forall(U, \forall x \in B(x)\ \phi, B)$.*

It is easy to check that the above theorems are optimal in the sense that in general $M_\forall$ does not hold for supersets of the ones provided.

It remains to show how to compute witness predicates for atomic (dis)equality constraints:

Theorem 6. *If for an equation $f = 0$ in variables $\{u_1, \ldots, u_k, v_1, \ldots, v_l\}$ and a box B for all $a_1 \in B(u_1)$,...,$a_k \in B(u_k)$,*

$$f \leq 0 \ \frac{a_1}{u_1} \ldots \frac{a_k}{u_k} \frac{\underline{B(v_1)}}{v_1} \ldots \frac{\underline{B(v_l)}}{v_l}$$

and

$$f \geq 0 \ \frac{a_1}{u_1} \ldots \frac{a_k}{u_k} \frac{\overline{B(v_1)}}{v_1} \ldots \frac{\overline{B(v_l)}}{v_l}$$

Then $M_\forall(\{u_1, \ldots, u_k\}, \phi, B)$.

Proof. Easy consequence of Theorem 1. □

Computing witness predicates for atomic inequality constraints is much easier, because in this case the solution has volume. For example we might try to prove that one of the box borders is contained in the solution set.

Given a first-order constraint with small quantification bounds we can now use Theorem 6 to compute witness predicates for atomic constraints, and Theorems 2, 3, 4, and 5 to propagate this information to witness predicates for the total constraint. Whenever we succeed to compute a witness predicate for the total constraint, we have proven it. It is trivial to write down an according algorithm.

For disproving a constraint we do the same on the negation of the constraint. Whenever we succeed to compute a witness predicate for the negation, we have disproven the original constraint.

5 Combination with First-Order Constraint Satisfaction

In this section we show how one can use the algorithm for computing witness predicates within the framework for continuous first-order constraint satisfaction described in an earlier paper [29]. For this we first introduce this framework informally, then give the formal details necessary for this paper, and afterwards combine it with the computation of witness predicates described in the last section.

For illustrating the main idea of first-order constraint satisfaction, we consider a bounded constraint of the form $(\exists y \in I_y \ \phi(x, y), I_x)$. The main idea is, to do an equivalence transformation on the constraint, replacing the intervals I_x and I_y by smaller ones. For example, we can replace the constraint $(\exists y \in [-2, 2] \ x^2 + y^2 \leq 1, [-2, 2])$, by the constraint $(\exists y \in [-1, 1] \ x^2 + y^2 \leq 1, [-1, 1])$. In this way we have, simultaneously, computed elements for which the constraint is false, and pruned the search space of existential quantifiers.

In a similar way, we can compute elements for which the constraint is false, and prune the search space of universal quantifiers: Replace the input constraint by its negation and push the negations down to the atomic constraints. By applying the procedure above to the result, we simultaneously compute elements for which the negated constraint is false, and prune the search space of its existential quantifiers. Thus we compute elements for which the original constraint is true, and pruned the search space of its universal quantifiers.

By the above process we can compute first-order constraints that are *first-order consistent*. Afterwards we branch the search space into pieces. For this we replace a sub-constraint of the form $\forall x \in I\ \phi$ by $\forall x \in I_1\ \phi\ \ \wedge \forall x \in I_2\ \phi$, or replace a sub-constraint of the form $\exists x \in I\ \phi$ by $\exists x \in I_1\ \phi\ \ \wedge \exists x \in I_2\ \phi$, where $I = I_1 \cup I_2$.

By iterating these steps one arrives at a branch-and-prune algorithm for proving and solving first-order constraints. This approach is in general successful for inputs that contain inequality constraints but no equality constraints. For extending the approach to equalities one needs a method for proving existential quantifiers and disproving universal quantifiers even if the according witnesses do not have volume.

Now let us formalize the process described above. For pruning we use the following:

Definition 1. *A* narrowing operator *is a function N on bounded constraints such that for bounded constraints (ϕ, B), and $(\phi', B') = N(\phi, B)$,*

- *$B \supseteq B'$ (contractance),*
- *for all $x \in B'$, $\phi\frac{x}{\mathcal{V}}$ iff $\phi'\frac{x}{\mathcal{V}}$ (soundness),*
- *there is no x such that $x \in \widehat{B \setminus B'}$ and $\phi\frac{x}{\mathcal{V}}$ (completeness).*

Note that here we split the original correctness condition [1] into soundness and completeness, and ignore some further properties that are not relevant here. Furthermore, the completeness condition puts a condition also on the borders of B' that have been narrowed by taking the closure $\widehat{B \setminus B'}$ instead of the set $B \setminus B'$.

Now, given a narrowing operator A for atomic constraints, we get a narrowing operator for conjunctions using the usual fix-point computation scheme. For the other logical symbols we can use the following rules (examples and properties of such a narrowing operator can be found in an earlier paper [29]):

Definition 2.

- $N\!\upharpoonleft_A (\phi_1 \vee \phi_2, B) = (\phi_1' \vee \phi_2', B_1' \uplus B_2')$
  where $(\phi_1', B_1') = N\!\upharpoonleft_A (\phi_1, B)$ *and* $(\phi_2', B_2') = N\!\upharpoonleft_A (\phi_2, B)$
- $N\!\upharpoonleft_A (\exists x \in I\ \phi, B) = (\exists x \in B'(x)\ \phi', B')$,
  where $(\phi', B') = N\!\upharpoonleft_A (\phi, B\frac{I}{x})$
- $N\!\upharpoonleft_A (\forall x \in I\ \phi, B) = (\forall x \in I\ \phi', D)$,
  where $(\phi', B') = N\!\upharpoonleft_A (\phi, B\frac{I}{x})$
  and $d \in D$ iff for all $r \in I$, $d\frac{r}{x} \in B'$

Now observe that first-order narrowing exactly computes the information needed for computing witness predicates. Namely, for bounded constraints of the form $(f = 0, B)$, where r is an equality or disequality, it uses the narrowing operator A to remove elements from the box B, for which $f \neq 0$, that is for which $f \leq 0$ or $f \geq 0$. This is exactly the information needed for Theorem 6.

Here we only use the case of Theorem 6 where $l = 1$, that is, we only compute witnesses of dimension $|\mathcal{V}|-1$ for atomic constraints. As this is the strongest case, this allows us to infer all the information that can be inferred by our propagation scheme.

The problem here is that, in general, the bounded constraint resulting from atomic narrowing, is not a sub-constraint of the final, first-order consistent constraint (because the conditions on the boxes are not fulfilled). This can occur in the case of disjunctions, where the narrowing operator proceeds according to the rule $N(\phi_1 \vee \phi_2, B) = N(\phi_1' \vee \phi_2', B_1' \uplus B_2')$ where $(\phi_1', B') = N(\phi_1, B)$ and $(\phi_2', B') = N(\phi_2, B)$. This means that we cannot use Theorem 3 for propagating the witness predicates computed for (ϕ_1', B_1') and (ϕ_2', B_2'). However, according to Theorem 7 below, witness predicates for the sub-constraints are only computed when this situation does not occur.

Theorem 7. *Let $(\phi', B') := N(\phi, B)$ such that $M_\forall(\{u_1, \ldots, u_k\}, \phi, B)$. Then, for all $i \in \{1, \ldots, k\}$, $B'(u_i) = B(u_i)$.*

Proof. Let i be such that $u_i \in U$. We have to prove that $B(u_i) = B'(u_i)$. The witness predicate tells us that there is a continuous function F_{u_i}, such that for all $(s_1, \ldots, s_k) \in [0,1]^k$, $s_i = 0$ implies $F_{u_i}(s_1, \ldots, s_k) = \underline{B(u_i)}$. By the fourth condition of witness predicates, this means that

$$\phi \frac{F_{x_1}(s_1, \ldots, s_k)}{x_1} \ldots \underline{B(u_i)} \ldots \frac{F_{x_n}(s_1, \ldots, s_k)}{x_n}$$

Thus the left border of $B(u_i)$ contains an element of ϕ. This implies that according to the completeness of narrowing operators, $\underline{B'(u_i)}$ has to be equal to $\underline{B(u_i)}$. In a similar way we proceed for the right border $\overline{B'(u_i)}$. □

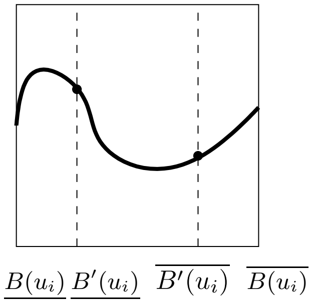
The only problem left, is to study, how branching influences the computed witness functions. Here we can use the following:

Theorem 8. *If $M_\forall(U, \phi, B)$, $B' \subseteq B$, and $\{v | B'(v) \neq B(v)\} \subseteq U$ then $M_\forall(U, \phi, B')$.*

Proof. Assume $M_\forall(U, \phi, B)$ with witness functions $F_{x_1}, \ldots, F_{x_{|\mathcal{V}|}}$, and B' such that $B' \subseteq B$ and $\{v | B'(v) \neq B(v)\} \subseteq U$. We have to prove that $M_\forall(U, \phi, B')$, that is, we have to find according witness functions.

Figure 5 illustrates the idea: The horizontal axis corresponds to a variable in U and the vertical axis corresponds to another variable. The new box B' has been narrowed along the variables in U. The old witness functions intersect the new box borders at certain points. We have to construct new witness functions that connect the leftmost intersection point with the right-most one.

This means that for the variables not in U, the new witness functions can correspond to the old ones. For the other variables, let $U = \{u_1, \ldots, u_k\}$. By continuity of $F_{u_1}, \ldots, F_{u_k}$ and the Boltzmann intermediate value theorem we know



that there are $\underline{s_1} \geq 0, \ldots, \underline{s_k} \geq 0$ such that for all $i \in \{1, \ldots, k\}$, $F_{u_i}(\underline{s_1}, \ldots, \underline{s_k}) = \underline{B'(u_i)}$ and $\overline{s_1} \leq 1, \ldots, \overline{s_k} \leq 1$ such that for all $i \in \{1, \ldots, k\}$, $F_{u_i}(\overline{s_1}, \ldots, \overline{s_k}) = \overline{B'(u_i)}$. Let $\underline{s_1} \geq 0, \ldots, \underline{s_k} \geq 0$ be the greatest such elements, and $\overline{s_1} \leq 1, \ldots, \overline{s_k} \leq 1$ the smallest one. For all $i \in \{1, \ldots, i\}$, we choose F'_{u_i} such that for all $(s_1, \ldots, s_k)$,

$$F'_{u_i}(s_1, \ldots, s_k) = F_{u_i}\left(\underline{s_1} + s_1(\overline{s_1} - \underline{s_1}), \ldots, \underline{s_1} + s_k(\overline{s_k} - \underline{s_k})\right).$$

These witness functions trivially fulfill the first, second and fourth condition needed for witness predicates. For proving the third condition, we consider two cases:

- The case where $i \in \{1, \ldots, \mathcal{V}\}$ such that $x_i \in U$. In this case $B'(x_i) \subseteq B(x_i)$. Then, for all $(s_1, \ldots, s_k) \in [0,1]^k$, $F'_{x_i}(s_1, \ldots, s_k) \geq \underline{B(x_i)}$, because this corresponds to a value of F'_{x_i} at a value greater than $\underline{s_1}, \ldots, \underline{s_k}$. In a similar way, $F'_{x_i}(s_1, \ldots, s_k) \leq \overline{B(x_i)}$. Thus $F_{x_i}(s_1, \ldots, s_k) \in B(x_i)$.
- The case where $i \in \{1, \ldots, |\mathcal{V}|\}$ such that $x_i \notin U$. In this case $B'(x_i) = B(x_i)$, and the condition is trivially fulfilled.

□

However, this theorem cannot be extended to the case where B' differs from B at the variable not in U: In this case the resulting witness functions could leave the box B'.

6 Implementation and Discussion

More research is needed to turn the above propagation scheme into a successful implementation. The difficulty is that it does not suffice to apply the propagation scheme to constraints where the bounds are sufficiently small. In addition, the sizes of the individual quantification bounds should fulfill a certain relation. For seeing this, take the example of a bounded constraint $(\exists y \in I_y \;\; \phi, I_x)$. Here, whenever for the elements in the solution set of ϕ the value of y depends strongly on x, the size of I_x should be much smaller than the size of I_y. That is, in the left-hand side of Figure 3, instead of the solid box, we should have a box with the dashed borders. But, in the case where for the elements in the solution set of ϕ the value of y does not depend strongly on x (see the right-hand side of Figure 3), this is not necessary. However, in general one cannot do without branching in the I_y direction, because there might be a high (even) number of solutions in this direction, which would need a lot of branching to separate them.

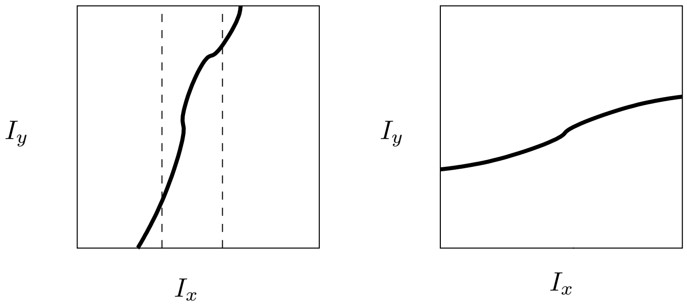


Fig. 3. Needed Quantification Bounds

Therefore we need heuristics for branching that take into account the computation of witness predicates, and choose the branch points accordingly.

Still, in order to check the general feasibility of the method, we have implemented it with simple largest-first branching heuristics. As expected, the implementation does not succeed in examples with a solution structure similar to the one on the left-hand side of Figure 3. However, for examples similar to the one on the right-hand side, the method is successful already now.

Take the input constraint

$$\forall x \in [-10, 10]\ \exists y \in [-2, 2]\ \exists z \in [-2, 2]$$
$$y + 0.1x^2yz^2 + 0.2y^2z = 1\ \wedge\ z + 0.1x^2y^2z + 0.2x^2z = 1$$

The state-of-the-art symbolic software QEPCAD [11] does not produce a solution for this constraint within a minute on a 600MhZ GNU/Linux PC (note that we ran the software as a dumb user—its performance might be significantly higher when setting special switches according to the problem). Our implementation proves the constraint in negligible time (in the range of milliseconds). The reason is, that QEPCAD has to take a huge effort to compute the exact witnesses proving the existential quantifiers (e.g., computation with real algebraic numbers, square-free basis, polynomial remainder sequences). However, this effort is not needed here, and much faster computation with floating point intervals suffices.

Thus we have reason to believe that—similar to the case of constraints without (dis)equalities—the method based on constraint satisfaction techniques will on average (i.e., for numerically well-posed inputs [30]) be significantly faster than current symbolic methods.

7 Conclusion

In this paper we have lain the basis for using constraint programming techniques for proving first-order constraints that contain (dis)equalities. We have shown the usefulness of the approach on a non-trivial example. The main problem left to arrive at a practically useful implementation, is the combination with box splitting heuristics. Additional future research will include:

- Generalizing and adapting preconditioning [15, 14, 25] to first-order constraints. A promising starting point is incremental preconditioning [5].
- Studying the numerical stability of first-order constraints with equalities—extending the earlier work for first-order constraint without equalities [30]. This will result in a description of the first-order constraint that algorithms based on approximation (e.g., floating-point arithmetic) can solve in principle.
- Speeding up computation by using numerical optimizers/root finders.
- Using witnesses of lower dimension.

The author thanks to Arnold Neumaier for important suggestions. This work has been supported by a Marie Curie fellowship of the European Union under contract number HPMF-CT-2001-01255.

References

1. F. Benhamou. Interval constraint logic programming. In A. Podelski, editor, *Constraint Programming: Basics and Trends*, volume 910 of *LNCS*. Springer Verlag, 1995.
2. F. Benhamou and F. Goualard. Universally quantified interval constraints. In *Proc. of the Sixth Intl. Conf. on Principles and Practice of Constraint Programming (CP'2000)*, number 1894 in LNCS, Singapore, 2000. Springer Verlag.
3. F. Benhamou, D. McAllester, and P. V. Hentenryck. CLP(Intervals) Revisited. In *International Symposium on Logic Programming*, pages 124–138, Ithaca, NY, USA, 1994. MIT Press.
4. B. F. Caviness and J. R. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition*. Springer, 1998.
5. C.-K. Chiu and J. H. man Lee. Efficient interval linear equality solving in constraint logic programming. *Reliable Computing*, 8:139–174, 2002.
6. G. E. Collins. Quantifier elimination for the elementary theory of real closed fields by cylindrical algebraic decomposition. In *Second GI Conf. Automata Theory and Formal Languages*, volume 33 of *LNCS*, pages 134–183. Springer Verlag, 1975. Also in [4].
7. Z. Danqing, L. Weiguo, and S. Zuhe. Solving underdetermined systems with interval methods. *Reliable Computing*, 5(1):23–33, 1999.
8. E. Hansen. *Global Optimization Using Interval Analysis*. Marcel Dekker, Inc, 1992.
9. P. V. Hentenryck, D. McAllester, and D. Kapur. Solving polynomial systems using a branch and prune approach. *SIAM Journal on Numerical Analysis*, 34(2), 1997.
10. P. V. Hentenryck, L. Michel, and Y. Deville. *Numerica: A Modeling Language for Global Optimization*. The MIT Press, 1997.

11. H. Hong. *Improvements in CAD-based Quantifier Elimination*. PhD thesis, The Ohio State University, 1990.
12. H. Hong. Symbolic-numeric methods for quantified constraint solving. In *International Symposium on Scientific Computing, Computer Arithmetic and Validated Numerics SCAN-95*, 1995. Invited Talk.
13. L. Jaulin and É. Walter. Guaranteed tuning, with application to robust control and motion planning. *Automatica*, 32(8):1217–1221, 1996.
14. R. B. Kearfott. Preconditioners for the interval gauss-seidel method. *SIAM Journal on Numerical Analysis*, 27(3):804–822, 1990.
15. R. B. Kearfott. A review of preconditioners for the interval gauss-seidel method. *Interval Computations*, 1(1):59–85, 1991.
16. R. B. Kearfott. *Rigorous Global Search: Continuous Problems*. Kluwer Academic Publishers, 1996.
17. R. B. Kearfott. On proving existence of feasible points in equality constrained optimization problems. *Mathematical Programming*, 83(1):89–100, 1998.
18. L. V. Kolev and I. P. Nenov. Cheap and tight bounds on the solution set of perturbed systems of nonlinear equations. *Reliable Computing*, 7(5):399–408, 2001.
19. W. Kulpa. The Poincaré-Miranda theorem. *The American Mathematical Monthly*, 104(6):545–550, 1997.
20. S. Malan, M. Milanese, and M. Taragna. Robust analysis and design of control systems using interval arithmetic. *Automatica*, 33(7):1363–1372, 1997.
21. G. Mayer. Epsilon-inflation in verification algorithms. *Journal of Computational and Applied Mathematics*, 60:147–169, 1994.
22. C. Miranda. Un 'osservazione su un teorema di Brouwer. *Bol. Un. Mat. Ital. Ser II*, 3:5–7, 1941.
23. A. Neumaier. The enclosure of solutions of parameter-dependent systems of equations. In R. E. Moore, editor, *Reliability in Computing*, pages 269–286. Academic Press, 1988.
24. A. Neumaier. Rigorous sensitivity analysis for parameter-dependent systems of equations. *Journal of Mathematical Analysis and Applications*, 144:16–25, 1989.
25. A. Neumaier. *Interval Methods for Systems of Equations*. Cambridge Univ. Press, Cambridge, 1990.
26. S. Ratschan. Applications of real first-order constraint solving — bibliography. `http://www.risc.uni-linz.ac.at/people/sratscha/appFOC.html`, 2001.
27. S. Ratschan. Real first-order constraints are stable with probability one. `http://www.risc.uni-linz.ac.at/people/sratscha/preprints`, 2001. Draft.
28. S. Ratschan. Approximate quantified constraint solving by cylindrical box decomposition. *Reliable Computing*, 8(1):21–42, 2002.
29. S. Ratschan. Continuous first-order constraint satisfaction. In *Proceedings of Artificial Intelligence and Symbolic Computation*, LNCS. Springer, 2002. To appear, `http://www.risc.uni-linz.ac.at/people/sratscha/preprints`.
30. S. Ratschan. Quantified constraints under perturbations. *Journal of Symbolic Computation*, 33(4):493–505, 2002.
31. A. Tarski. *A Decision Method for Elementary Algebra and Geometry*. Univ. of California Press, Berkeley, 1951. Also in [4].
32. V. Weispfenning. Quantifier elimination for real algebra - the quadratic case and beyond. *Applicable Algebra in Engineering, Communication and Computing*, 8(2):85–101, 1997.
33. M. A. Wolfe. On bounding solutions of underdetermined systems. *Reliable Computing*, 7(3):195–207, 2001.