Article Submitted to Journal of Symbolic Computation

Quantified Constraints Under Perturbation

Stefan Ratschan

Research Institute for Symbolic Computation,
Johannes Kepler University, A-4040 Linz, Austria
e-mail: Stefan.Ratschan@risc.uni-linz.ac.at
www: http://www.risc.uni-linz.ac.at/people/sratscha

Abstract

Quantified constraints (i.e., first-order formulae over the real numbers) are often exposed to perturbations: Constants that come from measurements usually are only known up to certain precision, and numerical methods only compute with approximations of real numbers. In this paper we study the behavior of quantified constraints under perturbation by showing that one can formulate the problem of solving quantified constraints as a nested parametric optimization problem followed by one sign computation. Using the fact that minima and maxima are stable under perturbation, but the sign of a real number is stable only for non-zero inputs, we derive practically useful conditions for the stability of quantified constraints under perturbation.

1. Introduction

Let a quantified constraint be a first-order formula over the real numbers. This means that it contains quantifiers ($\exists$, $\forall$), connectives ($\wedge$, $\vee$, $\neg$), predicate symbols (e.g., $=$, $<$, $\leq$), function symbols (e.g., $+$, $-$, $\times$, sin, exp), rational constants and variables ranging over real numbers. Solving such constraints has various applications: Studying the stability of differential equations [Liska and Steinberg, 1993, Hong et al., 1997], control theory [Dorato et al., 1997, Jirstrand, 1997, Dorato, 2000], and various other areas [Sturm, 1999, Ioakimidis, 1997, Ratschan, 2001a]. Traditionally, quantified constraints have been solved by symbolic methods [Tarski, 1951, Collins, 1975, Collins and Hong, 1991, Loos and Weispfenning, 1993], but lately (validated) numerical methods appeared for special cases [Shary, 1996a, 1999, Benhamou and Goualard, 2000, Garloff and Graf, 1999, Jaulin and Walter, 1996, Malan et al., 1997] and the general case [Ratschan, 2002, 2001b].

Numerical methods introduce perturbations, for example via rounding errors. Furthermore, constants that come from measurements usually are known only up

to uncertainties. So it is of fundamental importance, for *any* method that involves approximation, to know how the result depends on input perturbations [Higham, 1996, Wilkinson, 1964].

However, the according dependence of the solution set of quantified constraints on input perturbations has not been studied, yet. This has been a major obstacle to the design of the above algorithms and to studying some of their properties (termination, complexity, etc.). We try to remove this obstacle by the following contributions of this paper:

- We show that the solution set of (closed) quantified constraints is stable under perturbation except for one (unstable) sign computation. More specifically, we formulate quantified constraint solving as a nested parametric optimization problem followed by the computation of the sign of the resulting optimum.
- We provide two additional conditions for stability, one considering the behavior under exchanging strict inequalities with non-strict inequalities and one considering the volume of the occurring solution sets.
- We show that, for constraints containing only inequality predicates, stability under perturbation is independent of whether we use information about the continuity of constraints and equality of variables.

The paper is self-contained except for some basic knowledge of mathematical logic [Ebbinghaus et al., 1984]. Its structure is as follows: In Section 2, we give a formalization of the problem. In Section 3, we show that for a certain class of quantified constraints their solution set is stable under perturbation except for one sign computation. In Section 4, we present criteria that are useful in checking whether a given constraint is stable. In Section 5, we discuss various more general forms of perturbation. In Section 6, we discuss the generalization of the preceding results to arbitrary constraints, and in Section 7, we conclude the paper.

This work was supported by the Austrian Science Fund FWF in the frame of the project SFB F1303.

2. Stability under Perturbation

In this section we will make precise the notions of "perturbation", and "stability under perturbation". Here we will proceed in a similar way as usual in numerical analysis. However, we cannot adopt the according definitions directly, because quantified constraints have a structure that is quite different from the structure of the usual objects occurring in numerical analysis.

Let us first discuss some preliminaries: We fix a set V of variables with a total order on it. We denote by C the set of all (quantified) constraints with variables in V and with function symbols that denote continuous functions. Furthermore we require all atomic sub-constraints to be of the form θ `r` `0`, where θ is a term

and $\mathtt{r}$ a predicate symbol. The zero on the right-hand sides will give us a handle for perturbing atomic constraints.

We typeset expressions in the object language (i.e., constraints and all their syntactic components) in typewriter font. The symbol ϕ will function as a meta-variable over constraints, and the symbol θ will function as a meta-variable over terms. For any syntactic entity $\mathtt{s}$ (i.e., constraint, term, predicate symbol, function symbol) we denote its meaning by $[\![\mathtt{s}]\!]$. The meaning of a constraint is a solution set (a subset of $\mathbb{R}^{|V|}$), the meaning of a term is a function in $\mathbb{R}^{|V|} \to \mathbb{R}$, and the meaning $[\![\mathtt{x}]\!]$ of a variable $\mathtt{x}$ is a projection function (i.e., $[\![\mathtt{x}]\!](a_1, \ldots, a_{|V|}) = a_k$, where $\mathtt{k}$ is the k-th variable in V).

We assume that each occurring variable $\mathtt{x}$ ranges over a compact subset of the reals (e.g., a closed, finite interval) which we denote by $R_{\mathtt{x}}$. Furthermore we denote by R_V the Cartesian product of the ranges of all variables in V (according to their order). Such ranges arise naturally in most practical applications. They significantly simplify the exposition in this paper, although most of the results can be straightforwardly generalized to the non-compact case by introducing infinitely small and infinitely large elements.

We assume that each solution set of a constraint ϕ in C has a certain volume $\mathrm{Vol}([\![\phi]\!])$, where—as usual—the volume of a set is defined as the integral of the characteristic function of this set. So we have to take care that the used function and predicate symbols only result in solution sets for which we can integrate the characteristic function (this holds easily for constraints that only contain the predicate symbols $=$ and $\leq$, and the function symbols $+$ and $\times$). In $\mathbb{R}^0$ we assign the volume zero to the empty set and infinite volume to the set containing the empty tuple.

Now we proceed by clarifying the notion of perturbation. For this we need to know how to measure distance:

Definition: A function $d : A \times A \to \mathbb{R}_0^+$ is a *pseudo-metric* iff

- for all $x \in A$, $d(x, x) = 0$,
- for all $x, y \in A$, $d(x, y) = d(y, x)$, and
- for all $x, y, z \in A$, $d(x, z) \leq d(x, y) + d(y, z)$.

For solution sets x and y we define their distance $d(x, y)$ as the volume of the symmetric set difference (i.e., $\mathrm{Vol}(x \nabla y)$, where $x \nabla y = (x \cup y) \setminus (x \cap y)$). For vectors and functions we define their distance as the supremum of the distance of their corresponding elements, and for real constants x and y as $d(x, y) := |x - y|$.

A quantified constraint contains many entities that one could perturb. Observe that one can view a constraint as a mapping that takes the meaning of its constant symbols (i.e., real constants), and its variables (i.e., projection functions) and produces the according meaning (i.e., solution set) of the constraint. These two kinds of inputs often occur in perturbed form: Constants that come from measurements in nature are known only up to a certain accuracy; computation

with approximations to projection functions can be cheaper than computation with the exact projection functions [Ratschan, 2000c, 2002]; and rounding errors change these objects if we represent them by floating-point numbers. So we study perturbations of constant symbols and variables in this paper.

However, in order to keep the exposition simple, we will first do the main development only with perturbations of the zeros that occur on the right-hand side of predicates in atomic constraints. In Section 5 we will generalize the approach to perturbations of arbitrary constant symbols and variables.

Definition: Given a non-negative integer n and a $\delta \in \mathbb{R}^+$, an *n-ary δ-perturbation* is any real tuple $(c_1, \ldots, c_n)$ such that $d((c_1, \ldots, c_n), (0, \ldots, 0)) \leq \delta$. Given constraints ϕ and ϕ', each containing n atomic sub-constraints, ϕ' is the *result of a δ-perturbation of ϕ*, iff there is an n-ary δ-perturbation $(c_1, \ldots, c_n)$, such that ϕ' results from ϕ by replacing the k-th zero on the right-hand side of an atomic sub-constraint of ϕ with c_k.

Now we can use the notion of continuity from analysis for defining stability:

Definition: A function $f : C \to A$, where A is a set with a pseudo-metric d defined on it, is *stable for a constraint ϕ* iff for all real $\varepsilon > 0$ there exists a $\delta > 0$ such that for all ϕ' resulting from a δ-perturbation of ϕ, $d(f(\phi), f(\phi')) < \varepsilon$. A constraint ϕ is *stable* iff the semantic functional $[\![\cdot]\!]$ is stable for ϕ.

As an example for an unstable constraint ϕ consider $\exists \mathtt{x}\ \mathtt{x}^2 < 0$, for which $[\![\phi]\!]$ is $\{\}$ (i.e., ϕ is false). Choose $\varepsilon = \mathrm{Vol}(R_x)$ and let $\delta > 0$ be arbitrary but fixed. Then the tuple (δ) is a unary δ-perturbation, but for the result $\exists \mathtt{x}\ \mathtt{x}^2 < \delta$ of this perturbation of ϕ, which we denote by ϕ', we have $[\![\phi']\!] = R_x$ (i.e., ϕ' is true). Thus $d([\![\phi]\!], [\![\phi']\!]) = \mathrm{Vol}(R_x)$, independently of δ, and because $\mathrm{Vol}(R_x)$ is not smaller than ε, ϕ is unstable.

3. Stability Except for One Sign Computation

In this section we formulate quantified constraint solving as a nested parametric optimization problem followed by the computation of the sign of the resulting optimum. For this we restrict ourselves to a certain class of constraints, and show later, how to extend the results to the stability of general constraints.

Definition: A constraint is *monotonic* iff it does not contain negations and all of its atomic constraints either have the form $\theta > \mathtt{c}$ or the form $\theta \geq \mathtt{c}$.

We can transform arbitrary constraints to monotonic constraints as follows:

1. Push negations inside as far as possible,
2. Replace $\neg\theta < \mathtt{0}$ by $\theta \geq \mathtt{0}$, $\neg\theta \leq \mathtt{0}$ by $\theta > \mathtt{0}$, $\neg\theta > \mathtt{0}$ by $\theta \leq \mathtt{0}$, $\neg\theta \geq \mathtt{0}$ by $\theta < \mathtt{0}$, $\neg\theta = \mathtt{0}$ by $\theta \neq \mathtt{0}$, and $\neg\theta \neq \mathtt{0}$ by $\theta = \mathtt{0}$,

3. Replace atomic constraints of the form $\theta = \mathtt{0}$ by $\theta \leq \mathtt{0} \ \wedge \ \theta \geq \mathtt{0}$, and $\theta \neq \mathtt{0}$ by $\theta < \mathtt{0} \ \vee \ \theta > \mathtt{0}$, and
4. Replace atomic constraints of the form $\theta < \mathtt{0}$ by $-\theta > \mathtt{0}$, and $\theta \leq \mathtt{0}$ by $-\theta \geq \mathtt{0}$.

For a constraint ϕ we denote the result of this transformation by $T(\phi)$ (for example, $T(\mathtt{x}^2 = \mathtt{0} \wedge \mathtt{y} < \mathtt{0}) = -\mathtt{x}^2 \geq \mathtt{0} \wedge \mathtt{x}^2 \geq \mathtt{0} \wedge \mathtt{y} > \mathtt{0}$).

Now one can easily prove by induction over the structure of monotonic constraints that such constraints are monotonic in the following sense:

LEMMA 3.1: *For every monotonic constraint ϕ and the result ϕ' of a perturbation of ϕ with non-negative elements only, $[\![\phi]\!] \supseteq [\![\phi']\!]$.*

For monotonic constraints we can replace the discrete notion of truth by a continuous concept (a similar idea has been used by S. Shary [1996b] in a different context):

Definition: For every monotonic constraint ϕ in C, let $\tau(\phi)$, the *degree of truth*, be the following function in $\mathbb{R}^{|V|} \to \mathbb{R}$:

- $\tau(\theta \ \mathtt{r} \ \mathtt{0})(a_1, \ldots, a_{|V|}) = [\![\theta]\!](a_1, \ldots, a_{|V|})$
- $\tau(\phi_1 \wedge \phi_2)(a_1, \ldots, a_{|V|}) = \min\{\tau(\phi_1)(a_1, \ldots, a_{|V|}), \tau(\phi_2)(a_1, \ldots, a_{|V|})\}$
- $\tau(\phi_1 \vee \phi_2)(a_1, \ldots, a_{|V|}) = \max\{\tau(\phi_1)(a_1, \ldots, a_{|V|}), \tau(\phi_2)(a_1, \ldots, a_{|V|})\}$
- $\tau(\forall \mathtt{x} \ \phi')(a_1, \ldots, a_{|V|}) = \inf\{\tau(\phi')(a_1, \ldots, a_{k-1}, u, a_{k+1}, \ldots, a_{|V|}) | u \in R_{\mathtt{x}}\}$, where $\mathtt{x}$ is the k-th variable in V.
- $\tau(\exists \mathtt{x} \ \phi')(a_1, \ldots, a_{|V|}) = \sup\{\tau(\phi')(a_1, \ldots, a_{k-1}, u, a_{k+1}, \ldots, a_{|V|}) | u \in R_{\mathtt{x}}\}$, where $\mathtt{x}$ is the k-th variable in V.

One can view this definition as the interpretation of constraints in an infinitely-many-valued logic [Rescher, 1969, Urquhart, 1986]. Note that the suprema and infima are always attained because every variable $\mathtt{x}$ ranges over a compact set $R_{\mathtt{x}}$ and all function symbols denote continuous functions only. Note also that the degree of truth of closed constraints assigns the same real number everywhere—independent of its argument.

Since we can easily transform every constraint to a monotonic constraint, we accordingly extend the definition of τ to non-monotonic constraints (in fact one could also extend the above definition for constraints containing negations by interpreting negation as multiplication with -1, but this would have the danger of confusing intuition).

Here are some examples (in each case we assume that V is equal to the variables in the according constraint, that a, a_0, a_1 are arbitrary real numbers, and that $R_{\mathtt{x}}$ and $R_{\mathtt{y}}$ contain zero):

- $\tau(\exists \mathtt{x} \ -\mathtt{x}^2 \geq \mathtt{0})(a) = \sup\{\tau(-\mathtt{x}^2 \geq \mathtt{0})(u) \mid u \in R_{\mathtt{x}}\} = 0$

- $\tau(\exists \mathtt{x}\ -\mathtt{x}^2+\mathtt{1}\geq \mathtt{0})(a) = \sup\{\tau(-\mathtt{x}^2+\mathtt{1}\geq\mathtt{0})(u) \mid u \in R_{\mathtt{x}}\} = 1$
- $$\begin{aligned}\tau(\exists \mathtt{x}\ \forall \mathtt{y}\ -\mathtt{x}^2+\mathtt{y}^2\geq \mathtt{0})(a_0,a_1) = \\ \sup\{\tau(\forall \mathtt{y}\ -\mathtt{x}^2+\mathtt{y}^2\geq \mathtt{0})(u_0)|u_0\in R_{\mathtt{x}}\} = \\ = \sup\{\inf\{\tau(-\mathtt{x}^2+\mathtt{y}^2)(u_0,u_1)|u_1\in R_{\mathtt{y}}\}|u_0\in R_{\mathtt{x}}\} = \\ = \sup\{\inf\{-u_0^2+u_1^2|u_1\in R_{\mathtt{y}}\}|u_0\in R_{\mathtt{x}}\} = \sup\{-u_0^2|u_0\in R_{\mathtt{x}}\} = 0\end{aligned}$$

In certain cases the degree of truth captures the semantics of constraints:

LEMMA 3.2: *For every monotonic constraint ϕ that contains $\geq$ predicates only, for all $(a_1,\ldots,a_{|V|})$,*

$$(a_1,\ldots,a_{|V|})\in[\![\phi]\!] \text{ iff } \tau(\phi)(a_1,\ldots,a_{|V|})\geq 0.$$

Proof: We proceed by recursion over the structure of constraints. The lemma obviously holds for atomic constraints. Now consider constraints of the following forms:

$\phi_1\wedge\phi_2$**:** Let $(a_1,\ldots,a_{|V|})$ be arbitrary but fixed. Then

$$(a_1,\ldots,a_{|V|})\in[\![\phi_1\wedge\phi_2]\!]$$

iff both

$$(a_1,\ldots,a_{|V|})\in[\![\phi_1]\!] \text{ and } (a_1,\ldots,a_{|V|})\in[\![\phi_2]\!]$$

iff (by induction hypothesis)

$$\tau(\phi_1)(a_1,\ldots,a_{|V|})\geq 0 \text{ and } \tau(\phi_2)(a_1,\ldots,a_{|V|})\geq 0$$

iff

$$\min\{\tau(\phi_1)(a_1,\ldots,a_{|V|}),\tau(\phi_2)(a_1,\ldots,a_{|V|})\}\geq 0$$

iff (by definition of τ)

$$\tau(\phi_1\wedge\phi_2)(a_1,\ldots,a_{|V|})\geq 0.$$

$\exists \mathtt{x}\ \phi$**:** Let $(a_1,\ldots,a_{|V|})$ be arbitrary but fixed, and assume that $\mathtt{x}$ is the k-th variable in V. Then $(a_1,\ldots,a_{|V|})\in[\![\exists \mathtt{x}\ \phi]\!]$ iff

there is an $u\in R_x$, for which $(a_1,\ldots,a_{k-1},u,a_{k+1},\ldots,a_{|V|})\in[\![\phi]\!]$,

iff (induction hypothesis)

there is an $u\in R_x$ such that $\tau(\phi)(a_1,\ldots,a_{k-1},u,a_{k+1},\ldots,a_{|V|})\geq 0$

iff (by compactness of $R_{\mathtt{x}}$ and continuity the supremum is attained)

$$\sup\{\tau(\phi)(a_1,\ldots,a_{k-1},u,a_{k+1},\ldots,a_{|V|})|u\in R_{\mathtt{x}}\}\geq 0$$

iff (by definition of τ)

$$\tau(\exists \mathtt{x}\ \phi)(a_1,\ldots,a_{|V|})\geq 0.$$

The cases for disjunction and universal quantification are dual. □

One can easily adapt the above proof to get the analogous lemma for strict inequalities:

LEMMA 3.3: *For every monotonic constraint ϕ in C that contains $>$ predicates only, for all $(a_1, \ldots, a_{|V|})$,*

$$(a_1, \ldots, a_{|V|}) \in [\![\phi]\!] \text{ iff } \tau(\phi)(a_1, \ldots, a_{|V|}) > 0.$$

By combining Lemma 3.2 and Lemma 3.3 we know that one can extract the meaning of a constraint from the sign of its degree of truth, as long as this sign is strictly negative of positive:

LEMMA 3.4: *For every monotonic constraint ϕ in C, for all $(a_1, \ldots, a_n)$, if $\tau(\phi)(a_1, \ldots, a_{|V|}) > 0$ then $(a_1, \ldots, a_{|V|}) \in [\![\phi]\!]$, and if $\tau(\phi)(a_1, \ldots, a_{|V|}) < 0$ then $(a_1, \ldots, a_{|V|}) \notin [\![\phi]\!]$.*

It is a well-known fact in computable analysis [Pour-El and Richards, 1989, Weihrauch, 2000], that the minimum and maximum operations are stable (using any common pseudo-metric), and so we have by induction:

LEMMA 3.5: *For every monotonic constraint ϕ, τ is stable for ϕ.*

For monotonic constraints we can easily influence the degree of truth:

LEMMA 3.6: *For every monotonic constraint ϕ and result ϕ' of a δ-perturbation $(\delta, \ldots, \delta)$ of ϕ, $\tau(\phi') = \tau(\phi) - \delta$.*

Proof: After moving all the constants representing δ on the right-hand side of atomic sub-constraints to the according left-hand sides, the degree of truth of all atomic sub-constraints is lowered by δ and by induction also the degree of truth of ϕ is lowered by δ, that is $\tau(\phi') = \tau(\phi) - \delta$.

□

Since the sign of a real number is stable only for non-zero inputs, quantified constraints are stable under perturbation up to sign computation:

THEOREM 3.1: *A monotonic constraint ϕ in C is stable iff*

$$\mathrm{Vol}\{(a_1, \ldots, a_{|V|}) \in R_V | \tau(\phi)(a_1, \ldots, a_{|V|}) = 0\} = 0$$

Proof: $\Rightarrow$: Assume that $\mathrm{Vol}\{(a_1,\ldots,a_{|V|}) \in R_V | \tau(\phi)(a_1,\ldots,a_{|V|}) = 0\} > 0$. By Lemma 3.6, for every $\delta > 0$ there is a result ϕ' of δ-perturbation of ϕ such that for all $(a_1,\ldots,a_{|V|})$ such that $\tau(\phi)(a_1,\ldots,a_{|V|}) = 0$, $\tau(\phi')(a_1,\ldots,a_{|V|})$ is negative. In a similar way we can make these elements positive. So, by Lemma 3.4, the volume of the $(a_1,\ldots,a_{|V|})$ for which the property of being in $[\![\phi]\!]$ changes under small perturbations is greater than zero, and ϕ is unstable.

$\Leftarrow$: Assume that $\mathrm{Vol}\{(a_1,\ldots,a_{|V|}) \in R_V | \tau(\phi)(a_1,\ldots,a_{|V|}) = 0\} = 0$. Since τ is stable and the test for positivity and negativity of a real number is stable for non-zero inputs, by Lemma 3.4, the result ϕ' of a small perturbation of ϕ can only be different at $(a_1,\ldots,a_{|V|})$ for which $\tau(\phi)(a_1,\ldots,a_{|V|}) = 0$. The volume of these $(a_1,\ldots,a_{|V|})$ is zero and so $d(\phi,\phi') = \mathrm{Vol}([\![\phi]\!] \nabla [\![\phi']\!])$ tends to zero under small perturbations and ϕ is stable.

□

COROLLARY 3.1: *A closed monotonic constraint ϕ is stable iff $\tau(\phi)(a_1,\ldots,a_{|V|}) \neq 0$, for arbitrary $(a_1,\ldots,a_{|V|}) \in R_V$.*

4. Stability Conditions

In this section we provide two theorems that provide conditions when a given constraint is stable. We need some additional technical tools: We say that two predicate symbols p_1 and p_2 are *equal up to an infinitesimal predicate change* iff (p_1,p_2) is in $\{(<,\leq),(\leq,<),(>,\geq),(\geq,>)\}$. A constraint ϕ is *equal to a constraint ϕ' up to infinitesimal predicate changes* iff ϕ and ϕ' are syntactically equal, except that the according predicate symbols of ϕ and ϕ' are equal up to an infinitesimal predicate change.

Definition: A constraint ϕ in C is *stable under infinitesimal predicate changes* iff for all ϕ' that are equal to ϕ up to infinitesimal predicate changes, the volume of $d([\![\phi]\!],[\![\phi']\!])$ is zero.

THEOREM 4.1: *A monotonic constraint ϕ in C is stable under infinitesimal predicate changes iff* $\mathrm{Vol}\{(a_1,\ldots,a_{|V|}) \in R_V | \tau(\phi)(a_1,\ldots,a_{|V|}) = 0\} = 0$.

Proof: We prove that for an arbitrary but fixed $(a_1,\ldots,a_{|V|})$, the fact that $(a_1,\ldots,a_{|V|})$ is in $[\![\phi]\!]$, changes under infinitesimal predicate changes if and only if $\tau(\phi)(a_1,\ldots,a_{|V|}) = 0$. For this replace all inequalities in ϕ by $\geq$. By Lemma 3.2 this means that $(a_1,\ldots,a_{|V|}) \in [\![\phi]\!]$. But if we replace all inequalities by $>$, by Lemma 3.3 this means that $(a_1,\ldots,a_{|V|}) \notin [\![\phi]\!]$. So $(a_1,\ldots,a_{|V|}) \in [\![\phi]\!]$ is not stable under infinitesimal predicate changes. □

COROLLARY 4.1: *A monotonic constraint is stable iff it is stable under infinitesimal predicate changes.*

Now we prove a stability condition that uses volume quantifiers [Keisler, 1985, Bacchus, 1990b,a, Mostowski, 1957, Väänänen, 1997, Ratschan, 2000b]. For this we assign semantics also to constraints containing quantifiers that are annotated with an additional non-negative real number v. Here $(a_1, \ldots, a_{|V|})$ is in the solution set of $[\![\exists_{\mathtt{v}} \mathtt{x}\ \phi]\!]$ if and only if the volume of the $u \in R_x$ such that $(a_1, \ldots, a_{k-1}, u, a_{k+1}, \ldots, a_n)$ is in the solution set of ϕ is greater than v. In a similar way $(a_1, \ldots, a_{|V|})$ is in the solution set of $[\![\forall_{\mathtt{v}} \mathtt{x}\ \phi]\!]$ iff the volume of the $u \in R_x$ such that $(a_1, \ldots, a_{k-1}, u, a_{k+1}, \ldots, a_n)$ is not in the solution set of ϕ is less or equal v.

Now let us denote by $Z(\phi)$ the constraint that is obtained from ϕ by replacing all quantifiers by volume quantifiers whose annotation is zero.

THEOREM 4.2: *For every monotonic constraint ϕ in C, for every $(a_1, \ldots, a_{|V|})$ for which $\tau(\phi)(a_1, \ldots, a_{|V|})$ is non-zero, $(a_1, \ldots, a_{|V|}) \in [\![\phi]\!]$ iff $(a_1, \ldots, a_{|V|}) \in [\![Z(\phi)]\!]$.*

Proof: W.l.o.g. we assume that ϕ has the form $(\mathtt{Q}_1\mathtt{x}_1)\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi'$, where ϕ' is quantifier-free. We proceed by induction over the quantifiers $\mathtt{Q}_\mathtt{k}, \mathtt{Q}_{\mathtt{k}-1}, \ldots, \mathtt{Q}_1$. For the base case of a constraint without quantifiers there is nothing to prove.

Now assume that for all $(a_1, \ldots, a_{|V|})$ such that

$$\tau((\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')(a_1, \ldots, a_{|V|})$$

is non-zero,

$$(a_1, \ldots, a_{|V|}) \in [\![(\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi']\!] \text{ iff } (a_1, \ldots, a_{|V|}) \in [\![Z((\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')]\!]$$

We prove that for all $(a_1, \ldots, a_{|V|})$ such that

$$\tau((\mathtt{Q}_\mathtt{i}\mathtt{x}_\mathtt{i})(\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')(a_1, \ldots, a_{|V|})$$

is non-zero,

$$(a_1, \ldots, a_{|V|}) \in [\![(\mathtt{Q}_\mathtt{i}\mathtt{x}_\mathtt{i})(\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi']\!] \text{ iff } (a_1, \ldots, a_{|V|}) \in [\![Z((\mathtt{Q}_\mathtt{i}\mathtt{x}_\mathtt{i})(\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')]\!]$$

$\mathtt{Q}_\mathtt{i} = \exists$**:** Let $(a_1, \ldots, a_{|V|})$ be arbitrary but fixed.
If $\tau((\mathtt{Q}_\mathtt{i}\mathtt{x}_\mathtt{i})(\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')(a_1, \ldots, a_{|V|}) > 0$, then by definition of τ and compactness of $R_{\mathtt{x}_\mathtt{i}}$ there is a $u \in R_{\mathtt{x}_\mathtt{i}}$ such that

$$\tau((\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')(a_1, \ldots, a_{k-1}, u, a_{k+1}, a_{|V|}) > 0,$$

where $\mathtt{x}_\mathtt{i}$ is the k-th variable in V. By continuity, u has a neighborhood of non-zero volume such that for all u' in this neighborhood

$$\tau((\mathtt{Q}_{\mathtt{i}+1}\mathtt{x}_{\mathtt{i}+1})\ldots(\mathtt{Q}_\mathtt{k}\mathtt{x}_\mathtt{k})\phi')(a_1, \ldots, a_{k-1}, u', a_{k+1}, a_{|V|}) > 0,$$

and by the induction hypothesis

$$(a_1, \ldots, a_{k-1}, u', a_{k+1}, \ldots, a_{|V|}) \in [\![Z((\mathtt{Q_{i+1}x_{i+1}}) \ldots (Q_k x_k)\phi')]\!].$$

So by the definition of $\exists_0$,

$$(a_1, \ldots, a_{|V|}) \in [\![Z((\mathtt{Q_i x_i})(\mathtt{Q_{i+1}x_{i+1}}) \ldots (\mathtt{Q_k x_k})\phi')]\!].$$

If $\tau((\mathtt{Q_i x_i})(\mathtt{Q_{i+1}x_{i+1}}) \ldots (\mathtt{Q_k x_k})\phi')(a_1, \ldots, a_{|V|}) < 0$, then by definition of τ, for all $u \in R_{\mathtt{x_i}}$,

$$\tau((\mathtt{Q_{i+1}x_{i+1}}) \ldots (\mathtt{Q_k x_k})\phi')(a_1, \ldots, a_{k-1}, u, a_{k+1}, \ldots, a_{|V|}) < 0,$$

where $\mathtt{x_i}$ is the k-th variable in V. By the induction hypothesis, for all $u \in R_{\mathtt{x_i}}$,

$$(a_1, \ldots, a_{k-1}, u, a_{k+1}, \ldots, a_{|V|}) \notin [\![Z((\mathtt{Q_{i+1}x_{i+1}}) \ldots (\mathtt{Q_k x_k})\phi')]\!].$$

So by the definition of $\exists_0$,

$$(a_1, \ldots, a_{|V|}) \notin [\![Z((\mathtt{Q_i x_i})(\mathtt{Q_{i+1}x_{i+1}}) \ldots (\mathtt{Q_k x_k})\phi')]\!].$$

$\mathtt{Q_i} = \forall$**:** Dual

□

COROLLARY 4.2: *For every stable monotonic constraint ϕ in C, the volume of the $(a_1, \ldots, a_{|V|})$ for which $(a_1, \ldots, a_{|V|}) \notin [\![\phi]\!]$ iff $(a_1, \ldots, a_{|V|}) \in [\![Z(\phi)]\!]$ is zero.*

COROLLARY 4.3: *For every stable closed monotonic constraint ϕ, $[\![\phi]\!] = [\![Z(\phi)]\!]$.*

This means that if $[\![\phi]\!] \neq [\![Z(\phi)]\!]$ then we know that ϕ is unstable. However the other direction does not hold. For example the constraint $\exists \mathtt{x}\ \mathtt{0} \geq \mathtt{0}$ is unstable although $[\![\exists \mathtt{x}\ \mathtt{0} \geq \mathtt{0}]\!] = [\![\exists_0 \mathtt{x}\ \mathtt{0} \geq \mathtt{0}]\!]$.

5. Different Forms of Stability

In this section we consider the following more general forms of perturbations:

- Perturbations of all the constants occurring in constraints (instead of only the zeros occurring on the right-hand side of atomic sub-constraints).
- Perturbations of projection functions denoted by the variables in a constraint.

Here there are several possibilities for choosing the perturbations of the projection functions:

- Since the same variable can occur several times in a constraint, we can perturb different occurrences of the same variable either by the same or by different perturbations (the second case occurs when using interval methods to represent projection functions because interval arithmetic does not keep information about the identity of variables due to the so-called dependency problem [Neumaier, 1990]).
- We can use continuous or discontinuous perturbations.

Observe that each type of perturbation results in a *perturbation function* $P : C \times \mathbb{R} \to 2^C$ such that $P(\phi, \delta)$ is the set of all results of δ-perturbations of ϕ. Each perturbation function P results in an according notion of stability using P.

Let us denote by P_0 the perturbation function used up to this section—it perturbs only the zeros on the right-hand sides of atomic sub-constraints. Using the additional types of perturbations described above we get the new perturbation functions $P_c, P_{c,v/dep/cont}, P_{c,v/dep/disc}, P_{c,v/indep/cont}, P_{c,v/indep/disc}$. For every constraint ϕ and positive real number δ we have the subset relation depicted in Figure 1 (where $\to$ corresponds to $\supseteq$).

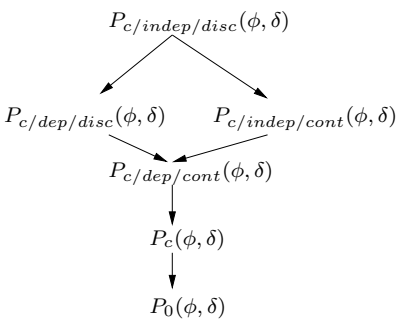

Figure 1: Perturbation Functions

LEMMA 5.1: *Let P_1, P_2 be perturbation functions. If for all $\delta \in \mathbb{R}$, $P_1(\phi, \delta) \supseteq P_2(\phi, \delta)$ then stability of ϕ using P_1 implies stability of ϕ using P_2.*

So the arrows in Figure 1 also are valid implications wrt. stability. But also the other direction is valid. For showing this, observe that the degree of truth is stable under all perturbation functions under consideration. So we can adapt the right-to-left direction of the proof of Theorem 3.1 to assure that if a monotonic constraint is stable then it is also stable under general perturbations. Thus all perturbation functions result in equivalent notions of stability for monotonic constraints—using additional information about continuity of functions and equality of variables does not result in a large set of stable constraints. The situation would change when considering variable perturbations without constant perturbation—but we do not discuss this case further in this paper.

6. Stability of General Constraints

Now we extend our considerations to constraints that are not monotonic. For this recall that one can view a quantified constraint as a mapping that takes the meaning of its constant symbols (i.e., real numbers), and its variables (i.e., projection functions) and produces the according meaning of the constraint (i.e., solution set). This mapping can be different for two equivalent constraints. On the other hand, if this mapping is the same for two constraints ϕ_1 and ϕ_2 then ϕ_1 and ϕ_2 are equivalent, and furthermore ϕ_1 is stable iff ϕ_2 is stable.

Now observe that for any constraint ϕ containing only the predicate symbols $<, \leq, >, \geq$ this mapping is the same for ϕ and its monotonic transformation $T(\phi)$. So we simply can extend the definition of degree of truth to such constraints and keep all the stability results in this paper.

Now consider constraints that contain equalities. For these the above mapping is changed by the monotonic transformation. But we can simulate every perturbation of such a general constraint on its monotonic transformation by assigning the same perturbation to the two variables/constants generated by the transformation of an equality. So for any perturbation function, there is an according subset that takes into account that certain variables/constants are the same, and, by Lemma 5.1, we get:

LEMMA 6.1: *For every constraint ϕ, if $T(\phi)$ is stable wrt. a perturbation function P then also ϕ is stable wrt. P.*

For the other direction we study how one can influence the result of a monotonic transformation of an equality by discontinuous transformations:

LEMMA 6.2: *For every constraint ϕ and $(a_1, \ldots, a_{|V|})$ s.t. $\tau(T(\phi))(a_1, \ldots, a_{|V|}) = 0$, for every $\delta > 0$ there is a result ϕ' of a δ-perturbation of ϕ using $P_{c,v/dep/disc}$ such that $\tau(T(\phi'))(a_1, \ldots, a_{|V|})$ is negative.*

Proof: We perturb all atomic sub-constraints of $T(\phi)$ that result from inequalities in ϕ according to Lemma 3.6.

Now consider a sub-constraint of $T(\phi)$ resulting from an equality $\theta = 0$ in ϕ. Its degree of truth is less or equal zero everywhere. Observe that for every perturbation of $\theta = 0$ there is an according perturbation of $T(\theta = 0)$. Furthermore for every constraint of the form $\theta = 0$ such that $\tau(\theta = 0)$ has no zero, the resulting $\tau(T(\theta = 0))$ is negative. So we will perturb $\theta = 0$ in such a way that the degree of truth of the result has no zeros.

First let c be a constant that is less or equal δ, such that $\tau(\theta = c)$ does not have infinitely many zeros. Now it is easy to find a discontinuous δ-perturbation of the variables in $\theta = c$ such that the result always jumps over the zero line, and thus has no zeros.

□

COROLLARY 6.1: *A constraint ϕ is stable using $P_{c,v/dep/disc}$ iff its transformation $T(\phi)$ is stable.*

For example a constraint of the form $\exists x\ \theta = 0$ is stable using $P_{c,v/dep/disc}$ iff $[\![\theta]\!]$ has no zero. Obviously the above holds also for $P_{c,v/indep/disc}$, but in general not for continuous perturbations.

7. Conclusion

In this paper we have studied the behavior of quantified constraints under perturbation. We have shown that the solution set of a quantified constraint is stable under perturbation except for one (unstable) sign computation; we have provided two practically useful conditions for stability; and we have shown that, for constraints containing only inequality predicates, stability under perturbation is independent of whether we use information about the continuity of constraints and equality of variables.

One can now apply these results to the behavior of algorithms that involve approximation. For algorithms that use validated numerical computation (e.g., interval arithmetic) we now know, for which types of inputs they should terminate. For algorithms that use floating-point arithmetic with rounding, we know for which types of inputs, one cannot expect correct answers. The results of this paper provide important tools for proving that concrete algorithms follow this expected behavior [Ratschan, 2001b].

Since the model used in this paper (and in any kind of numerical error analysis) is an abstraction, in rare cases a concrete method might even succeed to compute the solution set of an unstable constraint. For the author's method [Ratschan, 2001b, 2002, 2000a] this can happen when certain floating point operations can be done exactly without rounding errors.

What can be done if one encounters an unstable quantified constraint? One possibility would be to compute the solution of a similar, but stable problem, that is, to regularize the problem. This idea has traditionally been applied to certain ill-posed problems in numerical mathematics [Engl, 1993], but is recently frequently used when applying numerical methods to discontinuous algebraic problems [Bodnár et al., 2000, Huang et al., 2000, Schönhage, 1985]. In our case, the results of Section 4 give important hints on the consequences of such a regularization. For example, the counter-example $\exists \mathtt{x}\ \mathtt{0} \geq \mathtt{0}$ to the converse of Theorem 4.2 is one of the few instances of constraints which approximate quantifiers [Ratschan, 2000b] cannot regularize.

The degree of truth defined in this paper does not only provide a criterion for termination. One could also interpret it as a condition number—its size is a measure for the distance of an input constraint to an ill-posed input. For validated numerical methods the inverse of the condition number usually is a measure for the run-time, and for algorithms using floating-point arithmetic it

can act as a measure for the accuracy of the result. Further work will study the details.

A further observation is, that for certain functions (e.g., low degree polynomials) there are explicit formulae for the minimum and maximum. Thus it might be interesting to explicitly compute the degree of truth by substituting terms provided by such formulae, in a similar way as the method of quantifier elimination by substitution of parametrized test points [Loos and Weispfenning, 1993, Weispfenning, 1997].

The results in this paper are not applicable for checking continuous stability of constraints that contain equalities. This is also further work.

The author thanks Hoon Hong for the initial motivation for studying approximate quantified constraint solving, Richard Liska for interesting example problems that gave the impetus to writing this paper, and the anonymous referees for helpful comments.

References

Fahiem Bacchus. Lp, a logic for representing and reasoning with statistical knowledge. *Computational Intelligence*, 6:209–231, 1990a.

Fahiem Bacchus. *Representing and Reasoning with Probabilistic Knowledge*. MIT Press, 1990b.

Frederic Benhamou and Frederic 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.

Gábor Bodnár, Petru Pau, and Josef Schicho. Exact real computation in computer algebra. Technical Report 00-33, RISC-Linz, 2000.

B. F. Caviness and J. R. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition*. Springer, 1998.

George 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 *Lecture Notes in Computer Science*, pages 134–183. Springer Verlag, 1975. Also in Caviness and Johnson [1998].

George E. Collins and Hoon Hong. Partial cylindrical algebraic decomposition for quantifier elimination. *Journal of Symbolic Computation*, 12:299–328, 1991. Also in Caviness and Johnson [1998].

P. Dorato, Wei Yang, and C. Abdallah. Robust multi-objective feedback design by quantifier elimination. *Journal of Symbolic Computation*, 24:153–159, 1997.

Peter Dorato. Quantified multivariate polynomial inequalities. *IEEE Control Systems Magazine*, pages 48–58, October 2000.

H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic*. Springer Verlag, 1984.

Heinz W. Engl. Regularization methods for the stable solution of inverse problems. *Surveys on Mathematics for Industry*, 3:71–143, 1993.

Jürgen Garloff and Birgit Graf. Solving strict polynomial inequalities by Bernstein expansion. In N. Munro, editor, *The Use of Symbolic Methods in Control System Analysis and Design*, pages 339–352. The Institution of Electr. Eng. (IEE), 1999.

Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 1996.

Hoon Hong, Richard Liska, and Stanly Steinberg. Testing stability by quantifier elimination. *Journal of Symbolic Computation*, 24(2):161–187, 1997.

Y. Huang, H. J. Stetter, W. Wu, and L. Zhi. Pseudofactors of multivariate polynomials. In *Proc. ISSAC'00*, pages 160–167. ACM Press, 2000.

Nikolaos I. Ioakimidis. Quantifier elimination in applied mechanics problems with cylindrical algebraic decomposition. *International Journal of Solids and Structures*, 34(30):4037–4070, 1997.

Luc Jaulin and Éric Walter. Guaranteed tuning, with application to robust control and motion planning. *Automatica*, 32(8):1217–1221, 1996.

Mats Jirstrand. Nonlinear control system design by quantifier elimination. *Journal of Symbolic Computation*, 24(2):137–152, 1997.

H. J. Keisler. Probability quantifiers. In J. Barwise and S. Feferman, editors, *Model Theoretic Logics*, chapter XIV. Springer Verlag, New York, 1985.

Richard Liska and Stanly Steinberg. Applying quantifier elimination to stability analysis of difference schemes. *Computer Journal*, 36(5):497–593, 1993.

Rüdiger Loos and Volker Weispfenning. Applying linear quantifier elimination. *The Computer Journal*, 36(5):450–462, 1993.

S. Malan, M. Milanese, and M. Taragna. Robust analysis and design of control systems using interval arithmetic. *Automatica*, 33(7):1363–1372, 1997.

A. Mostowski. On a generalization of quantifiers. *Fundamenta Mathematicae*, 44:12–36, 1957.

Arnold Neumaier. *Interval Methods for Systems of Equations*. Cambridge Univ. Press, Cambridge, 1990.

Marian B. Pour-El and Jonathan I. Richards. *Computability in Analysis and Physics*. Springer-Verlag, 1989.

Stefan Ratschan. Approximate quantified constraint solving (AQCS). `http://www.risc.uni-linz.ac.at/research/software/AQCS`, 2000a. Software package.

Stefan Ratschan. Convergence of quantified constraint solving by approximate quantifiers. Technical Report 00-23, Research Institute for Symbolic Computation (RISC) - Linz, 2000b. Submitted for Publication.

Stefan Ratschan. Uncertainty propagation in heterogeneous algebras for approximate quantified constraint solving. *Journal of Universal Computer Science*, 6(9), 2000c.

Stefan Ratschan. Applications of real first-order constraint solving — bibliography. http://www.risc.uni-linz.ac.at/people/sratscha/appFOC.html, 2001a.

Stefan Ratschan. Continuous first-order constraint satisfaction. `http://www.risc.uni-linz.ac.at/people/sratscha`, 2001b. Submitted for Publication.

Stefan Ratschan. Approximate quantified constraint solving by cylindrical box decomposition. *Reliable Computing*, 8(1):21–42, 2002.

N. Rescher. *Many-valued Logic*. McGraw-Hill, New York, 1969.

Arnold Schönhage. Quasi-GCD computations. *J. Complexity 1*, pages 118–137, 1985.

Sergey P. Shary. Algebraic approach to the interval linear static identification, tolerance, and control problems, or one more application of Kaucher arithmetic. *Reliable Computing*, 2(1):3–33, 1996a.

Sergey P. Shary. A new approach to the analysis of static systems under interval uncertainty. In Götz Alefeld, Andreas Frommer, and Bruno Lang, editors, *Scientific Computing and Validated Numerics — SCAN'95*, volume 90 of *Mathematical Research*, pages 118–132. Akademie Verlag, 1996b.

Sergey P. Shary. Outer estimation of generalized solution sets to interval linear systems. *Reliable Computing*, 5:323–335, 1999.

Thomas Sturm. Reasoning over networks by symbolic methods. *Applicable Algebra in Engineering Communication and Computing*, 10(1):79–96, September 1999.

Alfred Tarski. *A Decision Method for Elementary Algebra and Geometry*. Univ. of California Press, Berkeley, 1951. Also in Caviness and Johnson [1998].

Alasdair Urquhart. Many-valued logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic, Vol. III: Alternatives in Classical Logic*, chapter III.2, pages 71–116. D. Reidel Publishing Company, 1986.

Jouko Väänänen. Generalized quantifiers. *Bulletin of the European Association for Theoretical Computer Science*, June 1997.

Klaus Weihrauch. *Introduction to Computable Analysis*. Texts in Theoretical Computer Science. Springer Verlag, 2000.

Volker Weispfenning. Quantifier elimination for real algebra - the quadratic case and beyond. *Applicable Algebra in Engineering, Communication and Computing*, 8(2):85–101, 1997.

J. H. Wilkinson. *Rounding Errors in Algebraic Processes*. Prentice Hall, 1964.