Technical Report

Quantifier Elimination for Trigonometric Polynomials by Cylindrical Trigonometric Decomposition

Petru Pau and Josef Schicho

Research Institute for Symbolic Computation

Johannes Kepler University, A-4040 Linz, Austria

e-mail: {ppau,jschicho}@risc.uni-linz.ac.at

www: http://www.risc.uni-linz/people/{ppau,jschicho}

(Received 15 March 1999)

Given a formula Φ in r variables, some of them quantified and/or occurring as arguments in trigonometric functions, we consider in this paper the problem of finding a quantifier-free formula equivalent to Φ.

We present an algorithm that computes first a decomposition of the space so that the polynomials occurring in Φ are sign-invariant over each cell of this decomposition. Then the cells over which Φ is true are collected: their description gives a quantifier-free formula equivalent to Φ. The algorithm is an adaption of a well-known algorithm for the same problem with polynomial functions.

1. Introduction

The quantifier elimination (shortly QE) problem is the following: *Given a formula with polynomial equations and inequalities and quantifiers, construct an equivalent formula (i.e. a formula with the same solution set over* $\mathbf{R}$*) without quantifiers.* It has been observed that a broad range of problems in mathematics, computer science, engineering and industrial computations can be reduced to the QE problem (see (Loos and Weispfenning, 1993; González-Vega, 1996; Hong, 1996; Weispfenning, 1997; Dorato *et al.*, 1997; Hong, 1997)).

The first algorithm which is practically useful has been given in (Collins, 1975), with improvements described in (Arnon *et al.*, 1988; Arnon and McCallum, 1982; Collins, 1994; Collins and Hong, 1991; Collins and Johnson, 1989; McCallum, 1988). Other algorithmic approaches are described in (Renegar, 1992; Weispfenning, 1998; Dolzmann and Sturm, 1997; González-Vega, 1998).

Many applications, especially in mechanical engineering and in numerical analysis (see (Liska and Steinberg, 1993; Gónzalez-López and Recio, 1993; Jirstrand, 1997)) lead to QE problems with trigonometric functions involved. The general first order theory of the real numbers with sine and cosine is undecidable: since the zeroes of the sine function are precisely the integral multiples of π, one can find diophantine arithmetic within it, which is undecidable (see also (Tarski, 1998)). Fortunately, in any reasonable applications it is possible to reduce the trigonometric problem to a strictly algebraic problem (see also (Gutierrez and Recio, 1998; Hong and Schicho, 1998)). Unfortunately, a straightforward reduction is only possible at the cost of introducing additional variables, which has severe consequences for the computational effort.

Here, we propose another approach: adapting Collins's CAD method to the trigonometric case. The method consists in two phases, *the decomposition phase* and *the evaluation phase*. First, the definition space of the polynomial/trigonometric functions is decomposed cylindrically into cells. On each cell the functions have constant sign. Then one collects the cells on which the initial formula is true. Their definition is reported as an equivalent quantifier-free formula.

During the decomposition process, which is carried out recursively, at each step, a variable is dropped by means of a special *projection* operator. Once a decomposition of a lower dimensional space is computed, its cells are *lifted* to a decomposition of the total space.

We try to use in this algorithm modules from Collins's CAD. For this purpose, at each step of decomposition, if the variable to be eliminated is an angle variable – i.e., it occurs as argument to sines and cosines – we transform the set of polynomials by means of the tangent of the half angle, and then we apply the normal decomposition procedure.

The decomposition computed using these algebraic polynomials gives an incomplete decomposition with respect to the initial variables, because, when using the tangent of the half angle, one must avoid π: here the tangent does not exist. Our method takes these values into account by using a lifting method which leads to a cylindrical decomposition of the whole space.

The decomposition thus generated would not be algebraic anymore, since the cells are computed as sets whose boundaries are roots of trigonometric polynomials: we will call it *cylindrical trigonometric decomposition* The cylindricity property ensures the validity of an algorithm that evaluates the formula in each cell, reporting the collection of cells over which it is true. The final output will be a quantifier-free formula (involving the initial trigonometric polynomials and some other relations) equivalent to the initial one.

The naive approach for solving this problem would be by substituting new variables for each sine and cosine, and adding formulas coresponding to identities of the form $\sin(x)^2 + \cos(x)^2 = 1$. We will show that this method is prohibitively expensive with respect to the number of operations.

Another approach – not so naive but still insufficient – would be to transform the angles by means of the tangent of their halfs and apply logic to reduce to a finite set of polynomial QE problems, and to use Collins's algorithm (or any other of the above) as a black box. Unfortunately, all CAD-based algorithms assume prenex normal form, and in order to establish this, one needs to introduce additional variables in general. In the worst case, the number of variables may double, hence the situation is as bad as with the naive approach.

The paper is organized as follows: Section 2 contains a description of the problem, and some theoretical preliminaries. In Section 3 we present the idea of the algorithm and give the necessary theoretical facts that sustain the method. Section 4 contains a detailed description of the algorithm. Section 5 contains its complexity analysis.

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

2. Theoretical Preliminaries

We consider here a logical formula in r variables, $x_1, \ldots, x_r$, of the form:

$$\Phi := \Theta_{f+1} x_{f+1} \ldots \Theta_r x_r R(x_1, \ldots, x_r).$$

We write Φ in functional notation only when it is necessary to show its arguments.

Each of the Θ's is either the existential or the universal quantifier.

R is a quantifier-free formula that consists of a set of relations $r_1 \ldots, r_m$ connected with logical connectives (and, or), each relations being of the form:

$$r_i(x_1, \ldots, x_r) ::= p_i(x_1, \ldots, x_r)\, \texttt{rel}\, 0 \text{ for } i = 1..m,$$

where `rel` can be $<, >, =, \neq, \leq, \geq$.

Each p_i is a trigonometric polynomial, i.e., an expression defined using elementary functions: *polynomials, sine, cosine*. We will denote by F the set of polynnomials occuring in R.

DEFINITION 2.1. *Let $r > 0, d > 0$, $d \leq r$ be natural numbers, $V := \{x_1, \ldots, x_r\}$ a set of variables, $1 \leq i_1 \leq \ldots \leq i_d \leq r$ integral numbers, and $A := \{x_{i_1}, \ldots, x_{i_d}\} \subseteq V$. The set of all algebraic polynomials in r variables $x_1, \ldots, x_r$, with angle variables $x_{i_1}, \ldots, x_{i_d}$ and coefficients in $\mathbf{K}$, $\mathbf{K}$ being one of $\mathbf{Z}$, $\mathbf{Q}$, or $\mathbf{R}$, denoted $Trig_{\mathbf{K}[x_{j_1}, \ldots, x_{j_{r-d}}]}[x_{i_1}, \ldots, x_{i_d}]$, where $1 \leq j_1 \leq \ldots \leq j_{r-d} \leq r$ and $\{x_{j_1}, \ldots, x_{j_{r-d}}\} = V - A$, is defined as:*

$$\begin{aligned} Trig_{\mathbf{K}[x_{j_1}, \ldots, x_{j_{r-d}}]}[x_{i_1}, \ldots, x_{i_d}] \quad := \quad & \Big\{ \sum_{k=1}^{n} a_k t_k \,| \\ & a_k \in \mathbf{K}[x_{j_1}, \ldots, x_{j_{r-d}}], \\ & t_k = \prod_{l=1}^{d} \sin(x_{i_l})^{e_{s_l}} \cos(x_{i_l})^{e_{c_l}}, \\ & n, e_{s_l}, e_{c_l} \in \mathbf{N}, n > 0 \Big\} \end{aligned}$$

Note that this definition is very restrictive: the angle variables are not intermixed with the algebraic variables. Moreover, the arguments of sines/cosines are variables. We believe that this is the most frequent form of such problems. In fact, linear expressions in the angle variables might be allowed as arguments of trigonometric functions, since there are identities that can be used to bring them to the form imposed by this definition.

PROBLEM 2.1. *Determine a quantifier free formula Ψ such that*

$$\Psi \Leftrightarrow \Phi.$$

For formulas involving only algebraic polynomials there exists the celebrated Collins's *Cylindrical Algebraic Decomposition* (CAD), as a main tool for QE. This algorithm works in two major steps:

1 First, it decomposes the space $\mathbf{R}^r$ into cells over which the polynomials in F are sign-invariant. This decomposition is constructed in two phases:

(a) The *projection* phase – in which a set of polynomials in one-dimension less is computed from F, so that, once a sign-invariant decomposition of $\mathbf{R}^{r-1}$ is computed, the roots of the polynomials in F are *delineable* over each cell of this decomposition.

(b) The *lifting* phase – from the sign-invariant decomposition of $\mathbf{R}^{r-1}$, using the real roots of the polynomials in F, a sign-invariant decomposition of $\mathbf{R}^r$ is computed.

The whole process of computing the decomposition is carried out recursively: At the k'th level of recursion, the k'th variable is eliminated by computing the projection of the polynomials. The decomposition of $\mathbf{R}^{k-1}$ is computed in a recursive call, and this decomposition is lifted by constructing the roots of the polynomials in k variables over each cell.
Associated to each cell, a sample point is also issued. This is used both for lifting a $k-1$-dimensional decomposition to a k dimensional one, and for evaluating the sign of polynomials in each cell.
In fact, although the decomposition process is carried out from r downto 1, the effective description of cells stops at dimension f, hence only a description of the definition of the space of the free variables is produced.

2 Once a sign-invariant decomposition is available, an *evaluation* process is initiated, which computes the truth value of the set of polynomials over each cell and keeps only those cells evaluated to true. Eventually, a set of cells in f dimensions is computed. The description of these cells is a quantifier-free formula that is equivalent to Φ.

If Φ contained only algebraic polynomials, Collins's algorithm would compute a CAD, expressed as a closed formula Ψ, in time:

$$O((2n)^{2^{2r+8}} m^{2^{r+6}} norm_{max}^3 n_{at-f}),$$

where r is the number of variables, m is the number of polynomials, n is the maximum degree of any polynomial, $norm_{max}$ is the maximum of the norms of these polynomials, and n_{at-f} is the number of occurrences of atomic formulas in Φ.

Let now $F \subset Trig_{\mathbf{K}[x_{j_1},\ldots,x_{j_{r-d}}]}[x_{i_1},\ldots,x_{i_d}]$. Since the trigonometric functions are periodical, for $1 \le k \le d$, the i_k'th variable can be safely considered only in the interval $(-\pi,\pi]$. An analogous of Collins's algorithm for trigonometric polynomials should compute first a cylindrical decomposition of the space $\mathcal{I} := I_1 \times \ldots \times I_r$, where:

$$I_i := \begin{cases} \mathbf{R}, & \text{if } x_i \notin A, \\ (-\pi,\pi], & \text{if } x_i \in A, \end{cases}$$

for $i = 1, 2, \ldots, r$. This decomposition will not be algebraic anymore, as the cells are defined by roots of trigonometric polynomials.

We will also use the notation $\mathcal{I}^k := I_1 \times \ldots \times I_k$, for $1 \le k \le r$.

The following definition is the trigonometric version of that given in (Collins, 1975):

Definition 2.2. **(Delineability)** *Let $f(x_1,\ldots,x_r)$ a trigonometric polynomial, $r \ge 2$, x_r an algebraic variable, and S a subset of $\mathcal{I}^{r-1}$. We say that the roots of f are delineable on S and $f_1,\ldots,f_k$, $k \ge 0$* delineate *the real roots of f, if the following conditions are satisfied:*

1 *There are* $m \geq 0$ *positive integers* e_i *such that, if* $(a_1, \ldots, a_{r-1}) \in S$*, then* $f(a_1, \ldots, a_{r-1}, x)$ *has exactly* m *distinct roots, with multiplicities* $e_1, \ldots, e_m$.
2 $f_1 < \ldots < f_k$ *are continuous from* S *to* I_r.
3 *If* $(a_1, \ldots, a_{r-1}) \in S$*, then* $f_i(a_1, \ldots, a_{r-1})$ *is a root of* $f(a_1, \ldots, a_{r-1}, x)$ *with multiplicity* e_i*, for* $1 \leq i \leq k$.
4 *If* $(a_1, \ldots, a_{r-1}) \in S$*,* $b \in I_r$ *and* $f(a_1, \ldots, a_{r-1}, b) = 0$*, then there is an* i*,* $1 \leq i \leq k$*, such that* $b = f_i(a_1, \ldots, a_{r-1})$.

If the r'th variable is an angle variable, the functions f_i introduced above will describe the laws of variation for the (real) roots of some trigonometric polynomials over S.

Definition 2.3. **(Trigonometric Delineability)** *Let* $f(x_1, \ldots, x_r)$ *a trigonometric polynomial,* $r \geq 2$*, and* S *a subset of* $\mathcal{I}^{r-1}$*. We say that the roots of* f *are delineable on* S *and* $g_1, \ldots, g_k$*,* $k \geq 0$ delineate *the real roots of* f*, if the following conditions are satisfied:*

1 *There are* $m \geq 0$ *positive integers* e_i *such that if* $(a_1, \ldots, a_{r-1}) \in S$*, then* $f(a_1, \ldots, a_{r-1}, x)$ *has exactly* m *distinct roots, with multiplicities* $e_1, \ldots, e_m$ *(here the multiplicity is seen in the analytical sense).*
2 $-\pi < g_1 < \ldots < g_k = \pi$ *are continuous from* S *to* I_r.
3 *If* $(a_1, \ldots, a_{r-1}) \in S$*, then* $g_i(a_1, \ldots, a_{r-1})$ *is a root of* $f(a_1, \ldots, a_{r-1}, x)$ *of multiplicity* e_i*, for* $1 \leq i \leq k-1$.
4 *If* $f(a_1, \cdots, a_{r-1}, \pi) = 0$*, for a certain point* $(a_1, \ldots, a_{r-1}) \in S$*, then* $f(a'_1, \cdots, a'_{r-1}, \pi) = 0$*, for every* $(a'_1, \ldots, a'_{r-1}) \in S$.
5 *If* $(a_1, \ldots, a_{r-1}) \in S$*,* $b \in I_r$ *and* $f(a_1, \ldots, a_{r-1}, b) = 0$*, then there is an* i*,* $1 \leq i \leq k$*, such that* $b = g_i(a_1, \ldots, a_{r-1})$.

Notice that g_{k-1} should be less than π on S.

Let $\{p_1, \ldots, p_n\}$ be a set of $n > 0$ trigonometric polynomials in r variables. Their roots are delineable on $S \subset \mathcal{I}^{r-1}$ if the roots of $\prod_{i=1}^n p_i$ are delineable on S.

Let now $S \in \mathcal{I}^{r-1}$, connected, and F a set of trigonometric polynomials in r variables with roots delineable on S. We define the *stack over* S, similarly as Collins ((Collins, 1975), Section 3). Let $f_1, \ldots, f_\nu, \nu \geq 0$, delineate the roots of F on S.

If $\nu = 0$, we set

- $S_1 = S \times \mathbf{R}$, if x_r is an algebraic variable,
- $S_1 = S \times (-\pi, \pi]$, if x_r is an angle.

If $\nu > 0$,

- if x_r is an algebraic variable, let
  - $S_1 = \{(a, b) | a \in S, b < f_1(a)\}$,
  - $S_{2k} = \{(a, b) | s \in S, b = f_k(a)\}$, for $1 \leq k \leq \nu$,
  - $S_{2k+1} = \{(a, b) | s \in S, f_k(a) < b < f_{k+1}(a)\}$, for $1 \leq k < \nu$,
  - $S_{2\nu+1} = \{(a, b) | s \in S, f_\nu(a) < b\}$.

– if x_r is an angle variable, let

· $S_1 = \{(a,b) | a \in S, -\pi < b < f_1\}$,
· $S_{2k} = \{(a,b) | s \in S, b = f_k(a)\}$, for $1 \leq k \leq \nu$,
· $S_{2k+1} = \{(a,b) | s \in S, f_k(a) < b < f_{k+1}(a)\}$, for $1 \leq k < \nu$.

Since F is a set of trigonometric polynomials, each member of F is continuous on S. This implies that F keeps constant sign on each S_i, $1 \leq i \leq 2\nu + 1$ (for and algebraic variable) or $1 \leq i \leq 2\nu$ (for an angle variable). Therefore, if we can produce a decomposition of $\mathcal{I}^{r-1}$, fine enough, so that the polynomials occurring in F are delineable on each of the cells, this decomposition can be extended to one of $\mathcal{I}^r$ which will be sign-invariant with respect to F.

Collins, in his original paper (Collins, 1975), resolves this problem by constructing, from F, a set F_{r-1} of algebraic polynomials in $r-1$ variables, so that the roots of F are delineable on each cell from the sign-invariant decomposition of $\mathbf{R}^{r-1}$ with respect to F_{r-1}. He defines for this a *projection operator* Proj, for the quantified variables, and AProj, for the free variables. He proves that if Proj(F) is sign-invariant on a cell S, then the roots of F are delineable on S; similarly, if AProj(F) is sign-invariant on a cell S, then the roots of F and of all derivatives of polynomials in F are delineable on S. This ensures a sign-invariant decomposition of the whole space be constructed algorithmically.

We try to develop here a similar method for trigonometric polynomials. In the following section, we define the main tool that will be used during the decomposition process: the `tan` transformation. We will prove also some similar results as in Collins's paper (Collins, 1975), based on an extended projection operator, results that sustain theoretically our algorithm.

3. Description of the Method

It is known that, if $x \neq \pi$, the following identities hold:

$$\sin(x) = \frac{2\tan\frac{x}{2}}{1+\tan^2\frac{x}{2}},$$

$$\cos(x) = \frac{1-\tan^2\frac{x}{2}}{1+\tan^2\frac{x}{2}}.$$

In fact, they are valid for $x = \pi$ too, but here they are computed as limits of the corresponding functions.

Now, $\tan\frac{x}{2}$ is a very nice-behaving function. It is a bijection from $(-\pi, \pi)$ to $\mathbf{R}$, thus inversable. It is continuous, and strictly increasing.

[$\mathtt{tan}_k$] We denote by $\mathtt{tan}_k(F)$ the formula in $x_1, \ldots, x_{k-1}, t_k, x_{k+1}, \ldots, x_r$, obtained from F in the following way:

1 replace the trigonometric functions in x_k with their expressions in terms of $\tan\frac{x_k}{2}$;
2 replace $\tan\frac{x_k}{2}$ by t_k;
3 do all the computations and eliminate the denominators (since they are of the form $\Pi_j(1+t_k^2)^j$, thus always positive).

This transformation is an operator that takes as input a formula F defined on $\mathcal{I}$ and outputs a formula F' defined on $\mathcal{I}' = I_1 \times \ldots \times I_{k-1} \times \mathbf{R} \times I_{k+1} \times \ldots \times I_r$; the k'th variable of F' is algebraic.

In all what follows, we will refer the transformation defined above as the `tan` transformation.

This transformation has a major drawback: since the tangent of half angle is a bijection from $(-\pi, \pi)$ to $\mathbf{R}$, a sign-invariant decomposition of $\mathcal{I}'$ with respect to F' gives no information about the points in $I_1 \times \ldots \times I_{k-1} \times \pi \times I_{k+1} \times \ldots \times I_r$.

We cope with this problem by defining *trigonometric projection operators* and the *cylindrical trigonometric decomposition.*

3.1. Trigonometric Projection Operators

For the following notions, we refer to Collins's paper (Collins, 1975): leading coefficient (lcf), leading term (lt), reductum (red), resultant (res), subresultant, principal subresultant coefficient (psc), discriminant, etc. We recall first the definitions of the projection operators, given by Collins (Collins, 1975).

Let F be a set of polynomials in r variables, $x_1, \ldots, x_r$,

$$\begin{aligned}
&\mathcal{B} = \{\mathrm{red}^k(p) | p \in F, k \geq 0, \deg(\mathrm{red}^k(p)) > 1\},\\
&\mathcal{L} = \{\mathrm{lcf}(b) | b \in \mathcal{B}\},\\
&\mathcal{S}_1 = \left\{\mathrm{psc}_k(b, b') | b \in \mathcal{B}, b' = \frac{\partial b}{\partial x_r}, 0 \leq k \leq \deg(b')\right\},\\
&\mathcal{S}_2 = \{\mathrm{psc}_k(b1, b2) | b1, b2 \in \mathcal{B}, 0 \leq k < \min(\deg(b1), \deg(b2))\}, \text{ and}\\
&\mathcal{P} = \mathcal{L} \cup \mathcal{S}_1 \cup \mathcal{S}_2.
\end{aligned}$$

$\mathcal{P}$ is called *the projection* of F.

Let now

$$\begin{aligned}
&\mathcal{V} = \left\{\frac{\partial^k b}{\partial x_r^k} | b \in \mathcal{B}, 0 < k < \deg(b)\right\},\\
&\mathcal{P}' = \left\{\mathrm{psc}_k(d, d') | d \in \mathcal{V}, d' = \frac{\partial d}{\partial x_r}, 0 \leq k < \deg(d')\right\}.
\end{aligned}$$

$\mathcal{P} \cup \mathcal{P}'$ is called *the augmented projection* of F.

We denote by $Proj(F)$ the projection of F, and by $AProj(F)$, the augmented projection.

Definition 3.1. **(Projection Operators)** *Let* $F \subset Trig_{\mathbf{K}[x_{j_1}, \ldots, x_{j_{r-d}}]}[x_{i_1}, \ldots, x_{i_d}]$ *be a finite set of trigonometric polynomials. The* trigonometric projection operators *are defined as follows:*

$$Proj_T(F) = \begin{cases} Proj(F), & \text{if } x_r \text{ is an algebraic variable,}\\ Proj(\mathtt{tan}_r(F)) \cup F|_{[x_r=\pi]}, & \text{if } i_d = r.\end{cases}$$

$$AProj_T(F) = \begin{cases} AProj(F), & \text{if } x_r \text{ is an algebraic variable,}\\ AProj(\mathtt{tan}_r(F)) \cup F|_{[x_r=\pi]}, & \text{if } i_d = r.\end{cases}$$

Here we have denoted with $F|_{[x_r=\pi]}$ *the set of polynomials obtained by substituting* π *for* x_r *in* F.

3.2. Cylindrical Trigonometric Decompositions

We define a *cylindrical trigonometric decomposition* (ctd) $\mathcal{D}$ of $\mathcal{I}$ similarly to (Collins, 1975):

For $r = 1$,

- if x_1 is an algebraic variable, let $\beta_1 < \ldots < \beta_\nu$ be algebraic numbers; then $\mathcal{D}$ is a sequence $S_1 \ldots, S_{2\nu+1}$, where
  * either $\nu = 0$ and $\mathcal{D} = I_1$,
  * or $\nu > 0$,
    · $S_{2i} = \{\beta_i\}$, for $1 \leq i \leq \nu$,
    · $S_1 = (-\infty, \beta_1)$,
    · $S_{2\nu+1} = (\beta_\nu, \infty)$,
    · and $S_{2i+1} = (\beta_i, \beta_{i+1})$, for $1 \leq i < \nu - 1$;
- if x_1 is an angle variable, let $-\pi < \beta_1 < \ldots < \beta_\nu = \pi$ be roots of trigonometric polynomials; then $\mathcal{D}$ is a sequence $S_1 \ldots, S_{2\nu}$, where
  * $\nu > 0$,
  * $S_{2i} = \beta_i$ for $1 \leq i \leq \nu$,
  * $S_1 = (-\pi, \beta_1)$, and
  * $S_{2i+1} = (\beta_i, \beta_{i+1})$, for $1 \leq i < \nu - 1$.

Let $r > 1$, and let $\mathcal{D}^{r-1} = (S_1, \ldots, S_\mu)$ be a ctd of $I_1 \times \ldots \times I_{r-1}$;

- if x_r is an algebraic variable, let $f_{i,1} < f_{i,2} < \ldots < f_{i,\nu_i}$ be continuous algebraic functions defined on S_i;
  * if $\nu_i = 0$, set $S_{i,1} = \mathbf{R}$;
  * if $\nu_i > 0$,
    · set $S_{i,2j} = \{(a,b)|a \in S_i \text{ and } b = f_{i,j}(a)\}$, for $1 \leq j \leq \nu_i$, and
    · for $1 \leq j < \nu_i$, set $S_{i,2j+1} = \{(a,b)|a \in S_i \text{ and } f_{i,j}(a) < b < f_{i,j+1}(a)\}$;
    · set $S_{i,1} = \{(a,b)|a \in S_i \text{ and } b < f_{i,1}(a)\}$ and
    · $S_{i,2\nu_i+1} = \{(a,b)|a \in S_i \text{ and } f_{i,\nu_i}(a) < b\}$;
- if x_r is an angle variable, for $1 \leq i \leq \mu$, let $f_{i,1} < f_{i,2} < \ldots < f_{i,\nu_i}$, $\nu_i > 0$, be continuous analytic functions defined on S_i, with $f_{i,\nu_i}(a) = \pi$, for all $a \in S_i$;
  * set $S_{i,2j} = \{(a,b)|a \in S_i \text{ and } b = f_{i,j}(a)\}$, for $1 \leq j \leq \nu_i$, and
  * for $2 < j \leq \nu_i$, set $S_{i,2j-1} = \{(a,b)|a \in S_i \text{ and } f_{i,j}(a) < b < f_{i,j+1}(a)\}$;
  * $S_{i,1} = \{(a,b)|a \in S_i \text{ and } -\pi < b < f_{i,1}(a)\}$.

A ctd of I^r is any sequence $S_{1,1}, \ldots, S_{1,\nu_1}, \ldots, S_{\mu,\nu_\mu}$ that can be obtained in this way.

If $\mathcal{P}$ is a set of trigonometric polynomials and S is a ctd in $\mathcal{I}$, $\mathcal{P}$ is *sign invariant* on S if every polynomial in $\mathcal{P}$ has constant sign on each cell of S.

The notions of *sample points*, *cylindrical samples*, *standard formula defining a cell*, etc. have the same definitions as in (Collins, 1975); the only difference is that, instead of *algebraic polynomials*, we should use *trigonometric polynomials*.

In the following we show how the trigonometric delineability if related to the concept of delineability as introduced in (Collins, 1975).

Lemma 3.1. **(Direct Delineability 1)** *Let F be a set of trigonometric polynomials in r variables, x_r an angle variable, and c a cell in the ctd $\mathcal{D}^{r-1}$. If F is delineable over c, then $F|_{[x_r=\pi]}$ is sign-invariant on c.*

Proof. Let $\pi < g_1 < \ldots < g_\nu = \pi$ be the functions in $r-1$ variables that delineate the roots of F, as in Definition 2.3.

Suppose that $F|_{[x_r=\pi]}$ is not sign-invariant on c; then there exist a polynomial $p \in F$ and $a_1, a_2 \in c$ such that $p(a_1,\pi) < 0, p(a_2,\pi) \geq 0$. Since p is continuous, by the intermediate value theorem there is $0 < \alpha \leq 1$ so that, for $b = \gamma(\alpha)$, we have $p(b,\pi) = 0$; here we have denoted with γ a curve defined on $[-1,1]$ with values in c, so that $\gamma(0) = a_1$ and $\gamma(1) = a_2$. The connectivity of c ensures the existence of such a curve. From this we deduce that $p(a_1,\pi) = 0$, which contradicts our assumption. □

Lemma 3.2. **(Direct Delineability 2)** *Let F be a set of trigonometric polynomials in r variable, x_r an angle variable, and c a cell in the ctd $\mathcal{D}^{r-1}$. If F is delineable on c, then $\mathtt{tan}_r(F)$ is delineable on c.*

Proof. Let $\pi < g_1 < \ldots < g_\nu = \pi$ be the functions in $r-1$ variables that delineate the roots of F, as in Definition 2.3. The functions $g_1, \ldots, g_{\nu-1}$ describe all the roots of F on c, except, maybe, π. By the bijectivity of the tangent of the half angle, for a point $a \in c$ and a polynomial $p \in F$, $p(a, g_i(a)) = 0$ iff $\mathtt{tan}_k(p)(a, \tan\frac{g_i(a)}{2}) = 0$, for $1 \leq i \leq \nu-1$. Since the tangent transformation preserves the multiplicity of a root of a function, a simple check shows that $\tan\frac{g_i}{2}$, for $i \leq i < \nu$, delineates the roots of $\mathtt{tan}_k(F)$. □

Lemma 3.3. **(Reverse Delineability)** *Let F be a set of trigonometric polynomials in r variables, x_r an angle variable, and c a cell in the ctd $\mathcal{D}^{r-1}$. If:*

> $\mathtt{tan}_r(F)$ *is delineable on c, and*
> $F|_{[x_r=\pi]}$ *is sign-invariant on c*

then F is delineable on c.

Proof. Let $f_1 < \ldots < f_{\nu-1}$ be the functions in $r-1$ variables that delineate the roots of $\mathtt{tan}_r(F)$, as in Definition 2.3. We use again the argument of the bijectivity of the tangent of half angle to deduce that the functions $g_i := \mathrm{atan}\frac{f_i}{2}$, $1 \leq i \leq \nu-1$, together with $g_\nu = \pi$, delineate F.

Now, since $F|_{[x_r=\pi]}$ is sign-invariant on c, $g_\nu(a)$ is either a root of F for all $a \in c$, or no polynomial in F has it as a root, for any $a \in c$. □

Theorem 3.1. **(Algebraic Delineability)** *Let F be a set of polynomials in r variables, x_r an algebraic variable. If $Proj_T(F)$ is sign-invariant on $c \subset \mathcal{I}^{r-1}$, then F is delineable over c.*

We do not give here a complete and formal proof of this theorem, but we rather refer to Collins (Collins, 1975) Theorem 5. The only change in the hypotheses is the fact that the polynomials we are dealing with are trigonometric, not algebraic. However, even for this kind of polynomials, the arguments given in the proof of Collins' Theorem 5 remain valid. The same is true for the series of statements given in the same paper as theoretical preparations for the theorem.

THEOREM 3.2. **(Trigonometric Delineability)** *Let F be a set of polynomials in r variables, with x_r an angle variable. If $Proj_T(F)$ is sign-invariant on a connected subset c of $\mathcal{I}^{r-1}$, then F is delineable over c.*

PROOF. If x_r were an algebraic variable, Theorem 3.1 would ensure the delineability of the roots of F on every cell of $\mathcal{D}$. Since x_r is an angle, we try to transform the problem to the algebraic case.

Let c be a connected cell in $\mathcal{I}^{r-1}$ and consider the set of polynomials $\mathtt{tan}_r(F)$; they are algebraic in the r'th variable. Since $Proj_T(F)$ contains their algebraic projection, by Theorem 3.1 their roots are delineable over c.

The particular expression of the trigonometric projection operator ensures the sign-invariance of $F|_{[x_r=\pi]}$ on c. This means that we have the assumptions of Lemma 3.3, therefore F is delineable on c. $\square$

The following theorem is a direct consequence of Theorems 3.1 and 3.2:

THEOREM 3.3. **(Delineability)** *Let F be a set of polynomials in r variables. If $Proj_T(F)$ is sign-invariant on a connected subset $c \subset \mathcal{I}^{r-1}$, then F is delineable over c.*

Remark that a similar theorem can not be given for the augmented projection, since in general $\mathtt{tan}_r(p') \neq (\mathtt{tan}_r(p))'$. However, the trigonometric augmented projection, as in Definition 3.1, ensures the delineability of all the derivatives of polynomials in $\mathtt{tan}_r(F)$, if x_r is an angle variable. Thus, in the lifting phase, a description of the functions $f_1, \ldots, f_{\nu-1}$, as in the proof of Theorem 3.2, can be computed; applying `atan` to these functions, one gets a description of the roots of F.

4. The Algorithm

The algorithm is very similar to Collins's quantifier elimination by cylindrical algebraic decomposition ((Collins, 1975)). The first phase is the computation of a *cylindrical trigonometric decomposition* (ctd) of the space of free variables, together with a *cylindrical trigonometric sample* (cts) for the whole space. The second phase consists in evaluating the initial formula in the sample points, and isolating from the ctd the cells in which the formula is true.

The computation of the ctd is done recursively, by eliminating in turn each variable, using projection operators, until a one-dimensional set of polynomials is obtained, and then by computing in turn d-dimensional decompositions from $d-1$-dimensional decompositions, for $1 < d < f-1$, where f is the number of free variables in the initial formula. The process also carries out the computation of d-dimensional sample points, this time for $1 \leq d \leq r$.

4.1. Cylindrically Decomposing the Space

The algorithm that computes the cylindrical decomposition of the space works recursively. It gets as input a set F of polynomials in r variables and f, the number of free variables, and returns a symbolic description of a cylindrical decomposition $\mathcal{D}_f$ of $\mathcal{I}^f$ and a set of sample points belonging to a cylindrical decomposition $\mathcal{D}_r$ of $\mathcal{I}^r$. $\mathcal{D}_r$ is sign-invariant with respect to the polynomials in F, and the cells of $\mathcal{D}_f$ are the geometric projections of the cells of $\mathcal{D}_r$.

The first step in the decomposition algorithm computes the **projection** set of polynomials in F using $Proj_T(F)$ (if $f < r$) or $AProj_T(F)$ (if $f = r$), and solves recursively the decomposition problem for $r-1$ variables.

At the second step, once the problem is solved for $r-1$ variables, the result returned by the recursive call is **lifted** to the r-variables case. If $f < r$, only the sample is lifted, otherwise the decomposition is lifted too.

The lifting of the $r-1$-dimensional cells to r-dimensional ones is done by calling the procedure `Define`. A complete specification of this procedure can be found in (Collins, 1975). We give here an informal description.

Defining the Cells in the r-Dimensional Space

Let p be a polynomial and c' an $r-1$-dimensional cell on which p and its derivatives are delineable. Let b' be the sample point of c'. The algorithm `Define` determines the symbolic description of the stack of cells in the cylinder $c' \times \mathbf{R}$. This description is meant to be given as a quantifier-free formula involving integral polynomials, namely, p and its derivatives.

Since the roots of p are delineable, each root u of $p^* := p(b', x_r)$, (which is a polynomial in x_r) belongs to a component of the solutions of $p(x_1, \ldots, x_r) = 0$ in $c' \times \mathbf{R}$. Using these roots we can determine the sign of p in each cell in the stack. In most of the cases this is not enough, because there can be many cells in which p has the same sign.

This is the reason why the derivatives must be involved: It is well-known that any two roots of p^* are separated by at least one root of its derivative. This means that, if we have a decomposition of the cylinder with respect to the derivative of p, each cell in this decomposition will have at most one root of p. Now the cells of the decomposition made by the roots of p is given in terms of the cells determined by its derivative conjuncted with relations of the form $p < 0, p = 0$ or $p > 0$.

Again, the process is carried out recursively; this time, first a decomposition of the cylinder over c with respect to the derivative is computed, and then this decomposition is used to construct the stack with respect to the roots of p.

It is obvious that this algorithm works for trigonometric polynomials as well, when the last variable is an algebraic variable.

We continue now with the description of the decomposition algorithm `TDecomp`; it is called as `TDecomp`(f, r, F, A), and returns the list $(\mathcal{D}, \beta)$. (See the sketch of `TDecomp` in Figure 1.) A is the set of angle variables. In the lifting phase, if the current variable, x_r, is in A, then the symbolic description of the cells of the decomposition is computed by considering the `tan`-transformed polynomials in F – since `Define` works only with algebraic polynomials in x_r. The decomposition thus obtained must be transformed afterwards, using the reverse `tan` transformation. Moreover, in order to get a ctd, we have to add cells of the form (c, π).

It remains to describe how to extend the sample β' of the decomposition of $\mathcal{I}^{r-1}$ to the sample β of the decomposition of $\mathcal{I}^r$.

Let b' be a sample point of a certain cell $c' \in \mathcal{D}'$. Each cell in the stack over c' will have

TDecomp

Input: f - the number of free variables;
r - the number of variables;
F - a set of trigonometric polynomials in r variables;
A – the set of angle variables.

Output: D –a ctd of $\mathcal{I}^f$;
β – a cylindrical trigonometric sample of $\mathcal{I}^r$.

begin
  if $r = 1$ then
    if $x_1 \in A$ then $aF := \mathtt{tan}_1(F)$ else $aF := F$;
    compute the isolation intervals for the real roots of polynomials in aF;
    using these intervals, generate a decomposition D of $\mathbf{R}$
      given symbolically as relations involving polynomials in aF
        and linear inequalities;
    if $x_1 \in A$ then
      apply the reverse $\mathtt{tan}$ transformation with respect to x_1 to the definition of D;
      add the cell "$\cos(x_1) = -1$" as the last cell of the decomposition;
    compute the sample β with respect to aF;
  else
    if $f < r$ then
      $pF := Proj_T(F)$; $f' := f$; $r' := r - 1$;
    else
      $pF := AProj_T(F)$; $f' := f - 1$; $r' := r - 1$;
    apply **TDecomp** to (f', r', pF, A'), yielding (D', β'),
      where D' is a ctd of $\mathcal{I}^{r-1}$, and β' the sample;
    extend β' to a set β of cylindrical sample for $\mathcal{I}^r$;
    if $f = r$ then
      $D := \emptyset$;
      for each cell $c \in D'$, with sample point β'_c
        let $p_1, \ldots, p_m$ be the polynomials in $\mathtt{tan}_r(F)$, if $x_r \in A$,
          or in F, if $x_r \notin A$;
        apply Collins's **Define** to compute a stack over c, ϕ,
          with respect to p_1, using the sample point β'_c;
        for $j := 2$ to m
          apply Collins's **Define** to compute a stack over c, ϕ^*
            with respect to p_j, using the sample point β'_c;
          refine ϕ so that the cells of ϕ^* can be decomposed
            into unions of cells in ϕ;
        add the stack ϕ to D;
      if $x_r \in A$ then
        apply the reverse $\mathtt{tan}$ transformation with respect to x_r
          to the definition of D
        for each cell $d \in D'$ with sample point $b_d \in \beta'$
          add the cell $d \wedge (\cos(x) - 1 = 0)$ to D;
          add the sample point (b_d, π) to β;
    else
      $D := D'$;
  return (D, β);
end.

Figure 1. The algorithm for computing a cylindrical trigonometric decomposition.

a sample point of the form (b', b_r). As these cells are defined by the roots of $\prod_{p\in F} p$, the last coordinates of the cells of even index (these correspond to the roots of polynomials in F) will be exactly the roots of the equation $\prod_{p\in F} p(b', x) = 0$. The cells of odd index will have sample points whose last coordinates have intermediate values.

If x_r is an angle variable, the sample points obtained as before will have the last coordinate in the algebraic space. Unlike for the cell description, these sample points are *not* transformed back to the angle space (by substituting $2\text{atan}\,(x_r)$ for x_r). This seems to be an inconsistency, because the sample points are stored in their algebraic form, and the polynomials that should be evaluated at these points are trigonometric.

It is easy to handle these evaluations; when a trigonometric polynomial p has to be evaluated at a point $b = (x_1^*, \ldots, x_r^*)$, for each coordinate $x_{i_d}^*$ corresponding to an angle variable, $2\text{atan}\,(x_{i_d}^*)$ should be plugged in p. This is easily done symbolically by using the relations presented at the beginning of section 3, namely, $\sin(2\text{atan}\,(x_{i_d}^*)) = 2x_{i_d}^*/(1+(x_{i_d}^*)^2)$.

A special case is the cells of the form $cos(x) = 0$; here the last coordinate of the corresponding sample points is stored as the symbol π.

4.2. Extracting the True-valued Cells

The second phase in the quantifier elimination algorithm consists in the extraction, from the decomposition computed previously, of the cells on which the initial formula Φ is true:

$$\Phi = \Theta_f\, x_f \ldots \Theta_r\, x_r\, R(x_1, \ldots, x_r).$$

The symbolic description of these cells is returned as a quantifier-free formula equivalent to Φ.

Let $\mathtt{D}_f$ be the decomposition of the f-dimensional space of the free variables. Collins' procedure `Eval` evaluates Φ in each cell of $\mathcal{D}_f$; it proceeds recursively, by computing the truth value of each cell c in $\mathcal{D}_i$ from the truth values of the cells in the stack over it, in $\mathcal{D}_{i+1}$, and does this for $f \le i < r$.

For a cell c in $\mathcal{D}_r$, its truth value is $R(\beta_c)$, where β_c is the sample point of c. Since all the polynomials occuring in R are sign-invariant on c, this truth value is the same in every other poit of c.

Let c be a cell in $\mathcal{I}^{r-1}$, and assume that the variable x_r is existentially quantified. The truth value of c will be:

$$\bigvee_{s\in S_c} R(s);$$

here we have denoted by S_c the stack of cells over c, in $\mathcal{D}_r$. If x_r is universally quantified, the truth value of c will be

$$\bigwedge_{s\in S_c} R(s).$$

There is no trouble in extending the procedure `Eval` to the trigonometric case.

Problem 2.1 can be solved by applying `TDecomp` and `Eval` (generalized to the trigonometric situation).

5. Algorithm Analysis

We recall that the number of operations executed by Collins's quantifier elimination algorithm is

$$O((2n)^{2^{2r+8}} m^{2^{r+6}} norm_{max}^3 n_{at-f}),$$

where:

r is the number of variables,
m is the number of polynomials,
n is the maximum degree of any polynomial,
$norm_{max}$ is the maximum of the norms of these polynomials,
n_{at-f} is the number of occurrences of atomic formulas in Φ.

It suffices to estimate the complexity of `TDecomp`. To do this, we compare its performance with the performace of Collins' algorithm `Decomp` applied to a set of polynomials obtained from F by `tan`-transforming all angle variables, together with the additional linear polynomials of the from x_{i_l}, for each angle variable x_{i_l}.

In the projecting phase corresponding to an angle variable x_{i_l}, the set of extra polynomials added by substituting π for x_{i_l} is comparable with the set of extra polynomials resulting from the additional input polynomial x_{i_l}.

In the lifting phase corresponding to an angle variable x_{i_l}, the set and the size of the extra cells with $x_{i_l} = \pi$ is comparable with the set and the size of the extra cells added by the contribution of the additional polynomial x_{i_l}.

Each `tan`-transform may double the degree in one of the angle variables. These raises of degrees are however independent of each other, therefore the total degree of polynomials obtained from F by transforming all angle variables is bounded by $2n$.

From these it follows that the number of operations is approximated from above by considering $m+d$ as the number of initial polynomials, and $2n$ as their maximum degree.

It remains to analyse the time spent for the `tan` and reverse `tan` transformations. In both cases, the Horner scheme can be used for doing the substitutions. This method is known to be linear with respect to the degree of polynomials. However, in our cases binomials are involved in the substitutions, which leads to a complexity quadratic in the degree and linear in the norm. Using the estimations for the degree and norm of the involved polynomials, it is easy to show that the costs are not dominant.

We get now for our algorithm:

$$O((4n)^{2^{2r+8}} (m+d)^{2^{r+6}} norm_{max}^3 n_{at-f}); \tag{5.1}$$

here we have kept the same notations as before.

Analysis of the naive method

The naive method consisted in substituting new variables for each sine and cosine that occur in R and adding the relations corresponding to $\sin^2(x_{i_l}) + \cos^2(x_{i_l}) = 1$, for each angle variable x_{i_l}. This means that:

- the number of variables increases with d – the number of angle variables;
- the number of polynomials in F increases with d;
- the number of atomic formulas in R increases by d.

The estimate of the number of operations is now:

$$O((2n+2d)^{2^{2r+2d+8}}(m+d)^{2^{r+d+6}} norm_{max}^3 at_f + d).$$

Obviously, this order is much bigger than in expression (5.1).

References

Arnon, D. S., Collins, G. E., McCallum, S. (1988). An adjacency algorithm for cylindrical algebraic decompositions of three-dimensional space. *Journal of Symbolic Computation*, **5**(1,2).

Arnon, D. S., McCallum, S. (1982). Cylindrical algebraic decomposition by quantifier elimination. In *Proceedings of the European Computer Algebra Conference (EUROCAM '82)*, volume 144 of *Lecture Notes Comp. Science.*, pages 215–222.

Collins, G. E. (1975). Quantifier elimination for the elementary theory of real closed fields by cylindrical algebraic decomposition. In *Lecture Notes In Computer Science*, pages 134–183. Springer. Vol. 33.

Collins, G. E. (1994). Quantifier elimination by cylindrical algebraic decomposition - 20 years of progress. In Caviness, B., Johnson, J., editors, *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Texts and Monographs in Symbolic Computation. Springer. To appear.

Collins, G. E., Hong, H. (1991). Partial cylindrical algebraic decomposition for quantifier elimination. *Journal of Symbolic Computation*, **12**(3):299–328.

Collins, G. E., Johnson, J. R. (1989). Quantifier elimination and the sign variation method for real root isolation. In *Proceedings of the ACM-SIGSAM 1989 International Symposium on Symbolic and Algebraic Computation*, pages 264–271. ACM.

Dolzmann, A., Sturm, T. (1997). Simplification of quantifier-free formulas over ordered fields. *Journal of Symbolic Computation*, **24**:450–462.

Dorato, P., Yang, W., Abdalla, C. (1997). Robust multi-objective feedback design by quantifier elimination. *Journal of Symbolic Computation*, **24**:153–160.

Gónzalez-López, M. J., Recio, T. (1993). Path tracking in motion planning. *The Computer Journal*, **36**(5):515–524. In a special issue on *computational quantifier elimination*, edited by H. Hong.

González-Vega, L. (1996). Applying quantifier elimination to the birkhoff interpolation problem. *Journal of Symbolic Computation*, **22**:83–103.

González-Vega, L. (1998). A combinatorial algorithm solving some quantiifier elimination problems. In *Quantifier Elimination and Cylindrical Algebraic Decomposition*, pages 365–375. Springer.

Gutierrez, J., Recio, T. (1998). Advances on the simplification of sine-cosine equations. *Journal of Symbolic Computation*, **26**:31–71.

Hong, H. (1996). The exact region of stability for maccormack scheme. *Computing*, **56**(4).

Hong, H. (1997). Testing stability by quantifier elimination. *Journal of Symbolic Computation*, **24**:161–188.

Hong, H., Schicho, J. (1998). Algorithms for trigonometric curves (simplification, implicitization, parameterization). *J. Symb. Comp.*, **26**(3):279–300.

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

Liska, R., Steinberg, S. (1993). Applying Quantifier Elimination to Stability Analysis of Difference Schemes. *The Computer Journal*, **36**(5):497–503.

Loos, R., Weispfenning, V. (1993). Applying linear quantifier elimination. *Computer Journal*, **36**(5):450–462.

McCallum, S. (1988). An improved projection operator for cylindrical algebraic decomposition. *Journal of Symbolic Computation*, **5**(1,2).

Renegar, J. (1992). On the computational complexity and geometry of the first-order theory of the reals i,ii,iii. *Journal of Symbolic Computation*, **13**(3):255–352.

Tarski, A. (1998). A decision method for elementary algebra and geometry. In Caviness, B., Johnson, J., editors, *Quantifier elimination and cylindrical algebraic decomposition*, pages 24–84. Springer.

Weispfenning, V. (1997). Simulation and optimization by quantifier elimination. *Applicable Algebra in Engineering, Communication and Computing*, **8**:85–101.

Weispfenning, V. (1998). A new approach to quantifier elimination for real algebra. In Caviness, B., Johnson, J., editors, *Quantifier Elimination and Cylindrical Algebraic Decomposition.* Springer. Texts and Monographs in Symbolic Computation.