MIT 6.7230 - Algebraic techniques and semidefinite optimization April 19, 2024

Lecture 16

Lecturer: Pablo A. Parrilo *Scribe: ???*

Quantifier elimination (QE) is a very powerful procedure for problems involving first-order formulas over real fields. The cylindrical algebraic decomposition (CAD) is a technique for "efficient" implementation of QE, that effectively reduces a seemingly infinite problem into a finite (but potentially large) instance. For much more information about QE and CAD (including a reprint of Tarski's original 1930 work), we recommend the book [CJ98].

1 Quantifier elimination

A quantifier-free formula is an expression consisting of polynomial equations ($f(x) = 0$) and inequalities ($f(x) \geq 0$) combined using the Boolean operators $\neg$ (negation), $\wedge$ (and), $\vee$ (or), and $\Rightarrow$ (implies). We often also allow strict inequalities $f(x) > 0$ and inequations $f(x) \neq 0$, since these are just shorthands for particular boolean combinations of equations and inequalities.

In general, a *formula* (in prenex form) is an expression in the variables $x = (x_1, \ldots, x_n)$ of the type:

$$(Q_1 x_1)...(Q_s x_s) \quad \mathcal{F}(f_1(x), \ldots, f_r(x)) \tag{1}$$

where each Q_i is one of the quantifiers $\forall$ (for all) and $\exists$ (there exists). Furthermore, $\mathcal{F}(f_1(x), ..., f_r(x))$ is assumed to be a quantifier-free formula. If there is a quantifier corresponding to the variable x_i, we say that x_i is *quantified*, or *free* otherwise.

Example 1. *The following are valid formulas*

$$(\forall x)\, [(x \geq 0) \Rightarrow (x^2 + ax + b \geq 0)]$$
$$(\forall x)(\exists y)\, [x > y^2]$$
$$(\forall \delta)(\exists \epsilon)\, [(\epsilon^2 + \delta^2 \leq 1) \vee (\epsilon \neq 0)] \Rightarrow [\delta < 1].$$

The first formula has two free variables (since the variables a and b are unquantified), while for the other two all variables are quantified.

We will interpret the symbols in a formula as taking only real values. Notice that a formula without free variables (usualled called a *closed* formula or a *sentence*) is either true or false. For instance, the last two expressions in Example 1 are sentences, with the first one being false and the second being true. Notice also that the truth value may depend on the order of the quantifiers.

Tarski showed that for every formula including quantifiers there is always an equivalent quantifier free formula. Obtaining the latter from the former is called quantifier elimination.

Theorem 2 (Tarski-Seidenberg)**.** *For every first-order formula over the real field there exists an equivalent quantifier-free formula. Furthermore, there is an explicit algorithm to compute this quantifier-free formula.*

The Tarski-Seidenberg theorem is an extremely powerful result, since it provides a complete characterization and algorithmic technique for an extremely large collection of problems involving polynomials. Unfortunately, there are very serious computational barriers to the efficient practical implementation of these ideas, since the resulting algorithms have extremely poor scaling properties, with respect to the number of variables (towers of exponentials). Newer methods, such as the (partial) cylindrical algebraic decomposition (CAD) technique due to Collins and described below, or the critical point method, are by comparison much better. Nevertheless, they still behave exponentially (or worse) in terms of the number of variables (and likely this is required, modulo complexity-theoretic conjectures).

2 Tarski-Seidenberg

Example 3. *Consider the quantified first-order formula:*

$$(\forall x)(\forall y)\,[(x^2 + ay^2 \leq 1) \Rightarrow (ax^2 - a^2xy + 2 \geq 0)]. \tag{2}$$

This formula is equivalent to the quantifier free expression:

$$(a \geq 0) \wedge (a^3 - 8a - 16 \leq 0),$$

which defines the interval $[0, a_\star]$*, where* $a_\star \approx 3.538$*. Thus, the original expression (2) is true only for* $a \in [0, a_\star]$*.*

2.1 Geometric interpretation

The Tarski-Seidenberg theorem allows us to understand what happens when we apply certain operations to semialgebraic sets. For instance, many natural constructions such as set closure, convex hulls, conic hulls, projections, etc. can easily be described in terms of first-order formulas. This implies that, by eliminating these quantifiers, "simpler" quantifier-free descriptions of these sets can be obtained. In particular, an important geometric interpretation of the Tarski-Seidenberg theorem is the following:

Theorem 4. *The projection of a semialgebraic set is semialgebraic.*

Recall that a spectrahedron is a basic semialgebraic set. Linear projections of spectrahedra are not necessarily basic semialgebraic (recall the Examples in Lecture 5), but they are always semialgebraic sets.

2.2 Applications

Static output feedback An early application of Tarski-Seidenberg in control theory was the "solution" of the static output feedback stabilization problem in [ABJ75]. Given matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, we want to find a matrix $K \in \mathbb{R}^{m\times p}$ such that the matrix $A + BKC$ is Hurwitz, i.e., all its eigenvalues are in the left-hand plane. Since the existence of such a matrix can be easily expressed as a formula in first order logic[1], the decidability and existence of an effective (but not efficient) algorithm immediately follows.

Simultaneous stabilization A very interesting result by Blondel [Blo94, BG93] shows that the simultaneous stabilization of three linear time-invariant systems is *not* decidable (and thus, cannot be semialgebraic). Notice however that the Tarski-Seidenberg theorem implies that for any given bound on the degree of the controller, the problem is decidable.

[1] For instance, $(\exists K)(\forall x)(\forall \lambda)\,[(A + BKC)x = \lambda x \vee x \neq 0] \Rightarrow [\Re(\lambda) \leq 0]$. Notice that we are being a bit sloppy with notation, since for a fully real formulation, we should split x and λ into real and imaginary parts. There are many other equivalent expressions, using for instance a Lyapunov equation, or the Routh array.

Game theory Nash equilibria, as well as almost all other game-theoretic solution concepts, can be expressed as first-order formulas over the reals. For suitable classes of games, and under the right conditions, (e.g., finite games, polynomial payoffs, stochastic games, etc.) this implies that equilibrium sets are semialgebraic; see e.g. [SSZ91].

3 Cylindrical Algebraic Decomposition (CAD)

There are a few approaches for effective implementation of the QE procedure. One of the most well-known, which is also relatively easy to understand, is the cylindrical algebraic decomposition (CAD) due to Collins [Col75]. We describe the basic elements of this approach below. We remark that much better algorithms (in the theoretical complexity sense) are known; see for instance the article by Renegar [Ren91] (also reprinted in [CJ98]) or [BPR03]. In particular, for CAD the number of operations usually scales in a doubly exponential fashion with the number of variables, while the newer methods are doubly exponential in the number of *quantifier alternations.*

3.0.1 Description

Given a set P of multivariate polynomials in n variables, a CAD is a special partition of $\mathbb{R}^n$ into components, called *cells*, over which all the polynomials have constant signs. The algorithm for computing a CAD also provides a point in each cell, called *sample point*, which can be used to determine the sign of the polynomials in the cell.

A cell is called *cylindrical* if it has the form $S \times \mathbb{R}^k$, for some $k \leq n$. A decomposition of $\mathbb{R}^n$ is a CAD if all polynomials have constant sign on each cell, and all cells are cylindrical.

The CAD associated to the formula (1) depends only on its quantifier-free part $\mathcal{F}(f_1(x), \ldots, f_r(x))$. Since all possible truth values of the formula are in correspondence with the values at the sample points, we can use the CAD to evaluate its truth value, and to perform quantifier elimination.

The basic CAD construction consists of two steps: *projection* and *lifting* (plus an additional third one, if formula construction is desired).

In the first projection phase, we compute successive sets of polynomials in $n-1, n-2, \ldots, 1$ variables. The main idea is, given an input set of polynomials, to compute at each step a new set of polynomials obtained by eliminating one variable at a time. In general, the elimination order does matter and a good choice leads to lower computational complexity.

The second phase (lifting) constructs a decomposition of $\mathbb{R}$, at the lowest level of projection, after all but one variable have been eliminated. This decomposition of $\mathbb{R}$ is successively extended to a decomposition of $\mathbb{R}^n$.

The basic operations necessary in the construction of CADs are (sub)resultants and (sub)discriminants.

`Complete` ToDo

An pretty complete implementation of (an improved version of) the CAD method for quantifier elimination is the software package QEPCAD [Bro03]. The software Mathematica has a restricted implementation of QE, via the `Resolve[]` command. The theorem prover Z3 [dMB08] from Microsoft Research also implements some limited cases of QE for the NRA fragment (Polynomial Real Arithmetic).

References

[ABJ75] B. D. O. Anderson, N. K. Bose, and E. I. Jury. Output feedback stabilization and related problems—solution via decision methods. *IEEE Transactions on Automatic Control*, 20:53–66, 1975.

[BG93] V. Blondel and M. Gevers. Simultaneous stabilizability of three linear systems is rationally undecidable. *Mathematics of Control, Signals, and Systems*, 6(2):135–145, 1993.

[Blo94] V. Blondel. *Simultaneous stabilization of linear systems*, volume 191 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag London Ltd., London, 1994.

[BPR03] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in real algebraic geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 2003.

[Bro03] C.W. Brown. *QEPCAD - Quantifier Elimination by Partial Cylindrical Algebraic Decomposition*, 2003. Available from `https://www.usna.edu/CS/qepcadweb/B/QEPCAD.html`.

[CJ98] B. F. Caviness and J. R. Johnson, editors. *Quantifier elimination and cylindrical algebraic decomposition*, Texts and Monographs in Symbolic Computation, Vienna, 1998. Springer-Verlag.

[Col75] G. E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata theory and formal languages (Second GI Conf., Kaiserslautern, 1975)*, pages 134–183. Lecture Notes in Comput. Sci., Vol. 33. Springer, Berlin, 1975.

[dMB08] Leonardo Mendonça de Moura and Nikolaj S. Bjørner. Z3: an efficient SMT solver. In C. R. Ramakrishnan and Jakob Rehof, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 14th International Conference, TACAS*, volume 4963 of *Lecture Notes in Computer Science*, pages 337–340. Springer, 2008.

[Ren91] J. Renegar. Recent progress on the complexity of the decision problem for the reals. In *Discrete and computational geometry (New Brunswick, NJ, 1989/1990)*, volume 6 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 287–308. Amer. Math. Soc., Providence, RI, 1991.

[SSZ91] S.H. Schanuel, L.K. Simon, and W.R. Zame. The algebraic geometry of games and the tracing procedure. *Game equilibrium models*, 2:9–43, 1991.