Overview of Quantifier Elimination (QE) algorithms

- What is quantifier elimination (variable elimination)?
- Quantifier elimination algorithms.
- Applications:
  - Geometry theorem proving,
  - A decidability result,
  - Control systems, feedback design.

What is quantifier elimination?

- Given a formula with quantifiers find an equivalent formula without quantifiers.
- Example: $\phi(a, b, c) = \forall x(ax^2 + bx + c > 0)$
- Note that a, b, c are *parameters* (free variables) and x is the quantified variable to be eliminated.
- The above $\phi(a, b, c)$ formula is equivalent to $\phi'(a, b, c) = (a > 0 \wedge b^2 < 4ac)$.

What is quantifier elimination?

(continued)

- A decision method for elementary algebra and geometry by Alfred Tarski (1951) has two steps:
- One can find in a mechanical way an equivalent formula without quantifiers.
- One can decide about a quantifier free sentence in a mechanical way whether it is true or not.
- The formulas are built up from variables ranging over the reals, addition, multiplication, equality, greater than equal, logical connectives ($\sim, \wedge, \vee$) and the existential quantifier.
- Decidability v. existence of quantifier elimination method.

What is quantifier elimination?

(continued)

- The language that is used to define the formulas determines the existence of a quantifier elimination method.
- If the real numbers are substituted with integers in the above language (integer programming) then it is not decidable therefore it does not admit a quantifier elimination method.
- Consider the following language:

$$L = \{0, 1, integerp, +, >, =, \sim, \wedge, \vee, mod_n, \exists, \forall\}.$$

- The following formula of L cannot be expressed without quantifiers (in L):

$$\phi(x) = (x > 0) \wedge (\exists\xi((\xi \in \mathbb{Z}) \wedge (0 < 2(x - \xi) < 1)).$$

- But it is equivalent to a quantifier free formula in the following $L' = L \cup \{[\cdot]\}$ language:

$$\phi'(x) = (x > 0) \wedge (0 < 2(x - [x]) < 1).$$

Quantifier elimination algorithms

- Some rewriting precedes most algorithms, for example to put the formula in prenex form.
- The quantifiers are usually eliminated one by one (inside out).
- Aggressive simplification between the elimination steps is necessary to ensure the efficiency of the algorithm.
- Some algorithms are partial, that is they place further restrictions on the formula.
- Cylindrical Algebraic Decomposition implemented in QEPCAD.

Applications - Geometry theorem proving

- Theorems of elementary geometry are considered as a test case for automatic theorem proving.
- Algebraic techniques have been developed: Wu-Ritt method, Gröbner basis techniques, complex elimination methods.
- The geometrical assertion is translated to an algebraic statement via a suitable chosen coordinate system.

$$\bigwedge_{i=1}^{l} f_i(x_1, x_2, \ldots, x_k) = 0 \rightarrow f_0(x_1, x_2, \ldots, x_k) = 0.$$

- The above mentioned techniques prove theorems in the field of complex numbers, therefore they cannot succeed if the equivalent of a theorem of plane geometry does not hold in the complex case.
- The quantifier elimination method does not have this restriction.

Applications - Geometry theorem proving

(continued)

- A. Dolzmann, T.Sturm, V. Weispfenning: A new approach for automatic theorem proving in real geometry (JAR 1998). Implemented in REDLOG.
- Elimination of quantifiers when the polynomials are quadratic.
- Higher exponents are dealt with by the use of implicit factorization and the so-called shifting: divide exponents with their greatest common divisors.
- Method is not complete; might easily exceed available resources; choice of coordinate system might be crucial.
- Examples:
  - Two lines intersect in one and only one point. The first line is the x-axis, the second is $mx + b$:

    $$\exists x(mx + b = 0 \wedge \forall y(y \neq x \rightarrow my + b \neq 0).$$

  - Angle at circumference v. angle at center
  - Qin-Heron's formula
  - Simson's theorem
  - Feuerbach's theorem

Applications - A decidability result

- Automata with dense and integer counters. (Similar concept: timed automata.)
- 'Binary' reachability: Is there an accepting run from a starting state with given counter values to a final state with given counter values?
- The binary reachability of a monotonic dense multi-counter machine is definable by a mixed formula (formulas of L defined above.) Note that monotone counters can be thought of as 'clocks'.
- The satisfiability of mixed formulas are decidable. This is the connection to quantifier elimination: L is shown to be decidable by quantifier elimination.
- It follows that the reachability problem for dense multi-counter machines are decidable when the counters are monotonic.
- The statement is also true if with at most one exception the counters are reversal bounded. (The one exception is called a free counter.)
- Gaoyan Xie et al. : Dense Counter machines and verification problems (CAV 2003)
- The satisfiability algorithm might be used for any machine not just the restricted class above. (But it might not work always.)

Applications - Control systems, feedback design

- Robert S. Boyer, Milton W. Green, J Strother Moore: The Use of a Formal Simulator to Verify a Simple Real Time Control Program
- Control problem: Keep a vehicle on a straight line in the presence of cross wind using the location as sensory input controlling the feedback.
- The wind changes in discrete increments/decrements. A more elaborate model might relax this condition on discreteness, might account for small sensory error and limitations on the engine etc.
- In general we call the process to be controlled a plant, it has a transfer function associated with it. The feedback mechanism is also referred to as a compensator.

Applications - Control systems, feedback design

(continued)

1. For the correction to be efficient (small tracking error) the feedback should be large. (Cruise control.) BUT it can be too large causing adverse effects. (Amplifier.)

2. The design of stable an robust (reliable) feedback systems is the goal.

3. Peter Dorato, Wei Yang, Chaouki Abdallah: Robust Multi-Objective Feedback Design by Quantifier Elimination. (J. Symbolic Computation)

4. The objectives translate to quantified polynomial inequalities. QEPCAD was used to eliminate quantifiers and solve a subset of problems.