Proof Support for Hybrid Systems Verification, I

Introduction to MetiTarski

Lawrence C Paulson, Computer Laboratory, University of Cambridge

Special function problems

$$0 < t \wedge 0 < v_f \implies ((1.565 + .313 v_f)\cos(1.16t) \\ + (.01340 + .00268 v_f)\sin(1.16t))\exp(-1.34t) \\ - (6.55 + 1.31 v_f)\exp(-.318t) + v_f + 10 \geq 0$$

$$0 \leq x \wedge x \leq 289 \wedge s^2 + c^2 = 1 \implies \\ 1.51 - .023\exp(-.019x) - (2.35c + .42s)\exp(.00024x) > -2$$

Both arising from **hybrid systems verification**

Both involving *inequalities*: safety constraints

And both proved *automatically*, in seconds!

An approach to prove such inequalities

1. **Upper and lower bounds** exist for many special functions.

2. These bounds are typically polynomials or *rational functions* (polynomial fractions).

3. *The theory of real polynomial inequalities is decidable.* (Real Closed Fields, or RCF)

These problems can be reduced to
decidable formulas!

Decidability of RCF via quantifier elimination

$$\exists x \, [ax^2 + bx + c = 0]$$

$$\Longleftrightarrow$$

$$b^2 \geq 4ac \wedge (c = 0 \vee a \neq 0 \vee \underset{b \neq 0}{\cancel{b^2 > 4ac}})$$

- If formula has no free variables, result is TRUE or FALSE
- Runtime ***doubly exponential*** in the number of quantifiers
- Implemented in QEPCAD, Z3 and the computer algebra systems Maple and Mathematica

Bounds for the function ln

- Based on the *continued fraction* for ln(x+1), which is much more accurate than the Taylor expansion

$$\frac{x-1}{x} \leq \ln x \leq x - 1$$

$$\frac{(1+5x)(x-1)}{2x(2+x)} \leq \ln x \leq \frac{(x+5)(x-1)}{2(2x+1)}$$

- High accuracy requires *high-degree polynomials*

Bounds for the function exp

- Can be got from Taylor series or continued fractions.
- Some bounds are only good on a *restricted interval*. We need several bounds to cover more of the real line.
- Here are a few of the possibilities for exp.

$$\exp(x) \geq 1 + x + \cdots + x^n / n! \qquad (n \text{ odd})$$

$$\exp(x) \leq 1 + x + \cdots + x^n / n! \qquad (n \text{ even}, x \leq 0)$$

$$\exp(x) \leq 1 / (1 - x + x^2/2! - x^3/3!) \qquad (x < 1.596)$$

Other bounds for functions

sqrt: based on Newton's method

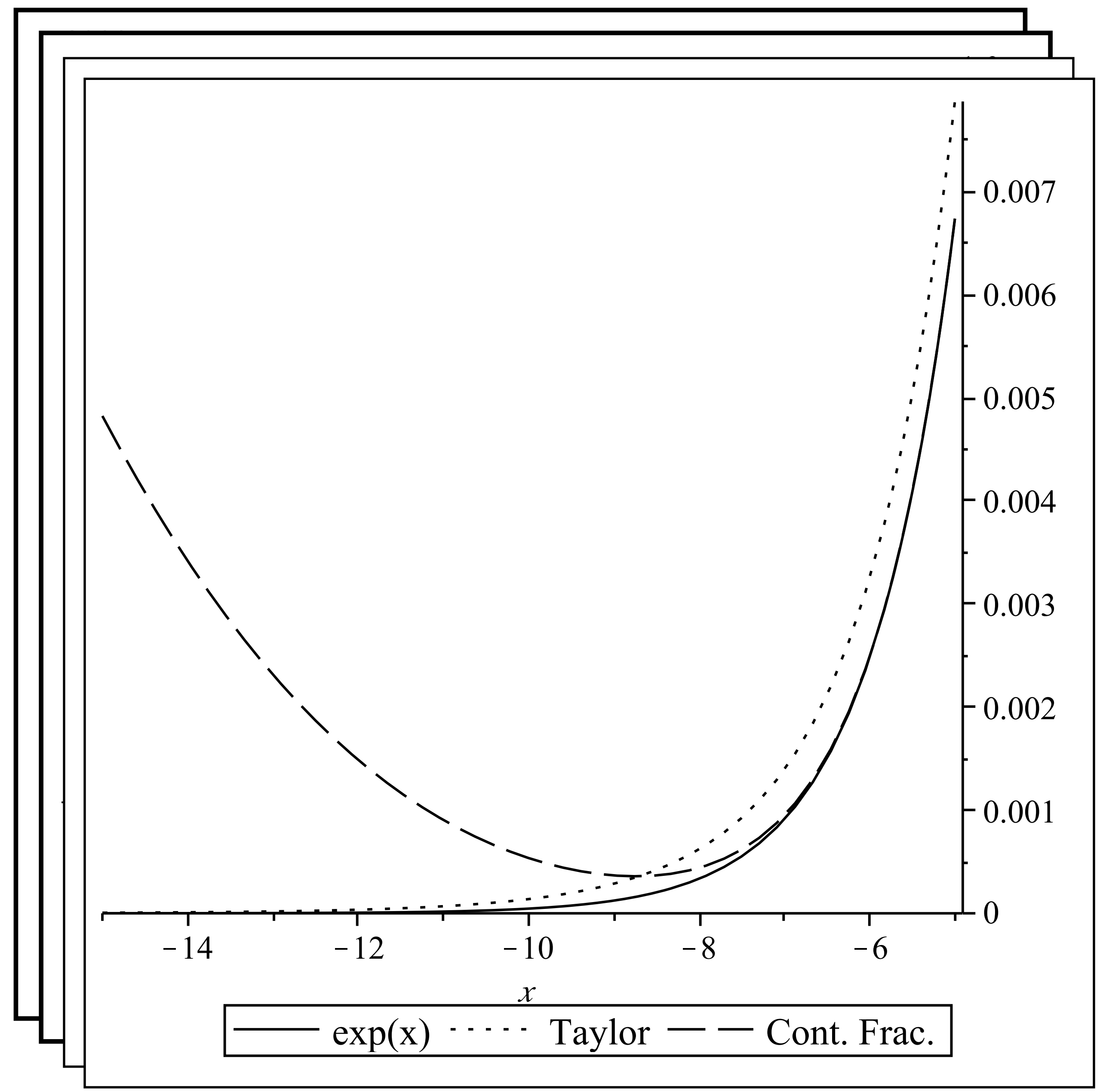
sin & cos: based on Taylor series

$\tan^{-1}$: based on continued fractions

$\sin^{-1}$: based on its Maclaurin series

Limitations and quirks of bounds

- Some are extremely accurate at first, but veer away drastically.
- Our upper bounds for the exponential function have limited ranges!



Tasks for a theorem proving algorithm

- ***Replace*** each occurrence of a function by a upper or lower bound, depending on the context
- ***Select*** the best bounds, balancing simplicity against accuracy
- ***Combine*** bounds valid for short intervals to cover longer intervals (by *case analysis*) and proving *side conditions*
- ***Reason*** in first-order logic using a suitable calculus

How do we do all this?

Um… hack a resolution prover?

Resolution theorem proving: refresher course

- Based on a **calculus** for **pure** first-order logic
- Proves theorems by *contradiction*
- Works with a set of disjunctions (*clauses*), viewed as a giant conjunction, and tries to **saturate** this set
- High-performance implementations, e.g. Vampire
- A clean but efficient implementation (Metis), coded in **Standard ML**.

A baby resolution example

Axioms

$$\forall x [P(x) \implies R(x) \vee Q(f(x))]$$

$$\forall x [R(x) \implies Q(x)]$$

$$\forall x [R(x) \vee f(x) = x]$$

Conjecture (to be negated)

$$\forall x \, [P(x) \implies Q(x)]$$

Resolution example in clause form (disjunctions)

$$\neg P(x) \vee R(x) \vee Q(f(x))$$

$$\neg R(x) \vee Q(x)$$

$$R(x) \vee f(x) = x$$

$$P(a) \quad \neg Q(a)$$

And a resolution proof

conclusion	premises
$R(a) \vee Q(f(a)):$	$P(a),\ \neg P(x) \vee R(x) \vee Q(f(x))$
$\neg R(a):$	$\neg Q(a),\ \neg R(x) \vee Q(x)$
$Q(f(a)):$	
$f(a) = a:$	$R(x) \vee f(x) = x$
$Q(a):$	
$\square:$	$\neg Q(a)$

Could resolution work for special function problems?

Possible meanings of a lower bound

$$1 + x \leq \exp x$$

This yields *four* different implications!

$$\exp x < y \implies 1 + x < y$$
$$\exp x \leq y \implies 1 + x \leq y$$
$$y < 1 + x \implies y < \exp x$$
$$y \leq 1 + x \implies y \leq \exp x$$

Encoding a lower bound for resolution

Express $1 + x \leq \exp x$

as the two clauses

$$y \leq \exp x \vee \neg(y \leq 1 + x)$$
$$\neg(\exp x \leq y) \vee 1 + x \leq y$$

there's a trick to do this
without repeating ourselves

And let $x < y$ abbreviate $\neg(y \leq x)$

Axioms to eliminate division

$$\neg(X \le Y \cdot Z) \vee X/Z \le Y \vee Z \le 0$$

$$\neg(X \le Y/Z) \vee X \cdot Z \le Y \vee Z \le 0$$

$$\neg(X \cdot Z \le Y) \vee X \le Y/Z \vee Z \le 0$$

$$\neg(X/Z \le Y) \vee X \le Y \cdot Z \vee Z \le 0$$

Special care needed, or these will eliminate multiplication instead!

Axioms defining the absolute value function

$$\neg(0 \le x) \vee |x| = x \qquad 0 \le x \vee |x| = -x$$

then from $\exp|c| < y$ deduce *two* consequences:

$$0 \le c \vee \exp(-c) < y$$
$$c < 0 \vee \exp(c) < y$$

The inference rule that accomplishes this is called <u>paramodulation</u>

Necessary modifications to resolution

- *Algebraic simplification* to canonical form, e.g. to identify

$$x^2 + x \quad x + x^2 \quad x(x+1)$$

- … and designed to **isolate** occurrences of functions
- A special inference rule to *split up products*
- ***Integration with the RCF decision procedure:***

  Use it to *delete* any literal that is inconsistent with known algebraic facts

A simple proof: $\forall x \; |e^x - 1| \leq e^{|x|} - 1$

negating the claim

$$e^{|c|} < 1 + |e^c - 1|$$

absolute value (neg)

$$0 \leq c \vee e^{-c} < 1 + |e^c - 1|$$

absolute value (neg)

$$1 \leq e^c \vee 0 \leq c \vee e^{-c} < 2 - e^c$$

lower bound: $1 - c \leq e^{-c}$

$$1 \leq e^c \vee 0 \leq c \vee e^c < 1 + c$$

lower bound: $1 + c \leq e^c$

$$1 \leq e^c \vee 0 \leq c$$

absolute value (pos)

$$e^{|c|} < e^c \vee e^c < 1$$

lower bound: $1 + c \leq e^c$

$$e^{|c|} < e^c \vee c < 0$$

absolute value, etc.

$$c < 0$$

$$1 \leq e^c$$

upper bound (complicated!)

$$\square$$

```
SZS output start CNFRefutation for abs-problem-11.tptp
cnf(lgen_le_neg, axiom, (X <= Y | ~ lgen(0, X, Y))).

cnf(leq_le

cnf(exp_lo
    (X < -

cnf(exp_lo
    (~ lge


     lgen(

cnf(exp_up
    (0 < X
     ~ lge
     lgen(

cnf(abs_no

cnf(abs_ne

fof(abs_pr
    (! [X]

fof(subgoa
    infere

fof(negate
    infere

fof(normal
    infere

fof(normal
    infere
```

```
cnf(refute_0_6, plain, (exp(abs(skoXC1)) - 1 < abs(exp(skoXC1) - 1)),
    inference(canonicalize, [], [normalize_0_1])).

cnf(refute_0_7, plain, (exp(ab
    inference(arithmetic, [],

cnf(refute_0_8, plain, (0 <= s
    inference(subst, [], [abs_

cnf(refute_0_9, plain,
    (exp(-skoXC1) < 1 + abs(-1
     1 + abs(-1 + exp(skoXC1))
    introduced(tautology, [equ

cnf(refute_0_10, plain,
    (exp(-skoXC1) < 1 + abs(-1
     1 + abs(-1 + exp(skoXC1))
    inference(resolve, [], [re

cnf(refute_0_11, plain,
    (exp(-skoXC1) < 1 + abs(-1
    inference(resolve, [], [re

cnf(refute_0_12, plain,
    (-1 + exp(skoXC1 * -1) < a
    inference(arithmetic, [],

cnf(refute_0_13, plain,
    (0 <= -1 + exp(skoXC1) | a
    inference(subst, [], [abs_

cnf(refute_0_14, plain,
    (-1 + exp(skoXC1 * -1) < -
     abs(-1 + exp(skoXC1)) !=
     abs(-1 + exp(skoXC1)) <=
    introduced(tautology, [equ
```

```
cnf(refute_0_70, plain,
    (skoXC1 *
     (-2304 +
      skoXC1 *
      (1152 +
       skoXC1 * (-384 + skoXC1 * (84 + skoXC1 * (-12 + skoXC1))))) <=
     -2304 |
     skoXC1 *
     (4608 +
      skoXC1 *
      (-3456 +
       skoXC1 *
       (1536 +
        skoXC1 * (-468 + skoXC1 * (96 + skoXC1 * (-13 + skoXC1)))))) <= 0),
    inference(decision, [], [refute_0_37])).

cnf(refute_0_71, plain,
    (skoXC1 *
     (-2304 +
      skoXC1 *
      (1152 +
       skoXC1 * (-384 + skoXC1 * (84 + skoXC1 * (-12 + skoXC1))))) <=
     -2304), inference(resolve, [], [refute_0_70, refute_0_69])).

cnf(refute_0_72, plain,
    (-2304 <
     skoXC1 *
     (-2304 +
      skoXC1 *
      (1152 + skoXC1 * (-384 + skoXC1 * (84 + skoXC1 * (-12 + skoXC1)))))),
    inference(decision, [], [refute_0_37])).

cnf(refute_0_73, plain, ($false),
    inference(resolve, [], [refute_0_71, refute_0_72])).
SZS output end CNFRefutation for abs-problem-11.tptp
```

Summary

Special function problems can be solved, replacing functions by *algebraic upper or lower bounds*.

We end up with *polynomial inequalities*: in other words, RCF problems

... and first-order formulae involving $+$, $-$, $\times$ and $\leq$ (on reals) are **decidable**.

An *RCF decision procedure* and *resolution theorem proving* are the core technologies.