MetiTarski: Past and Future

Prof. Lawrence C Paulson, University of Cambridge

Interactive Theorem Proving, 13–15 August, 2012

Díd you know?

Over the real numbers,

non-linear arithmetic is...

decidable

We can decide statements involving +, −, ×!

And that can be harnessed to prove statements involving

sin, cos, exp, ln, ...!!

MetiTarski: a resolution theorem prover for the real numbers

- proves first-order statements involving functions such as exp, ln, sin, cos, $\tan^{-1}$, …

Text

- using axioms bounding these functions by rational functions

- … and heuristics to isolate and remove function occurrences

- integrated with the RCF* decision procedures QEPCAD, Mathematica, Z3

*RCF (*real-closed field*): any field that's "first-order" equivalent to the reals

some theorems that MetiTarski can prove

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

$$0 \leq x \wedge x \leq 1.46 \times 10^{-6} \implies (64.42\sin(1.71 \times 10^6 x) - 21.08\cos(1.71 \times 10^6 x))e^{9.05\times 10^5 x} + 24.24 e^{-1.86\times 10^6 x} > 0$$

$$0 \leq x \wedge 0 \leq y \implies y\tanh(x) \leq \sinh(yx)$$

Each is proved in a few seconds!

What about the decidability of real arithmetic?

- Tarski (1948): every first-order RCF formula can be replaced by an equivalent, *quantifier-free* one.
- *Quantifier elimination* implies the decidability of RCF
- … and also the decidability of Euclidean geometry.

real quantifier elimination: a well-known example

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

$$\Longleftrightarrow$$

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

$$b \neq 0$$

The equivalent quantifier-free formula can be messy…

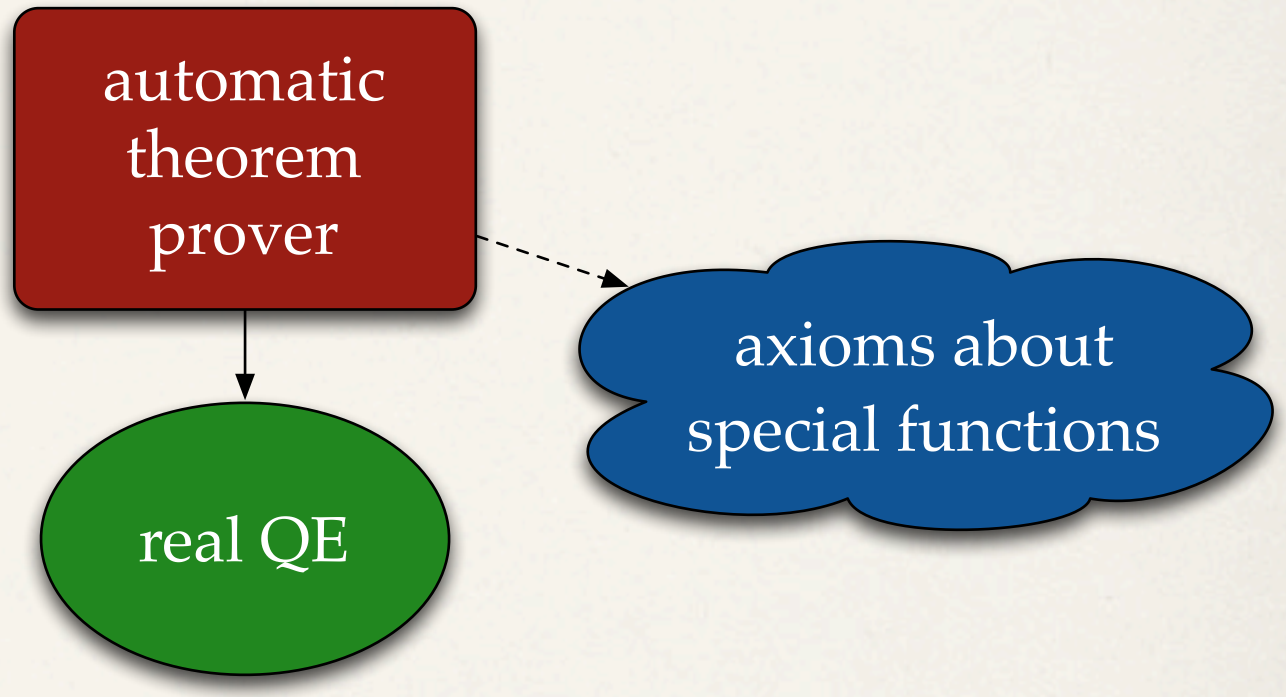
real QE is expensive!

- Tarski's algorithm has *non-elementary* complexity! There are usable algorithms by Cohen, Hörmander, etc.
- The key approach: *cylindrical algebraic decomposition* (Collins, 1975)
- But quantifier elimination can yield a **huge** quantifier-free formula
- … *doubly exponential* in the number of quantifiers (Davenport and Heintz, 1988)

No efficient algorithm can exist. Do we give up?
Of course not…

let's combine real QE with theorem proving

- To prove statements involving real-valued special functions.
- This *theorem-proving* approach delivers machine-verifiable evidence to justify its claims.
- Based on heuristics, it often finds proofs—but with no assurance of getting an answer.
- Real QE will be called as a **decision procedure**.



Given the cost of real QE, isn't this stupid?

- High complexity does not imply uselessness—as with the boolean satisfiability (SAT) problem
- … or higher-order unification, the (semi-decidable) basis of Isabelle.
- This is *fundamental research*. Theorem proving for real-valued functions has been largely unexplored.

the basic idea

Our approach involves replacing functions by *rational function upper or lower bounds*.

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

… and first-order formulae involving +, −, × and ≤ (on reals) are **decidable**.

Real QE and *resolution theorem proving* are the core technologies.

a simple proof:

$$\forall x \; |e^x - 1| \leq e^{|x|} - 1$$

negating the claim

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

absolute value

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

absolute value

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

lower bound: 1-c ≤ e^{-c}

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

lower bound: 1+c ≤ e^c

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

0 ≤ c ⇒ 1 ≤ e^c

$$1 \leq e^c$$

absolute value

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

absolute value, etc.

$$c < 0$$

$$\square$$

the key to the integration: *algebraic literal deletion*

- A list of RCF clauses (algebraic, with no variables) is maintained.
- Every literal of each new clause is examined.
- A literal will be *deleted* if—according to the decision procedure—it is *inconsistent* with its context.
- MetiTarski also uses the decision procedure to detect *redundant* clauses (those whose algebraic part is deducible from known facts).

examples of literal deletion

- *Unsatisfiable* literals such as $p^2 < 0$ are deleted.
- If $x(y+1) > 1$ has previously been deduced, then $x=0$ will be deleted.
- The context includes the *negations of adjacent literals* in the clause:
  $z > 5$ is deleted from $z^2 > 3 \vee z > 5$
- … because quantifier elimination reduces $\exists z\ [z^2 \leq 3 \wedge z > 5]$ to FALSE.

some bounds for ln

- based on the continued fraction for ln(x+1)
- *much* more accurate than the Taylor expansion
- Simplicity can be exchanged for accuracy.
- With these, the maximum degree we use is 8.

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

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

bounds for other functions

- a mix of *continued fraction* approximants and truncated *Taylor series*, etc, modified to suit various argument ranges and accuracies
- a tiny bit of **built-in knowledge** about signs, for example, $\exp(x) > 0$
- NO fundamental mathematical knowledge, for example, the geometric interpretation of trigonometric functions
- MetiTarski can reason about any function that has well-behaved *upper and lower bounds* as rational functions.

Have these bounds been proved correct? Some have, some haven't.

introducing the RCF solvers

QEPCAD (Hoon Hong, C. W. Brown et al.)
Venerable. Very fast for univariate problems.

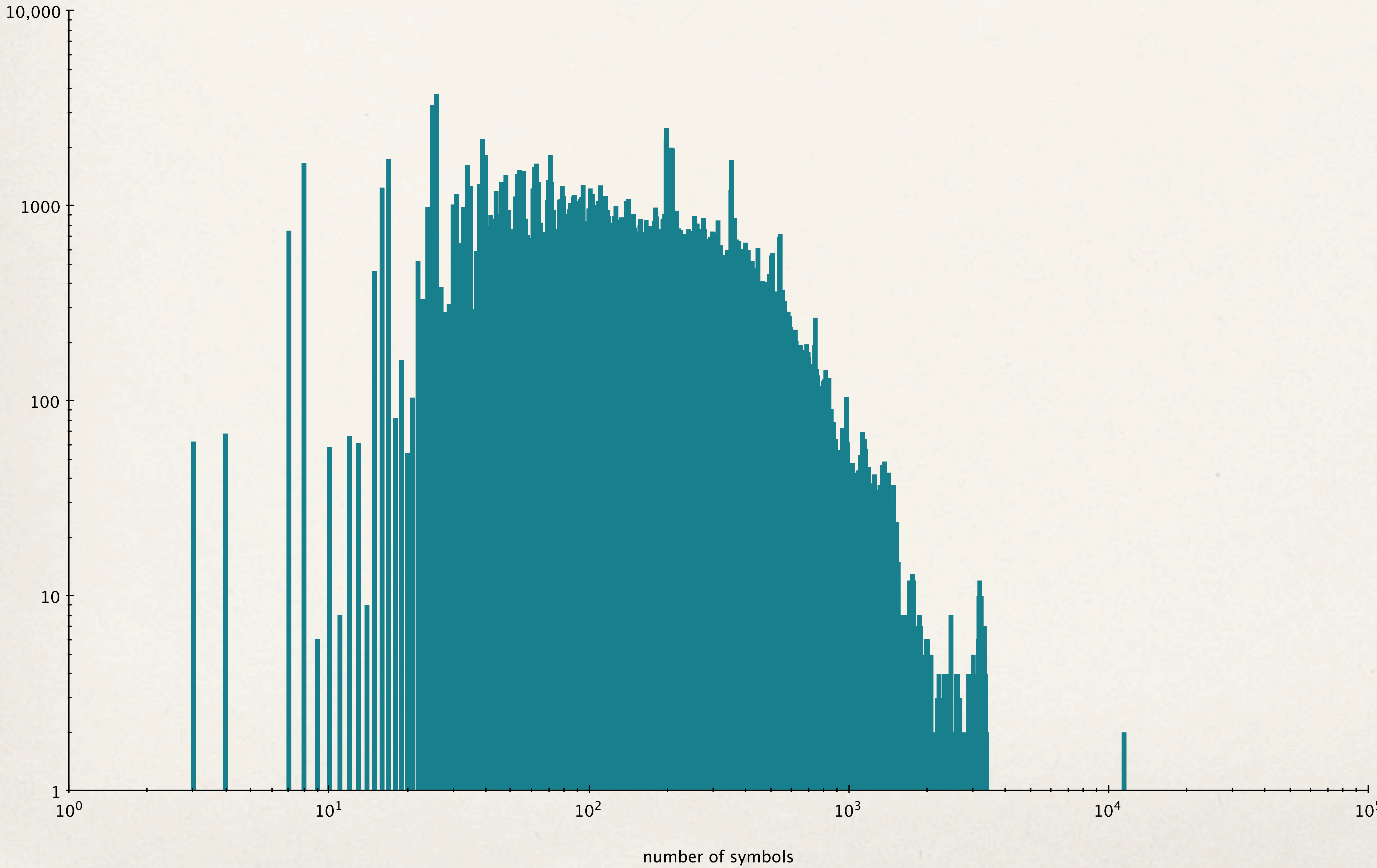
Mathematica (Wolfram research)
Much faster than QEPCAD for 3–4 variables

Z3 (de Moura, Microsoft Research)
An SMT solver with non-linear reasoning.

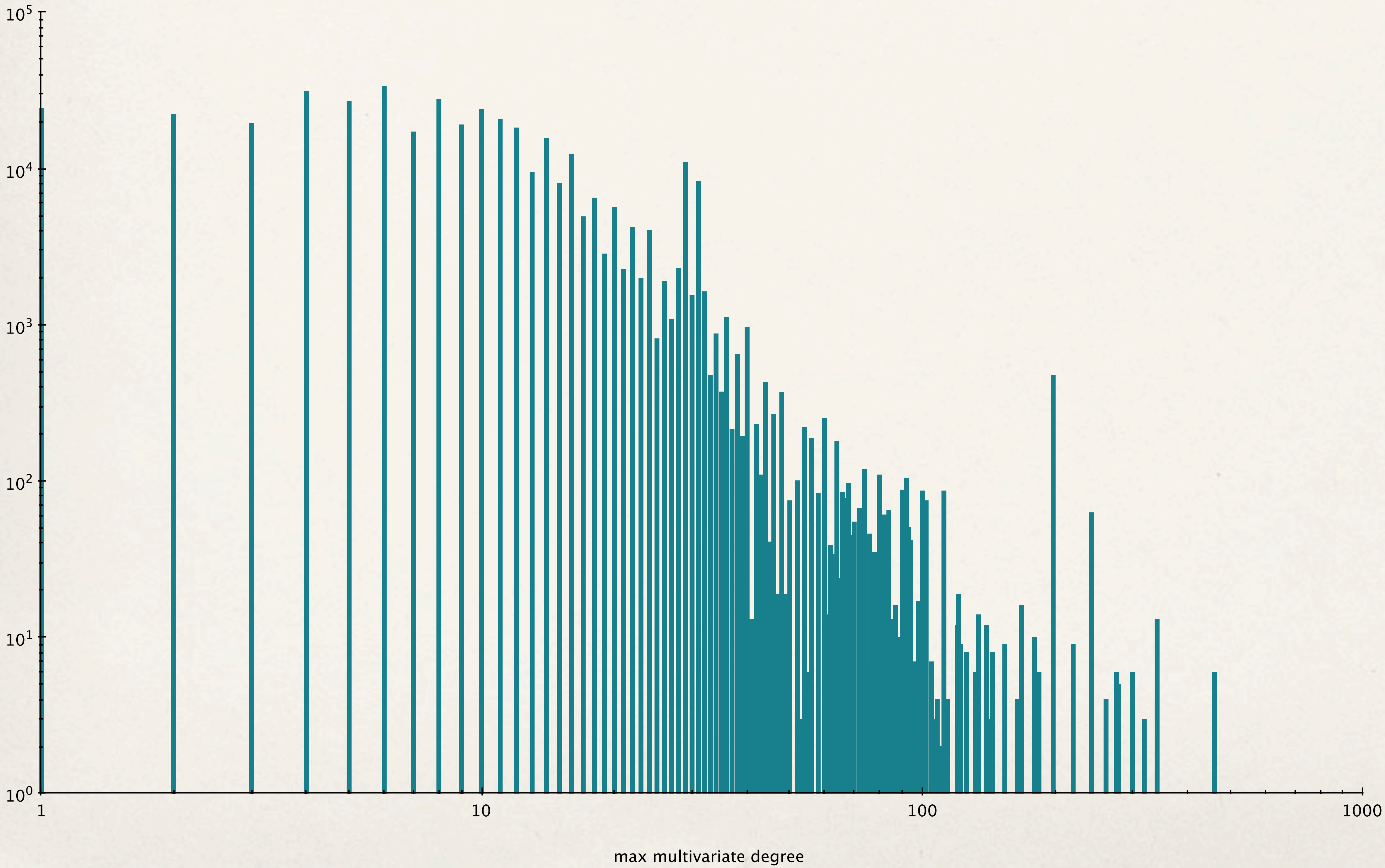
statistics about the RCF problems

- 400,000 RCF problems generated from 859 MetiTarski problems.
- Number of *symbols*: in some cases, 11,000 or more!
- Maximum *degree*: up to 460!
- But… number of *variables*? Typically just 1. *Very few* above 8.

distribution of problem sizes (in symbols)



distribution of polynomial degrees (multivariate)



a heuristic: *model sharing*

- MetiTarski applies QE only to existential formulas, $\exists x\ \exists y$ …
- Many of these turn out to be satisfiable,…
- and many satisfiable formulas have the *same model*.
- By maintaining a list of "successful" models, we can show many RCF formulas to be satisfiable **without performing QE**.

… because most of our RCF problems are satisfiable…

Problem	All RCF		SAT RCF		% SAT	
	#	secs	#	secs	#	secs
CONVOI2-sincos	268	3.28	194	2.58	72%	79%
exp-problem-9	1213	6.25	731	4.11	60%	66%
log-fun-ineq-e-weak	496	31.50	323	20.60	65%	65%
max-sin-2	2776	253.33	2,221	185.28	80%	73%
sin-3425b	118	39.28	72	14.71	61%	37%
sqrt-problem-13-sqrt3	2031	22.90	1403	17.09	69%	75%
tan-1-1var-weak	817	19.5	458	7.60	56%	39%
trig-squared3	742	32.92	549	20.66	74%	63%
trig-squared4	847	45.29	637	20.78	75%	46%
trigpoly-3514-2	1070	17.66	934	14.85	87%	84%

In one example, 2172 of 2221 satisfiable RCF problems can be settled using model sharing, with only 37 separate models.

introducing Strategy 1

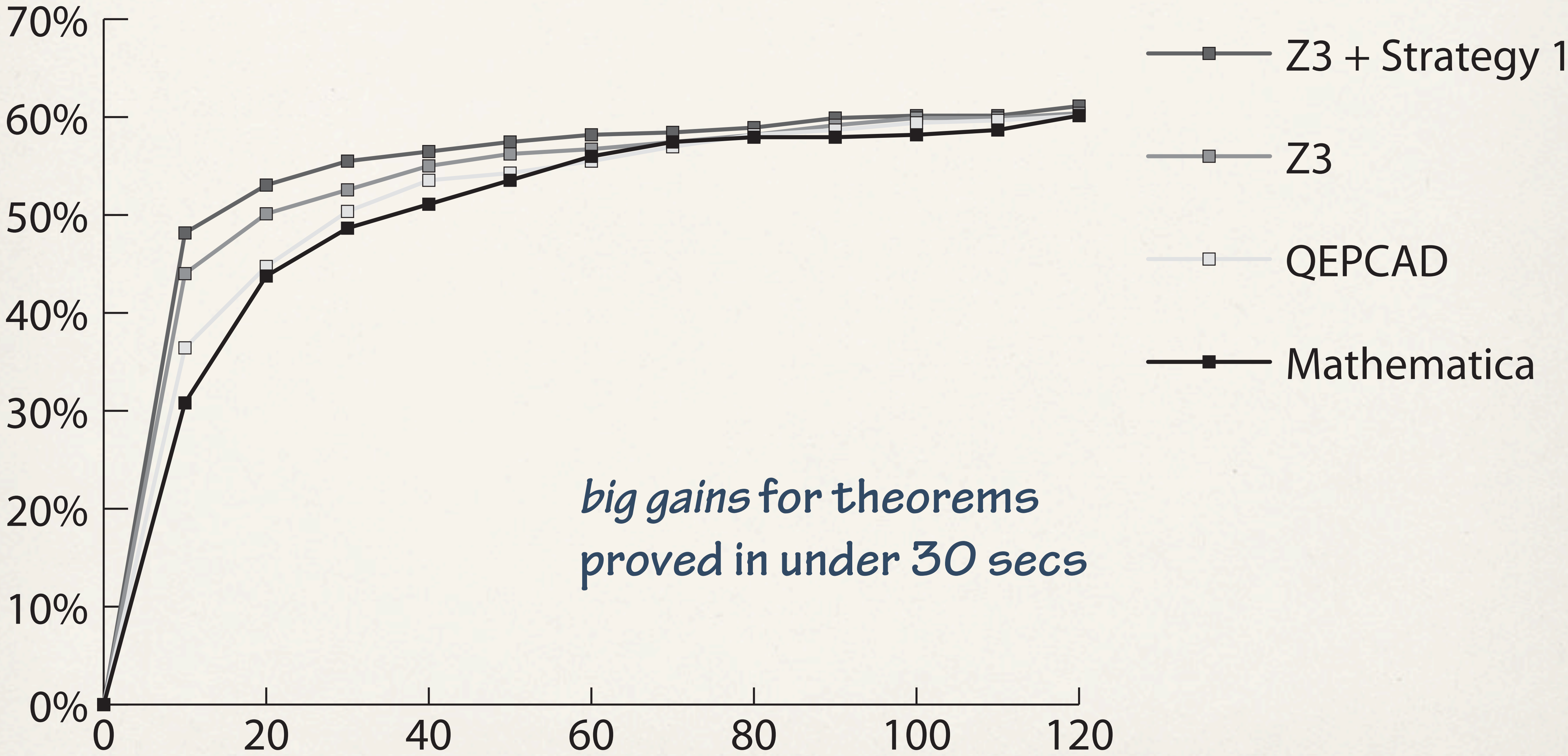
model sharing

+

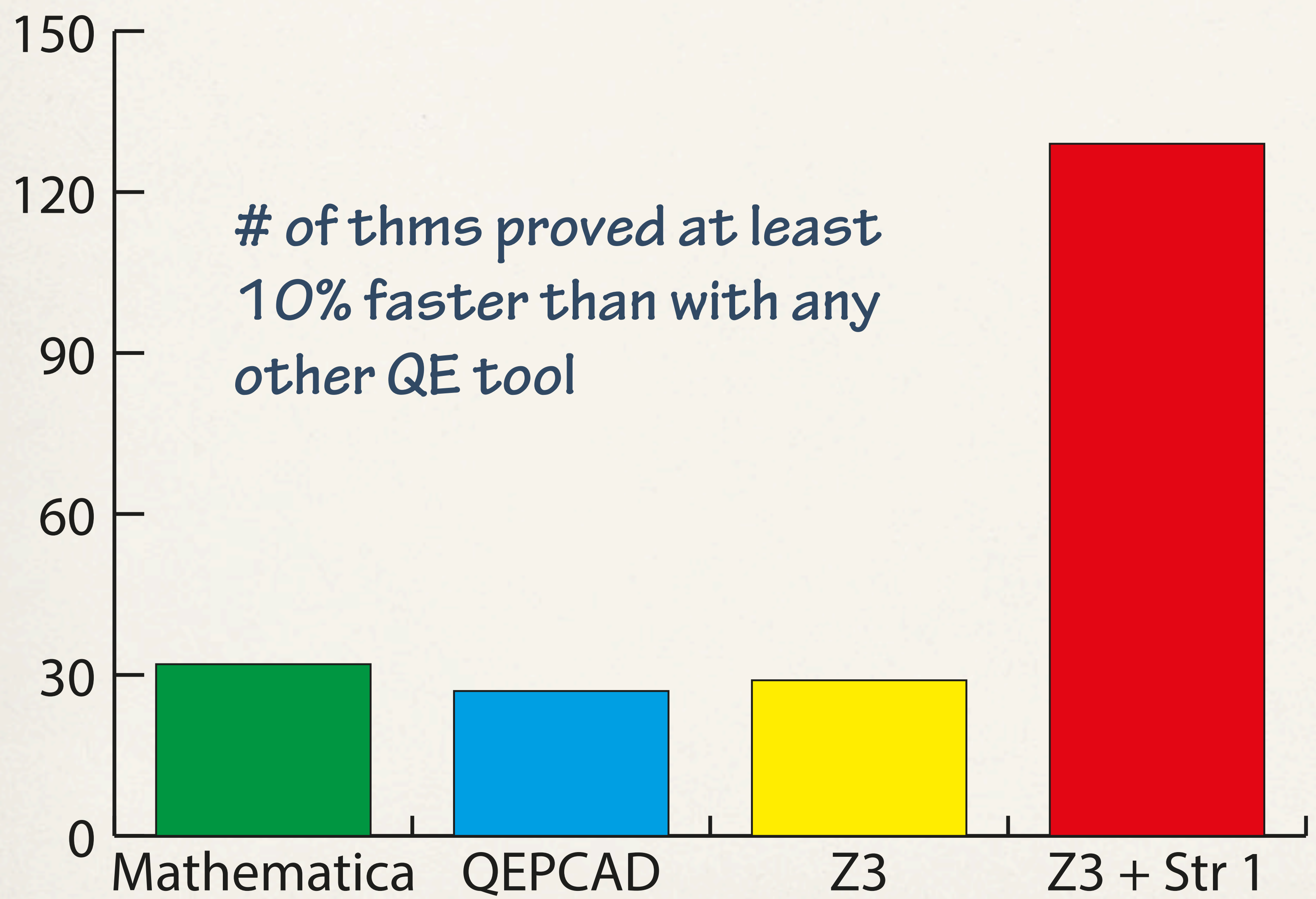
omitting the standard test for *irreducibility*

= Strategy 1

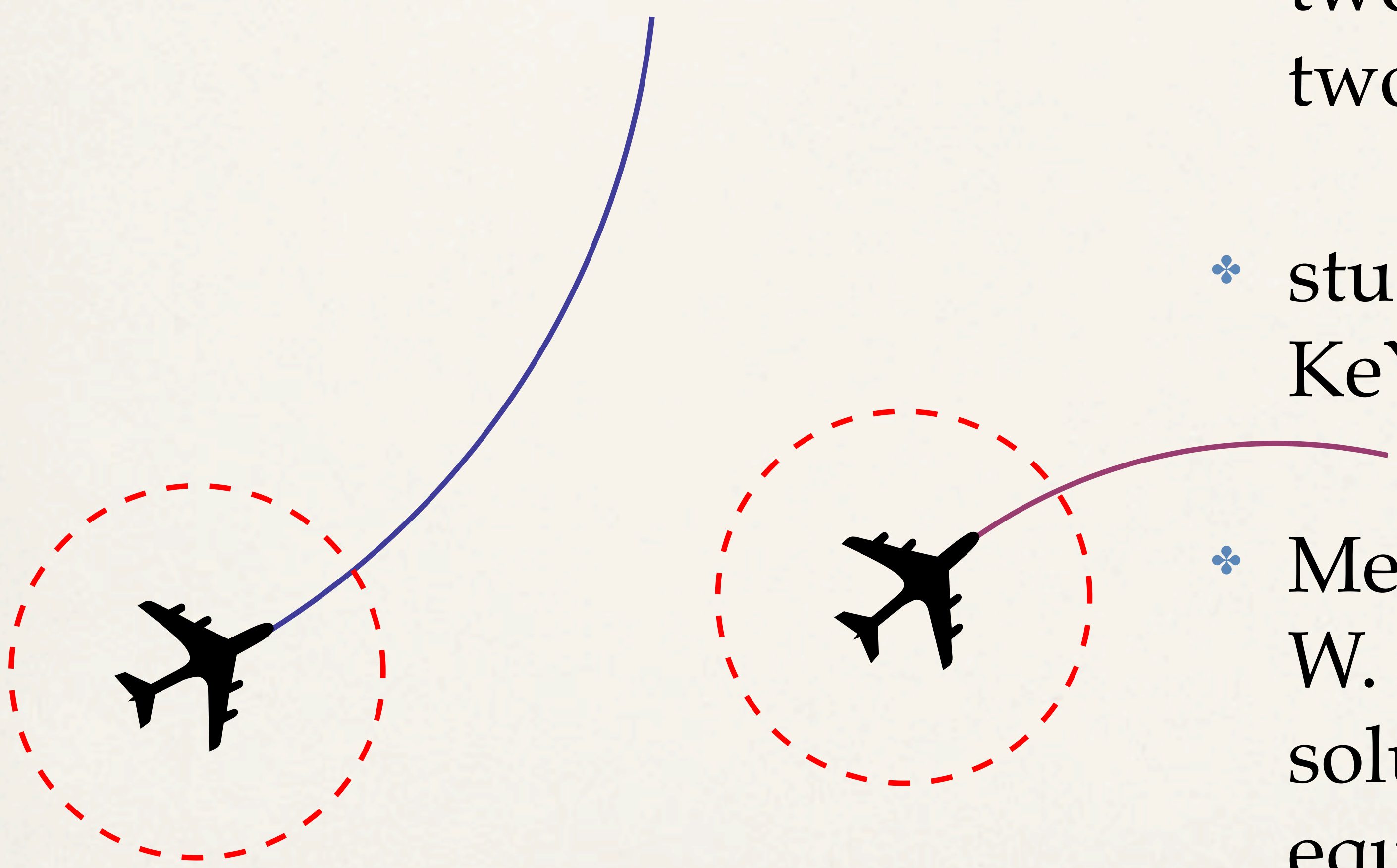
comparative results
(% proved in up to 120 secs)



Strategy 1 finds the fastest proofs



a collision avoidance problem



- two aircraft, *x* and *y*, flying in two dimensions (for simplicity)
- studied by Platzer (2010), using KeYmaera
- MetiTarski treatment due to W. Denman, using closed-form solutions of the differential equations of motion

The system of differential equations for aircraft x

x_1 denotes *position* in the first coordinate;
d_1 denotes *velocity*

$$x_1'(t) = d_1(t) \quad x_2'(t) = d_2(t) \quad d_1'(t) = -\omega d_2(t) \quad d_2'(t) = \omega d_1(t)$$
$$x_1(0) = x_{1,0} \quad x_2(0) = x_{2,0} \quad d_1(0) = d_{1,0} \quad d_2(0) = d_{2,0}$$

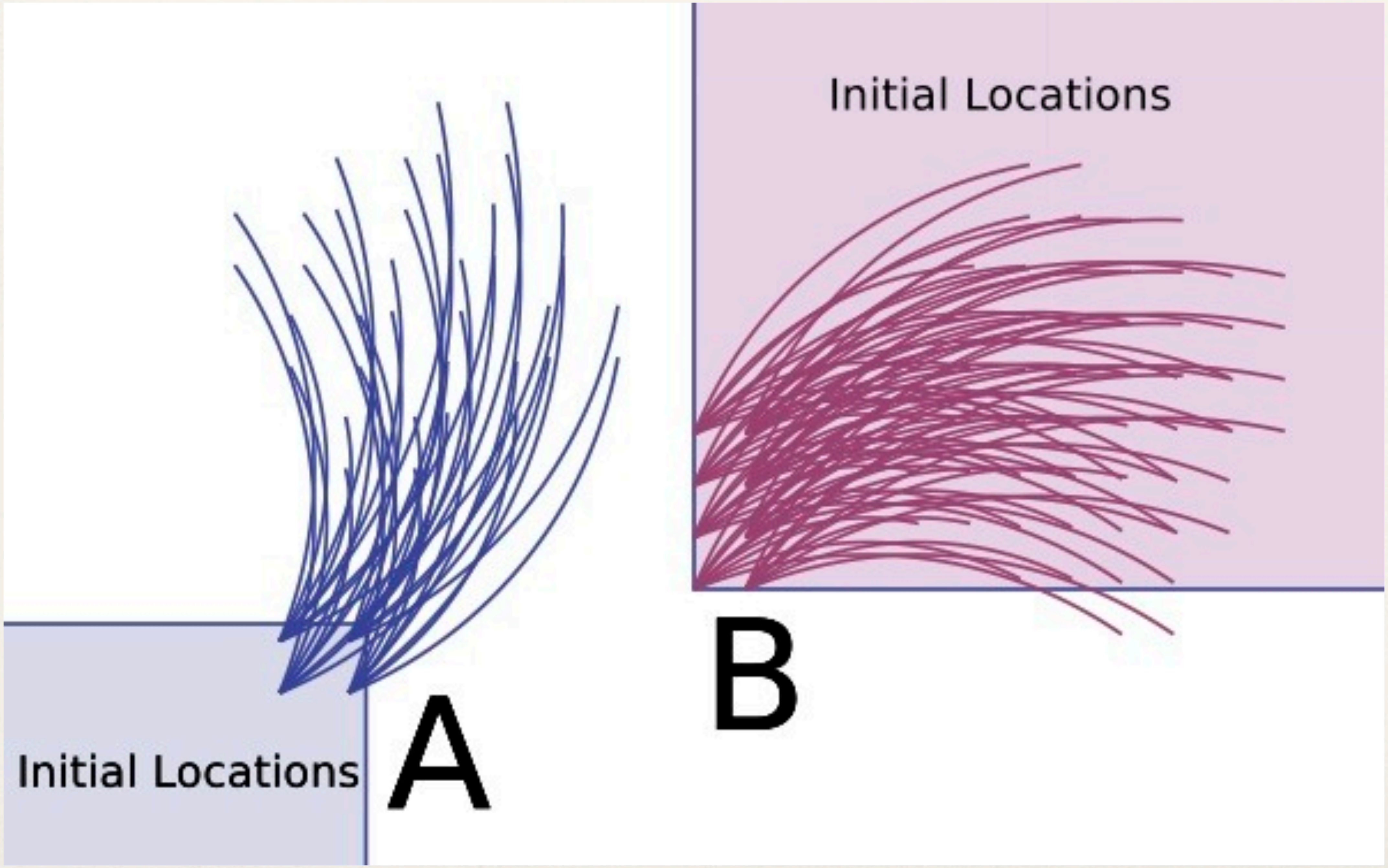
x_2 denotes *position* in the **second** coordinate;
d_2 denotes *velocity*

… and the closed-form solution

$$x_1(t) = x_{1,0} + \frac{d_{2,0}\cos(\omega t) + d_{1,0}\sin(\omega t) - d_{2,0}}{\omega}$$

$$x_2(t) = x_{2,0} - \frac{d_{1,0}\cos(\omega t) - d_{2,0}\sin(\omega t) - d_{1,0}}{\omega}$$

possible paths of the two aircraft



the desired safety property

Two aircraft following those equations…

subject to certain other parameters…

must maintain a *safe distance*, p:

$$(x_1(t) - y_1(t))^2 + (x_2(t) - y_2(t))^2 > p^2$$

the resulting MetiTarski problem

```
fof(airplane_easy,conjecture,
  (! [T,X10,X20,Y10,Y20,D10,D20,E10,E20] :
    (
      ( 0 < T & T < 10 & X10 < -9 & X20 < -1 & Y10 > 10 & Y20 > 10 &
        0.1 < D10 & D10 < 0.15 & 0.1 < D20 & D20 < 0.15 &
        0.1 < E10 & E10 < 0.15 & 0.1 < E20 & E20 < 0.15 )
      =>
      ( (X10 - Y10 - 100*D20 - 100*E20 + (100*D20 + 100*E20)*cos(0.01*T)
        + (100*D10 - 100*E10)*sin(0.01*T))^2 +
        (X20 - Y20 + 100*D10 + 100*E10 + (-100*D10 - 100*E10)*cos(0.01*T)
        + (100*D20 - 100*E20)*sin(0.01*T))^2 )
      > 2 )
  )
).
include('Axioms/general.ax').
include('Axioms/sin.ax').
include('Axioms/cos.ax').
```

remarks about this proof

- 9 variables!
- originally required 924 seconds (using Z3)
- can take as little as 30 seconds, depending on configuration

other possible applications

- *hybrid systems*, especially those involving transcendental functions
- showing stability of dynamical systems using *Lyapunov functions*
- real error analysis…?
- any application involving *ad hoc* real inequalities

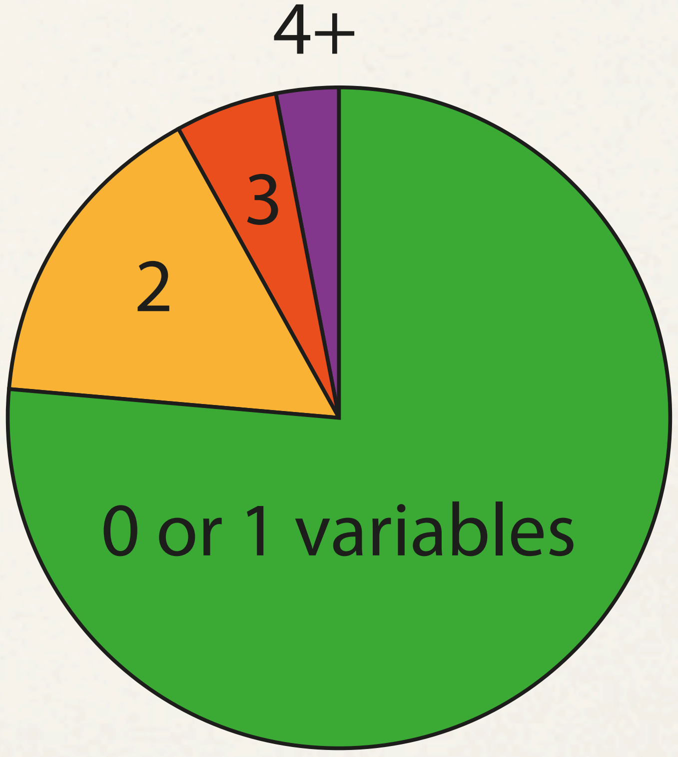
We are still looking…

inherent limitations

- Only non-sharp inequalities can be proved.
- Few MetiTarski proofs are mathematically elegant.
- Problems involving **nested** function calls can be very difficult.

research challenges

- Real QE is still much too slow! It's usually a serious bottleneck.
- We need to handle many more variables!
- Upper/lower bounds sometimes need *scaling* or *argument reduction*: how?
- How can we set the numerous options offered by RCF solvers?



conclusions

- MetiTarski really works on some very hard problems!
- We are continually working on both improvements and applications.
- Performance (especially of real QE) remains a challenge.
- Our main goal: to handle problems involving more variables.

the Cambridge team


James Bridge


Grant Passmore


Zongyan Huang


William Denman

acknowledgements

- *Edinburgh*: Paul Jackson; *Manchester*: Eva Navarro
- Assistance from C. W. Brown, A. Cuyt, I. Grant, J. Harrison, J. Hurd, D. Lester, C. Muñoz, U. Waldmann, etc.
- Behzad Akbarpour formalised most of the engineering examples.
- The research was supported by the Engineering and Physical Sciences Research Council [grant numbers EP/C013409/1,EP/I011005/1,EP/I010335/1].

EPSRC

Engineering and Physical Sciences Research Council