Decision procedures, heuristic procedures, and formally verified mathematics

Jeremy Avigad
(joint work with Ed Dean, Kevin Donnelly, Harvey Friedman, and John Mumma)

Department of Philosophy and Department of Mathematical Sciences
Carnegie Mellon University
(currently visiting the INRIA-MSR Joint Research Centre in Orsay)

March, 2010

Overview

The prospect of formally verified mathematics raises the question as to what kinds of mathematical inferences can be automated, and how.

In this talk, I will discuss some decidability results loosely related to specific verification problems:

- big O reasoning
- real-valued inequalities
- Euclidean diagrammatic reasoning

Caveats

Note the decidability results do not bear directly on practical problems:

- A decision procedure that runs too slowly on the examples you care about is worthless.
- An unprincipled hack that gets all your inferences is fine.
- Some undecidability results rely on coding and contrived examples that never come up in practice.
- The set of inferences that can be verified in ZFC with less than 10^{100} symbols is decidable (in constant time!).

I will speculate on practical matters at the end.

Big O reasoning

Consider the set of functions from any infinite set S to any ordered ring R.

$f = O(g)$ means $\exists C \, \forall x \, (|f(x)| \leq C|g(x)|)$.

$f = g + O(h)$ means $f - g = O(h)$.

These notions are used widely in mathematics and computer science.

Big O reasoning

Examples:

$$\left.\begin{aligned} f + g &= h + O(k) \\ g + l &= h + O(k) \end{aligned}\right\} \Rightarrow f = l + O(k)$$

and

$$\left.\begin{aligned} f + g &= h + O(k) \\ g &= O(l) \\ k &= O(l) \end{aligned}\right\} \Rightarrow f = h + O(l)$$

Big O reasoning

Take the first-order language with variables $f, g, h, \ldots$, symbols for $0, +, -, \min, \max$, and absolute value, and a ternary relation $f = g + O(h)$.

Intended semantics: functions from an infinite set to an ordered ring.

Theorem. The validity of quantifier-free formulas is decidable.

For the simplicity, I'll focus on nonnegative functions and Horn clauses, and no subtraction within the ”O”.

Big O reasoning

Axioms:

1. $f = g \leftrightarrow f = g + O(0)$
2. $+$ is associative and commutative, with identity 0
3. for fixed h, the relation $f = g + O(h)$ is reflexive, symmetric, and transitive
4. monotonicity: $f = O(f + g)$
5. transitivity: $f = g + O(h) \wedge h = O(k) \rightarrow f = g + O(k)$
6. linearity:
   - 6.1 $f_1 = g_1 + O(h) \wedge f_2 = g_2 + O(h) \rightarrow f_1 + f_2 = g_1 + g_2 + O(h)$
   - 6.2 $f_1 + f_2 = g_1 + g_2 + O(h) \wedge f_1 = g_1 + O(h) \rightarrow f_2 = g_2 + O(h)$
   - 6.3 for each positive integer k, the axiom $kf = kg + O(h) \rightarrow f = g + O(h)$

Big O reasoning

Consequences:

- If $r = O(s)$, then an equation up to $O(r)$ also holds up to $O(s)$
- $f + g = O(h)$ implies $f = O(h)$.
- For any positive $k_1, \ldots, k_m$,

$$O(k_1 f_1 + \ldots + k_m f_m) = O(f_1 + \ldots + f_m),$$

In an equation $r = s + O(t)$, all that is relevant are the variables appearing in t, and the parts of r and s that do not involve variables in t. For example,

$$3f_1 + 2f_2 = 5f_3 + O(f_2 + 3f_4)$$

is equivalent to

$$3f_1 = 5f_3 + O(f_2 + f_4).$$

Big O reasoning

Algorithm: to establish $s = O(t)$, set $t' = t$, and iteratively

1. look for equations of the form $q = O(t')$, where q has positive coefficients.
2. If f is a component of q, then $O(t') = O(t' + f)$. So:
    2.1 (get more equations) replace t' by $t' + f$
    2.2 (simplify equations) henceforth ignore f

Use linear algebra for 1. Step 2 can only occur finitely many times.

The only clever part: use the duality principle from linear programming to show that if this terminates without $s = O(t')$, there is a counterexample.

Big O reasoning

Notes:

- Can extend to arbitrary universal formulas.
- Can lift the restriction to nonnegative functions.
- Can extend to "eventually" reading.
- Can add, e.g., rational coefficients.
- Can add classes of fixed functions – polynomials, exponents, logarithms.

With multiplication, there are additional laws, such as:

$$r_1 = O(s_1) \wedge r_2 = O(s_2) \rightarrow r_1 \cdot r_2 = O(s_1 \cdot s_2).$$

I do not know whether entailment is still decidable, but the (now heuristic) algorithm can certainly be extended.

Real inequalities : an example

Ramsey's theorem tells us that for every k there is an N large enough, so that no matter how one colors the edges of the complete graph on N vertices red and blue, there is a homogeneous subset of size k.

Here is a lower bound on N:

Theorem (Erdös) For all $k \geq 2$, if $N < 2^{k/2}$, there is a coloring of the complete graph on N vertices with no homogeneous subset of size k.

For $k = 2$ and $k = 3$ is it easy to check this by hand.

For $k \geq 4$, show that with nonzero probability, a random coloring has this property.

Real inequalities: an example

For $k \geq 4$, suppose $N < 2^{k/2}$, and suppose we color each edge red with probability $1/2$.

The probability that any given subset of size k is homogeneous is $2^{-\binom{k}{2}+1}$.

So the probability of a homogeneous subset is at most $\binom{N}{k}2^{-\binom{k}{2}+1}$.

But $\binom{N}{k} = \frac{N(N-1)(N-2)\cdots(N-k+1)}{k(k-1)\cdots 1} \leq \frac{N^k}{2^{k-1}}$.

So we have

$$\binom{N}{k}2^{-\binom{k}{2}+1} \leq \frac{N^k}{2^{k-1}}2^{-\binom{k}{2}+1} < 2^{\frac{k^2}{2}-\binom{k}{2}-k+2} = 2^{-\frac{k}{2}+2} \leq 1.$$

Real inequalities

Such “straightforward” reasoning in mathematics is typical.

These particular inferences are quantifier-free. (Mild uses of quantifiers come in with phrases like “sufficiently large,” or “choose $N >> x$.”)

They involve little more than basic arithmetic.

In principle, the theory of real closed fields is decidable. But:

- RCF procedures are slow, and arguably misguided for inferences like these.
- Worse: they do not extend to straightforward inferences with monotone functions, trigonometric functions, exponentiation and logarithm, arbitrary sums and products, etc.

Real inequalities

One idea: work backwards, using, for example,

$$0 < s, 0 < t \Rightarrow 0 < st$$

and

$$0 < s < t \Rightarrow 1/t < 1/s.$$

But backchaining is nondeterministic. For example:

- We also have $s < 0, t < 0 \Rightarrow 0 < st$ and $s < t < 0 \Rightarrow 1/t < 1/s$.
- We can prove $s + t + u < r + v$ by proving $s + u < r$ and $t \leq v$.
- We can also prove $s + t + u < r + v$ by proving $s + u < r + 3$ and $t \leq v - 3$ or by proving $s < (r + v)/2$ and $t + u < (r + v)/2$.

Real inequalities

Next idea: work forwards. For example, from $n \leq (K/2)x$, $0 < C$, and $0 < \varepsilon < 1$, we have

- $C + 3 > 1$
- $3(C + 3) > 1$
- $\frac{\varepsilon}{3(C+3)} < 1$
- $1 + \frac{\varepsilon}{3(C+3)} < 2$

and hence

$$(1 + \frac{\varepsilon}{3(C + 3)}) \cdot n < 2(K/2)x = Kx.$$

But clearly we need some guidance!

Real inequalities

Third idea: combine local procedures.

Theorem. Suppose T_1 and T_2 are “stably infinite” and decidable. Suppose that the languages are disjoint, except for the equality symbol. Then the universal fragment of $T_1 \cup T_2$ is decidable.

In particular, if T_1 and T_2 have only infinite models, they are stably infinite.

This allows you to design decision procedures for individual theories and then put them together.

With additional hypotheses on the source theories, the decision procedures can be made efficient (Nelson-Oppen, Shostak, . . .).

Real inequalities

Theorem. The theory of $(\mathbb{R}, 0, +, <)$ has quantifier-elimination, and so is decidable.

For universal formulas, Fourier-Motzkin is doubly exponential in principle, but works well in practice. More efficient methods are available (e.g. Weispfenning's "test point" method).

Theorem. The theory of $(\mathbb{R}, 1, \cdot, <)$ has quantifier-elimination and so is decidable.

In fact, modulo case splits on the signs of terms, this reduces to the previous theorem.

Corollary. The universal fragment of the union of these two theories is decidable.

Real inequalities

The bad news: the union of the two theories just described doesn't include distributivity.

The good news: many inferences don't need it, except for constants (for example, $3(r + s) = 3r + 3s$).

The bad news: adding symbols for constants, or multiplication by constants, introduces nontrivial overlap between the languages. Nelson-Oppen methods break down.

General question: what happens when you combine local procedures, when the theories have nontrivial overlap?

Real inequalities

Specifically: let $f_a(x) = ax$ for rational constants a.

Let $T_{add}[\mathbb{Q}]$ be the theory of $(\mathbb{R}, 0, 1, +, -, <, \ldots, f_a, \ldots)$.

Let $T_{mult}[\mathbb{Q}]$ be the theory of $(\mathbb{R}, 0, 1, \times, \div, \sqrt[n]{\cdot}, <, \ldots, f_a, \ldots)$.

Let $T_{common}[\mathbb{Q}] = T_{add}[\mathbb{Q}] \cap T_{mult}[\mathbb{Q}]$.

Let $T[\mathbb{Q}] = T_{add}[\mathbb{Q}] \cup T_{mult}[\mathbb{Q}]$. This theory seems to be very useful.

$T_{add}[\mathbb{Q}]$, $T_{mult}[\mathbb{Q}]$, $T_{common}[\mathbb{Q}]$ all have quantifier elimination.

But the presence of the new symbols in the common language makes the situation much more complex.

Real inequalities

Harvey Friedman and I showed:

- $T[\mathbb{Q}]$ has good normal forms.
- Valid equations are independent of the ordering.
- $T[\mathbb{Q}]$ is undecidable.
- In fact, the $\forall\forall\forall\exists \ldots \exists$ fragment is complete r.e.
- Assuming that the solvability of Diophantine equations in the rationals is undecidable, then so is the existential fragment of $T[\mathbb{Q}]$.

Most important:

- The universal fragment of $T[\mathbb{Q}]$ *is* decidable.

More generally, we consider theories $T[F]$, for arbitrary computable subfields F of $\mathbb{R}$.

Real inequalities

One can simultaneously define normal forms and an ordering on terms in normal form.

$$4(1 + 3x_1 + 4x_1x_7)^2(x_1^2x_2^3 + 4x_3^2x_9^2)^3$$

Two terms are provably equal if and only if they have the same normal form.

In that case, they are provably equal in the theory without the ordering.

Real inequalities

Our decidability results are not practical. But the proofs provide ideas and guidelines.

We propose the following strategy: given a sequent

$$r_1 < s_1, r_2 \leq s_2, \ldots, r_k < s_k \Rightarrow t < u,$$

put all terms in normal form, and try to refute

$$r_1 < s_1, r_2 \leq s_2, \ldots, r_k < s_k, u \leq t.$$

To do this, you need to find an interpolant.

Iteratively use the additive and multiplicative parts to derive new inequalities, $p < aq$ or $p \leq aq$, between "subterms."

Real inequalities

Disadvantages:

- The procedure is not complete (need disjunctions).
- The procedure may not terminate.
- Need to consider arbitrary pairs of subterms.

Advantages:

- The method has the right flavor: forward reasoning, but focusing on "potentially useful" comparisons.
- It includes arithmetic and multiplicative decision procedures.
- It works on the kinds of examples I described above.

We expect that the method will work well in practice, but experimentation is needed.

Real inequalities

The method is, furthermore, open-ended and extensible:

- One can judiciously incorporate distributivity.
- One can judiciously incorporate disjunctions (case splits).
- One can add rules for e^x, $\ln x$, sin, cos, ...
- One can add general rules for monotone functions.

There are:

- interesting implementation issues
- interesting theoretical issues

Euclidean diagrammatic reasoning

For more than two thousand years, Euclid's *Elements* was held to be the paradigm for rigorous argumentation.

But the nineteenth century raised concerns:

- Conclusions are drawn from diagrams, using "intuition" rather than precise rules.
- Particular diagrams are used to infer general results (without suitable justification).

Axiomatizations due to Pasch and Hilbert, and Tarksi's formal axiomatization later on, were thought to make Euclid rigorous.

Euclidean diagrammatic reasoning

But in some ways, they are unsatisfactory.

- Proofs in the new systems look very different from Euclid's.
- The initial criticisms belie the fact that Euclidean practice was remarkably stable for more than two thousand years.

Our project (John Mumma, Ed Dean, and me):

- Describe a formal system that is much more faithful to Euclid.
- Argue that the system is sound and complete (for the theorems it can express) relative to Euclidean fields.
- Suggest that the system can easily be implemented using contemporary automated reasoning technology.

Euclidean diagrammatic reasoning

Observations:

- Proof generally have two parts: the construction, and the demonstration.
- Diagrams are used only to record "coexact" information (a term due to Ken Manders). "Exact" information is licensed explicitly in the text.
- Theorems and proofs have a very restricted logical form.

Dean, Mumma, and I designed a formal system with these features, with explicit:

- construction rules, and
- deductive inferences.

Diagram inferences are implicit in both.

Construction rules

"Let p be a point on L"
No prerequisites.

"Let p be a point distinct from q and r"
No prerequisites.

"Let L be the line through p and q"
Requires $p \neq q$.

"Let p be the intersection of L and M."
Requires that L and M intersect.

And so on...

Deductive inferences

Four types:

1. Diagram inferences: any fact that can be "read off" from the diagram.
2. Metric inferences: essentially linear arithmetic on lengths, angles, and areas.
3. Diagram to metric: for example, if q is between p and r, then $\overline{pq} + \overline{qr} = \overline{pr}$, and similarly for areas and angles.
4. Metric to diagram: for example, if p is the center of γ, q is on γ, and $\overline{pr} < \overline{pq}$, then r is inside γ.

Diagram inferences

Both construction inferences and diagram inferences require an account of what can be "read off" from the diagram.

We get this by closing the diagrammatic facts introduced in the proof under various rules, including:

- properties of "between"
- properties of "same side"
- "Pasch rules," relating "between" and "same side"
- triple incidence rules
- circle rules
- intersection rules

These yield conclusions that are generally valid, that is, common to all possible realizations.

Diagram inferences

The set of diagrammatic consequences of a given context amount to the set of consequences of some universal axioms, in a language with no function symbols.

Hence, this set is trivially decidable.

To model the *Elements*, we actually characterized a subset of these (the "direct consequences").

These treat the axioms as rules (up to contrapositive equivalents) and chain forward, without case splits.

But even the full set of first-order consequences were easily obtained using resolution provers (we tried E and Spass) and some SMT solves (like Z3 and CVC3).

Discussion

Some thoughts about automated support in formal verification:

There is a tension between domain general methods and domain specific methods. What we need are general approaches to domain specific reasoning (e.g. with domain specific features encoded by specific rules and parameters).

One wants transparency: one should have a sense of when the methods should succeed, and when they fail, it should be possible to determine why (using traces, or "binary" checks).

One wants flexibility to get things working again (adding local information, setting parameters, adjusting behavior based on context).

One wants efforts to scale.

Discussion

Further speculation:

- "Guided" forward reasoning seems promising, especially if one can limit the data gathered (type information, set inclusions, inequalities, relationships in a diagram, big O equations, etc.).
- Cooperation between specialized modules seems important.
- A lot more experimentation is needed, with real mathematical contexts.
- It would be helpful to have a better theory, to characterize the situations in which one can expect good behavior.

This provides good opportunities for collaborations between logicians, mathematicians, and computer scientists (and philosophers).