Slaughtering Competition Problems with Quantifier Elimination

22 DEC 2021 - TAGS: SAGE , FEATURED

Anytime I see questions on mse that ask something "simple", I feel a powerful urge to chime in with "a computer can do this for you!". Obviously if you're a researching mathematician you shouldn't waste your time with something a computer can do for you, but when you're still learning techniques (or, as is frequently the case on mse, solving homework problems), it's not a particularly useful comment (so I usually abstain). The urge is *particularly* powerful when it comes to the contrived inequalities that show up in a lot of competition math, and today I saw a question that *really* made me want to say something about this! I still feel like it would be a bit inappropriate for mse, but thankfully I have a blog where I can talk about whatever I please :P So today, let's see how to hit these problems with the proverbial nuke that is quantifier elimination!

I want this to be a fairly quick post, so I won't go into too much detail. The gist is the following powerful theorem from model theory:

Tarski-Seidenberg Theorem[1]

If φ is any formula of the form

- $p(\overline{x}) = 0$, for $p \in \mathbb{R}[\overline{x}]$
- $p(\overline{x}) < 0$, for $p \in \mathbb{R}[\overline{x}]$
- combinations of the above using $\lor$, $\land$, $\neg$, $\to$
- combinations of the above using $\exists$ and $\forall$

Then φ is equivalent to a formula *without* quantifiers.

I'm legally required to give the following example:

The formula $\exists x.\, ax^2 + bx + c = 0$ (which has a quantifier) is equivalent to the formula $b^2 - 4ac \geq 0$

As a more complicated example, we have

$$\forall x.\, \exists y.\, (x > 0 \to ax + by + xy > c)$$

is equivalent to

$$b \geq 0 \ \lor \ c + ab < 0$$

Of course, this means if we want to know whether the above formula really is true for some choice of $a, b, c \in \mathbb{R}$, we can just plug into this quantifier free formula and check!

Now, you might be wondering: "How did you find this quantifier free expression?", and the answer is, of course, sage! Sage has interfaces with a lot of pre-existing software, and for us the relevant interface is to QEPCAD, which will actually *do* quantifier elimination for us!

To get started, you have to make sure you have qepcad installed in a way that sage can access. You'll want to run `sage -i qepcad` just in case (it wasn't installed for me).

Next, let's see how we eliminated quantifiers from the "more complicted example" above!

```
qepcad('(A x)(E y)[x > 0 ==> a x
```

EVALUATE

SHARE

```
b >= 0 \/ c + a b < 0
```

Help | Powered by SageMath

Yup. It's *that* easy!

If you're interested in reading more about this, you should check out the documentation, but I'm also going to give a handful of examples in the next section[2]!

So then, let's slaughter some competition problems[3]!

First, the problem that made me write this post in the first place (here is the mse link again)

Let $a, b \geq 0$ with $a^4 + b^4 = 17$.

Prove $15(a+b) \geq 17 + 14\sqrt{2ab}$

This isn't given to us as polynomials, but of course it's easy for us to fix that by rewriting it as

$$(15(a+b) - 17)^2 \geq 14^2 \cdot 2ab$$

then we simply ask sage[4]:

```
qf = qepcad_formula
a,b = var('a,b')
P = qf.and_(a >= 0, b >= 0, a^4 +
Q = qf.atomic((15 * (a+b) - 17)^2
qepcad(qf.implies(P,Q))
```

EVALUATE

SHARE

```
TRUE
```

Help | Powered by SageMath

We can extend this too. The asker conjectures that $(1, 2)$ and $(2, 1)$ are the only choices of (a, b) for which we get equality. As a bonus, we can check this:

```
 = qepcad_formula
b = var('a,b')
= qf.atomic((15 * (a+b) - 17)^2 =
pcad(Q, assume=[a >= 0, b >= 0, a
```

EVALUATE

SHARE

```
[{'a': 2, 'b': 1}, {'a': 1, 'b': 2}]
```

Help | Powered by SageMath

So we see that these really *are* the only points where equality holds!

Let's take another example I remember seeing recently (the original mse link is here):

> Let x, y, z be positive real numbers. Show
>
> $$\left(x+\frac{1}{x}\right)\left(y+\frac{1}{y}\right)\left(z+\frac{1}{z}\right) \geq \left(x+\frac{1}{y}\right)\left(y+\frac{1}{z}\right)\left(z+\frac{1}{x}\right)$$

Again, we cannot plug this into sage directly, because it's not a *polynomial* inequality. But multiplying through by xyz on both sides solves that issue.

```
qf = qepcad_formula
x,y,z = var('x,y,z')
Q = x*y*z * (x + 1/x)*(y + 1/y)*(
Q = qf.atomic(Q.expand())
qepcad(Q, assume=[x > 0, y > 0, z
```

EVALUATE

SHARE

```
TRUE
```

Help | Powered by SageMath

and even though the asker doesn't mention it, one thing that Steele makes very clear in the (excellent) book *The Cauchy-Schwarz Masterclass* is that whenever working with a new inequality, we should ask where it's sharp.

So we ask sage!

It turns out this inequality is sharp at infinitely many points, so instead of asking for the list of all points, we ask for a geometric description of the solution set.

```
qf = qepcad_formula
x,y,z = var('x,y,z')
Q = x*y*z * (x + 1/x)*(y + 1/y)*(
Q = qf.atomic(Q.expand())
qepcad(Q, assume=[x > 0, y > 0, z
```

EVALUATE

SHARE

```
x > 0
/\
y - x = 0
/\
y^2 z^2 - x y z^2 + x^2 z^2 + z^2 - x
```

Help | Powered by SageMath

Now, this admits some simplification, since we know that $x = y$ by the second line. I'll leave it to you to figure out exactly what set this is if you're interested.

As an exercise, you should be on the lookout for places to use this tool!

Next time somebody is asking about some wacky inequality, or really *any* question about sets definable by polynomial (in)equations in $\mathbb{R}^n$, you should think about whether you can slaughter the problem without much thought by asking a computer!

As a more concrete exercise to show the flexibility of this method, pick your favorite theorem in euclidean geometry. Rephrase it using coordinates, then ask sage if it's true!

As a *very* concrete exercise, can you do this with Ptolemy's Theorem?

Also, if you find yourself using this, definitely come back and let me know! I would love to hear about places where this comes up in the wild!

Also also, if you have other (possibly surprising) uses for sage or other programs that automatically answer ceretain problems, definitely let me know! This is one of the parts of mathematical logic that I get most geeky about!

See you next time ^_^

---

1. As a cute thought exercise, you should try to provide geometric meaning to this claim. It's telling us that if you take the solution set of polynomial inequations, then project from $\mathbb{R}^{n+m}$ down to $\mathbb{R}^n$, the resulting set is *still* definable by polynomial inequations!

   This should sound somewhat miraculous, and it's worth trying out some

Thankfully, by the end of the post, you'll have all the tools you need in order to work out some examples on your own ^_^. ↩

2. In fact, there are $\sim\star\sim$ bonus quantifiers $\sim\star\sim$ built into QEPCAD! For instance, we can ask for

   - "there exist exactly 5 x so that $\varphi(x)$" (and obviously there's nothing special about 5)
   - "there exist infinitely many x so that $\varphi(x)$"
   - "the set of x so that $\varphi(x)$ is a connected set"

   and while these aren't going to be useful for the purposes of this post, I still wanted to mention them! ↩

3. I know that I'm currently treating this like a kind of party trick, but being able to ask a computer whether an implication between polynomial inequalities is true (and being able to find a counterexample if it isn't) is *super* useful in practice! In fact, André Platzer at CMU crucially uses this machinery in order to automatically prove that robots will not bump into each other (etc.). See, for instance, his book *Logical Foundations of Cyber-Physical Systems*, as well as the accompanying lectures on youtube. ↩

4. For some reason typing this out wasn't working for me, but using the constructors directly got things going… There's probably something to do with the parsing that I don't understand, and this isn't too much of a hassle! ↩

0 reactions

1 comment – *powered by giscus*

Oldest Newest