# Quantifier Elimination

BY TIGRAN MKRTCHYAN
SAARLAND UNIVERSITY

14.12.2012

# Abstract



- Quantifier elimination (QE) is the main technique to eliminate quantifiers of a formula F until only a quantifier-free formula G that is equivalent to F remains.
- Task of proving the verification conditions.
- Decide validity in $T_Z$ and $T_Q$

# Outline



1) Motivating example
2) Formal Description
3) Cooper's method
4) Ferrante & Rackoff's method
5) Summary

# Motivating Example



Consider the formula

$F : \exists x.\ 2x = y$ ,

which expresses the set of rationals y that can be halved. Intuitively, all rationals can be halved, so a quantifier-free equivalent formula is :

$G : \top$ ,

which expresses the set of all rationals. Also, G states that F is valid.

# Motivating Example



Consider the same formula

$F : \exists x.\ 2x = y$ ,

which expresses the set of integers y that can be halved (to produce another integer). Intuitively, only even integers can be halved.

For example, an equivalent formula to F is

$G : 2 \mid y$ ,

which expresses the set of even integers: integers that are divisible by 2.

# Outline



1) Motivating example
2) Formal Description
3) Cooper's method
4) Ferrante & Rackoff's method
5) Summary

# Formal Description



- Formally, a theory T admits quantifier elimination if there is an algorithm that, given Σ-formula F, returns a quantifier-free Σ-formula G that is T -equivalent to F. Then T is decidable if satisfiability in the quantifier-free fragment of T is decidable.

# Formal Description: Remark



- In developing a QE algorithm for theory $T$ , we need only consider formulae of the form $\exists x.\ F$ for quantifier-free formula $F$.

- For given arbitrary formula $G$, choose the innermost quantified formula $\exists x.\ H$ or $\forall x.\ H$. In the latter case, rewrite $\forall x.\ H$ as $\neg(\exists x.\ \neg H)$ and focus on the subformula $\exists x.\ \neg H$ inside the negation.

  In the existential case, replace $\exists x.\ H$ in $G$ with $H'$.
  In the universal case, replace $\forall x.\ H$ in $G$ with $\neg H'$.

# Formal Description: Remark (example)



$G1 : \exists x.\ \forall y.\ \exists z.\ F1[x, y, z]$ ,

The innermost quantified formula is $\exists z.\ F1[x, y, z]$. Applying the QE algorithm for T to this subformula returns $F2[x, y]$:

$G2 : \exists x.\ \forall y.\ F2[x, y]$ .

The innermost quantified formula is now $\forall y.\ F2[x, y]$; rewriting, we have

$G3 : \exists x.\ \neg(\exists y.\ \neg F2[x, y])$ .

Applying the QE algorithm to existential subformula $\exists y.\ \neg F2[x, y]$ produces $F3[x]$.

$G4 : \exists x.\ \neg F3[x]$ .

Finally, applying the QE algorithm one more time to G4 produces a quantifier free formula G5, where G5 is T -equivalent to G1.

# Formal Description : Theory of Integers



- $\Sigma_Z$ : {. . . ,−2,−1, 0, 1, 2, . . . ,−3·,−2·, 2·, 3·, . . . , +, −, =, <}
- $\exists x.\ 2x = y$,
- Augment the theory $T_Z$ with an infinite but countable number of unary divisibility predicates $k \mid \cdot$ for $k \in Z+$ ;

$x > 1 \wedge y > 1 \wedge 2 \mid x + y$ is satisfiable, but

$\neg(2 \mid x) \wedge 4 \mid x$ is not satisfiable.

- $\forall x.\ k \mid x \leftrightarrow \exists y.\ x = ky$ (divides) for $k \in Z+$.
- Modified $T_Z$ admits QE.

# Outline



1) Motivating example
2) Formal Description
3) Cooper's method
4) Ferrante & Rackoff's method
5) Summary

# Cooper's Algorithm : Abstract



- It is a quantifier elimination procedure, which also works from the inside out, eliminating existentials.
- Its *big* advantage is that it doesn't need to normalize input formulas to DNF.
- Description is of simplest possible implementation; many tweaks are possible.

# Cooper's algorithm : Preprocessing



- To eliminate the quantifier in $\exists x.\ P(x)$:

1. Normalize so that only operators are <, and divisibility ($c|e$), and negations only occur around divisibility leaves.
2. Compute least common multiple $c$ of all coefficients of $x$, and multiply all terms by appropriate numbers so that in every term the coefficient of $x$ is $c$.
3. Now apply

$$(\exists\, x.\ P(cx)) \equiv (\exists\, x.\ P(x) \wedge c|x).$$

# Preprocessing Example



$\forall x,y \in Z.\ 0 < y \wedge x < y \Rightarrow x + 1 < 2y$

*(normalize)*

$\equiv \neg\exists x,y.\ 0 < y \wedge x < y \wedge 2y < x + 2$

*(transform y to 2y everywhere)*

$\equiv \neg\exists x,y.\ 0 < 2y \wedge 2x < 2y \wedge 2y < x + 2$

*(give y unit coefficient)*

$\equiv \neg\exists x,y.\ 0 < y \wedge 2x < y \wedge y < x + 2 \wedge 2 | y$

# Two cases



- How might $\exists x.\ P(x)$ be true?
- Either:
  - there is a least $x$ making $P$ true; or
  - there is no least $x$: however small you go, there will be a smaller $x$ that still makes $P$ true
- Construct two formulas corresponding to both cases.

# Case 1: Infinitely many small solutions



- Look at the atomic formulas in *P*, and think about their values when *x* has been made arbitrarily small:
  - $x < e$: if *x* goes as small as we like, this will be T
  - $e < x$: if *x* goes small, this will be $\perp$
  - $c \mid x+e$: *unchanged*
- This constructs $P_{-\infty}$, a formula where *x* only occurs in divisibility terms.
- Say δ is the l.c.m. of the constants involved in divisibility terms. Need just test $P_{-\infty}$ on 1,…, δ.

# $P_{-\infty}$ example



- For $\exists y.\ 0 < y \wedge 2x < y \wedge y < x + 2 \wedge 2|y$
  - $0 < y$ will become $\perp$ as $y$ gets small
  - $2x < y$ also becomes $\perp$ as $y$ gets small
  - $y < x + 2$ will be T as $y$ gets small
  - $2|y$ doesn't change (it tests if $y$ is even or not)
- So in this case,

$$P_{-\infty}(y) \equiv (\perp \wedge \perp \wedge \mathrm{T} \wedge 2|y) \equiv \perp$$

# Case 2: Least solution



- The case when there is a least $x$ satisfying $P$.
- For there to be a least $x$ satisfying $P$, it must be the case that one of the terms $e < x$ is T, and that if $x$ was any smaller the formula would become $\perp$.
- Let $B = \{b \mid b < x$ is a term of $P\}$
- Need just consider $P(b+j)$, where $b \in B$ and $1 \leq j \leq \delta$.
- Final elimination formula is:

$$(\exists x.\ P(x)) \equiv \bigvee_{j=1..\delta} P_{-\infty}(j) \vee \bigvee_{j=1..\delta} \bigvee_{b \in B} P(b+j)$$

# Example continued



- For

$$\exists y.\ 0 < y \wedge 2x < y \wedge y < x + 2 \wedge 2 | y$$

- least solutions, if they exist, will be at

$$y = 1,\ y = 2,\ y = 2x + 1, \text{ or } y = 2x + 2$$

- The divisibility constraint eliminates two of these.
- Original formula is equivalent to:

$$(2x < 2 \wedge 0 < x) \vee (0 < 2x + 2 \wedge x < 0)$$

Which is unsatisfiable.

# Cooper's method



- The algorithm is given a $\Sigma_Z$ -formula $\exists x.\ F[x]$ as input, where F is quantifier-free but may contain free variables in addition to x.

- It then proceeds to construct a quantifier-free $\Sigma_Z$ -formula that is $T_Z$-equivalent to $\exists x.\ F[x]$ according to the following (5) steps.

# Cooper's method



- Step 1
- Put F[x] in NNF.
- The output $\exists x.\ F1[x]$ is $T_Z$-equivalent to $\exists x.\ F[x]$ and is such that F1 is a positive Boolean combination (only $\wedge$ and $\vee$) of literals.

# Cooper's method



- Step 2
- Replace literals according to the following $T_Z$-equivalences, applied from left to right:

$$s = t \quad \Leftrightarrow s < t + 1 \wedge t < s + 1$$

$$\neg(s = t) \Leftrightarrow s < t \vee t < s$$

$$\neg(s < t) \Leftrightarrow t < s + 1$$

- The output $\exists x.\ F_2[x]$ is $T_Z$-equivalent to $\exists x.\ F[x]$ and contains only literals of the form
- $s < t$ , $k \mid t$ , or $\neg(k \mid t)$ ,
- where s, t are $\Sigma_Z$-terms and $k \in Z_+$.

# Cooper's method



- Example :

  Applying the $T_Z$-equivalences to

$\neg(x < y) \wedge \neg(x = y + 3)$

  produces the $T_Z$-equivalent formula

$y < x + 1 \wedge (x < y + 3 \vee y + 3 < x)$ .

# Cooper's method



- Step 3
- Collect terms containing x s.t. literals have the form

$hx < t$ , $t < hx$ , $k \mid hx + t$ , or $\neg(k \mid hx + t)$ ,

- where t is a term that does not contain x and h, k $\in \mathbb{Z}_+$. The output is the formula $\exists x.\ F3[x]$, which is $T_{\mathbb{Z}}$-equivalent to $\exists x.\ F[x]$.

# Cooper's method



- Collecting terms in

$x + x + y < z + 3z + 2y - 4x$

- produces the $T_Z$-equivalent formula

$6x < 4z + y$ .

# Cooper's method



- Step 4 : Let $\delta' = \text{lcm}\{h : h \text{ is a coefficient of } x \text{ in } F3[x]\}$ ,where lcm returns the least common multiple of the set. Multiply atoms in F3[x] by constants so that $\delta'$ is the coefficient of x everywhere:

$$hx < t \Leftrightarrow \delta' x < h' t \qquad \text{where } h'h = \delta'$$

$$t < hx \Leftrightarrow h' t < \delta' x \qquad \text{where } h'h = \delta'$$

$$k \mid hx + t \Leftrightarrow h'k \mid \delta' x + h' t \qquad \text{where } h'h = \delta'$$

$$\neg(k \mid hx + t) \Leftrightarrow \neg(h'k \mid \delta' x + h' t) \qquad \text{where } h'h = \delta'$$

- This results in formula $F'3$ in which all occurrences of x occur in terms $\delta' x$. Replace $\delta' x$ terms with a fresh variable $x'$ to form

$F''3 : F'3\ \{\delta' x \rightarrow x'\}$ .

# Cooper's method



- Finally, construct

$\exists x'.\ F''_3[x'] \wedge \delta' \mid x'$ : $F_4[x']$

- The divisibility literal constrains the fresh variable $x'$ to be divisible by $\delta'$, which exactly captures the values of $\delta' x$. $\exists x'.\ F_4[x']$ is Tz-equivalent to $\exists x.\ F[x]$.

  Moreover, each literal of $F_4[x']$ that contains $x'$ has one of the following forms:

(A) $x' < a$

(B) $b < x'$

(C) $h \mid x' + c$

(D) $\neg(k \mid x' + d)$

- where a, b, c, d are terms that do not contain x, and $h, k \in Z+$.

# Cooper's method



- Step 5
- Construct the left infinite projection $F_{-\infty}[x']$ from $F_4[x']$ by replacing
- (A) literals $x' < a$ by $\top$ and
- (B) literals $b < x'$ by $\perp$ .

- The idea is that very small numbers (the left side of the "number line") satisfy (A) literals but not (B) literals.

# Cooper's method



- Let

$$\delta = \text{lcm}\left\{\begin{array}{l} h \text{ of (C) literals } h \mid x' + c \\ \\ k \text{ of (D) literals } \neg(k \mid x' + d) \end{array}\right\}$$

- and B be the set of b terms appearing in (B) literals. Construct
- F5 : $\bigvee_{j=1;\,\delta} F_{-\infty}[j] \vee \bigvee_{j=1;\,\delta} \bigvee_{b\in B} F_4[b + j]$ .
- F5 is quantifier-free and Tz-equivalent to $\exists x.\ F[x]$.

# Cooper's method



$$(\exists x.\, P(x)) \equiv \bigvee_{j=1..\delta} P_{-\infty}(j) \vee \bigvee_{j=1..\delta} \bigvee_{b \in B} P(b+j)$$

- The first major disjunct of F5 contains only divisibility literals. It asserts that an infinite number of small numbers n satisfy F4[n].
- For if there exists one number n that satisfies the Boolean combination of divisibility literals in P−∞, then every n – λδ, for λ ∈ Z+, also satisfies P−∞.
- The second major disjunct asserts that there is a least n ∈ Z that satisfies F4[n]. This least n is determined by the b terms of the (B) literals.

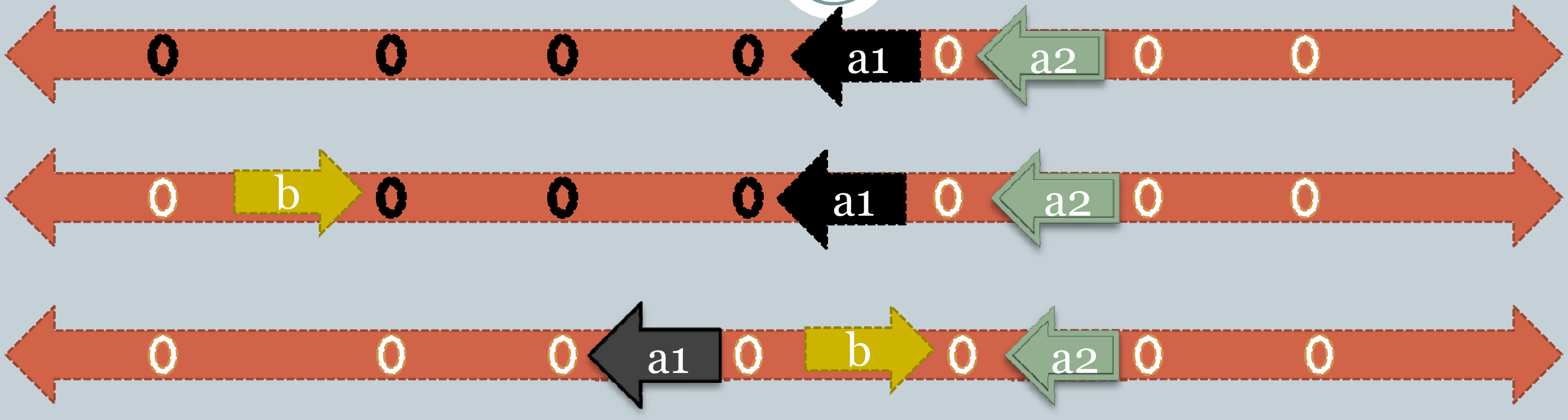
# Cooper's method



- If $m \mid \delta$, then $m \mid n$ iff $m \mid n + \lambda\delta$ for all $\lambda \in Z$.
- Since $\delta$ is chosen in Step 5 to be the l.c.m. no divides literal can distinguish between two integers $n$ and $n + \lambda\delta$,
- If $n \in Z$ satisfies $F[n]$, then so does $n - \lambda\delta$ for $\lambda \in Z_+$. Then surely a small enough number exists that satisfies all (A) literals and falsifies all (B) literals of F4, mirroring the construction of $F-\infty$.
- suppose that some number n satisfies F4[n]. Decreasing this number continues to satisfy the same (A) literals. It cannot decrease past some value $b*$ without changing the truth of some (B) literal.
- (A) literals $x' < a$ by $\top$ and (B) literals $b < x'$ by $\perp$ .

# Cooper's method






(a) Left infinite projection (b) δ-interval (c) false

- (a) illustrates a formula $x < a1 \wedge x < a2 \wedge \delta \mid x$: each left-pointing arrow represents a $x < a_i$ literal. The left infinite projection is satisfied.
- (b) illustrates an additional $x > b$ literal; now, the δ-interval following the right-pointing arrow at b is searched. It contains satisfying points.
- $b > a1$ in (c), so the δ-interval does not contain a satisfying point.

# Cooper's method (1) example



- Consider $\Sigma_Z$-formula

$\exists x.\ 3x - 2y + 1 > -y \wedge 2x - 6 < z \wedge 4 \mid 5x + 1 \qquad : F[x]$

- After Step 3, we have

$\exists x.\ 2x < z + 6 \wedge y - 1 < 3x \wedge 4 \mid 5x + 1 \qquad :F3[x]$

- Collecting coefficients of x in Step 4, we find

$\delta' = \mathrm{lcm}\{2, 3, 5\} = 30$ .

# Cooper's method



- Multiplying when necessary, we rewrite the formula so that 30 is the coefficient of every occurrence of x:
- $\exists x.\ 30x < 15z + 90 \wedge 10y - 10 < 30x \wedge 24 \mid 30x + 6$ .
- Replacing 30x with fresh x′ and conjoining a divides atom completes Step 4:
- $\exists x'.\ x' < 15z + 90 \wedge 10y - 10 < x' \wedge 24 \mid x' + 6 \wedge 30 \mid x' \quad : F_4[x']$

# Cooper's method



For Step 5, construct the left infinite projection
$F_{-\infty}[x] : \top \wedge \bot \wedge 24 \mid x' + 6 \wedge 30 \mid x'$ , which simplifies to $\bot$. Compute
$\delta = \text{lcm}\{24, 30\} = 120$ and $B = \{10y - 10\}$ . Replacing $x'$ by $10y - 10 + j$ in

$$F5 : \bigvee_{j=1;\,120} \left\{ \begin{array}{c} 10y - 10 + j < 15z + 90 \wedge 10y - 10 < 10y - 10 + j \\ \wedge\ 24 \mid 10y - 10 + j + 6 \wedge 30 \mid 10y - 10 + j \end{array} \right\}$$

$$F5 : \bigvee_{\mathbf{j=1;\,120}} \left\{ \begin{array}{c} 10y + j < 15z + 100 \wedge 0 < j \\ \wedge\ 24 \mid 10y + j - 4 \wedge 30 \mid 10y + j - 10 \end{array} \right\}$$

- 

- F5 is quantifier-free and Tz-equivalent to $\exists x.\ F[x]$.

# Cooper's method 2 example



- Consider again the formula defining the set of even integers:

$\exists x.\ 2x = y$ : $F[x]$

- Rewriting according to Steps 2 and 3 produces

$\exists x.\ y - 1 < 2x \wedge 2x < y + 1$ . Then

$\delta' = \mathrm{lcm}\{2, 2\} = 2$ ,

- so Step 4 completes with

$\exists x'.\ y - 1 < x' \wedge x' < y + 1 \wedge 2 \mid x'$ : $F4[x']$

# Cooper's method 2 example



- Computing the left infinite projection $F_{-\infty}$ produces $\perp$, as $F_4[x']$ contains a (B) literal as a conjunct.
  However, $\delta = \text{lcm}\{2\} = 2$ and $B = \{y - 1\}$ , so

$$F_5 : \bigvee_{j=1} \{y - 1 < y - 1 + j \wedge y - 1 + j < y + 1 \wedge 2 \mid y - 1 + j\}$$ ,or

$$F_5 : \bigvee_{j=1} \{0 < j \wedge j < 2 \wedge 2 \mid y + j - 1\}$$ , and then

$F_5 : 2 \mid y$ ,

which is quantifier-free and $T_Z$-equivalent to $\exists x.\ F[x]$.

# Cooper's method (3) example



- Consider the formula

$\exists x.\ (3x + 1 < 10 \lor 7x - 6 > 7) \land 2 \mid x \qquad :F[x]$

- Rewriting to isolate x terms produces

$\exists x.\ (3x < 9 \lor 13 < 7x) \land 2 \mid x$, so $\delta' = \text{lcm}\{3, 7\} = 21$.

- After multiplying coefficients by proper constants,

$\exists x.\ (21x < 63 \lor 39 < 21x) \land 42 \mid 21x$, replace $21x$ by $x'$:

$\exists x'.\ (x' < 63 \lor 39 < x') \land 42 \mid x' \land 21 \mid x' \qquad :F4[x']$

# Cooper's method (3) example



$F_{-\infty}[x'] : (\top \vee \bot) \wedge 42 \mid x' \wedge 21 \mid x'$ , or, simplifying,
$F_{-\infty}[x'] : 42 \mid x' \wedge 21 \mid x'$ . Finally,
$\delta = \mathrm{lcm}\{21, 42\} = 42$ and $B = \{39\}$ , so

- $F_5 : \bigvee_{j=1;\,42} (42 \mid j \wedge 21 \mid j) \vee$

$$\bigvee_{j=1;\,42} ((39 + j < 63 \vee 39 < 39 + j) \wedge 42 \mid 39 + j \wedge 21 \mid 39 + j) .$$

- Since $42 \mid 42$ and $21 \mid 42$, the left main disjunct simplifies to $\top$, so that $\exists x.\ F[x]$ is $T_{\mathbb{Z}}$-equivalent to $\top$. Thus, $F$ is $T_{\mathbb{Z}}$-valid.

# Cooper's method: Theorem



- **Theorem :** Given $\Sigma_Z$-formula $\exists x.\ F[x]$ in which F is quantifier- free, Cooper's method returns a $T_Z$-equivalent quantifier-free formula.

- **Proof**. The transformations of the first four steps produce formula F4. By inspection, we assert that in $T_Z$

$\exists x.\ F[x] \Leftrightarrow \exists x.\ F4[x]$ .

The focus of the proof is to prove that $\exists x.\ F4[x] \Leftrightarrow F5$ in $T_Z$:

# Cooper's method: Theorem



- $\exists x.\ F4[x] \Leftrightarrow \bigvee_{j=1;\ \delta} F{-\infty}[j] \vee \bigvee_{j=1;\delta} \bigvee_{b\in B} F4[b + j]$ .
- We accomplish the proof in two steps.
- 1. $F5 \Rightarrow \exists x.\ F4[x]$:
  We assume the existence of an interpretation I such that

I $|= F5$ and prove that I $|= \exists x.\ F4[x]$.

- 2. $\exists x.\ F4[x] \Rightarrow F5$:
  We assume the existence of an interpretation I such that
- I $|= \exists x.\ F4[x]$ and prove that I $|= F5$.

# Cooper's method



- (1) Assume then that $I \models F5$, so that one of the disjuncts of F5 is true under I. If one of the second set of disjuncts is true, say $F4[b* + j*]$, then $I \rhd \{x \to b* + j*\} \models F4[x]$ . $I \models \exists x.\ F4[x]$ .
- Otherwise, one of the first set of disjuncts is true, so for some $j* \in [1, \delta]$, $I \rhd \{x \to j*\} \models F{-\infty}[x]$.
  By construction of $F{-\infty}$, there is some $\lambda > 0$ such that $I \rhd \{x \to j* - \lambda\delta\} \models F4[x]$.
- That is, there is some $j* - \lambda\delta$ that is so small that the inequality literals of F4 evaluate under $I \rhd \{x \to j* - \lambda\delta\}$ exactly as in the construction of $F{-\infty}$.
  Thus, $I \models \exists x.\ F4[x]$ in this case as well.

# Cooper's method



- (2) Assume $I \models \exists x.\ F_4[x]$. Thus, some $n \in \mathbb{Z}$ exists such that $I \triangleright \{x \mapsto n\} \models F_4[x]$. If for some $b* \in B$ and $j* \in [1, \delta]$, $I \models n = b* + j*$, then $I \models F_4[b* + j*]$.
- As $F_4[b* + j*]$ is a disjunct of $F_5$, $I \models F_5$.
- Otherwise, consider whether $I \triangleright \{x \mapsto n - \delta\} \models F_4[x]$. If not, then one of the (B) literals, say $b* < x$ for some $b* \in B$, of $F_4$ becomes false under $I$ in the transition from $n$ to $n - \delta$. But then $I \models n = b* + j*$ for some $j* \in [1, \delta]$, contradicting our assumption that $n$ is not equal to some $b* + j*$.
- Hence, it must be the case that $I \triangleright \{x \mapsto n - \delta\} \models F_4[x]$.

# Cooper's method



- By induction using this argument

$I \rhd \{x \rightarrow n - \lambda\delta\} \models F4[x]$ for all $\lambda > 0$.

For some $\lambda$, $n - \lambda\delta$ becomes so small that

$I \rhd \{x \rightarrow n - \lambda\delta\} \models F4[x] \leftrightarrow F{-\infty}[x]$ , so

$I \rhd \{x \rightarrow n - \lambda\delta\} \models F{-\infty}[x]$ .

- That is, $n - \lambda\delta$ is so small that the inequality literals of F4 evaluate under $I \rhd \{x \rightarrow n - \lambda\delta\}$ exactly as in the construction of $F{-\infty}$.
  Now, since $F{-\infty}$ contains only divides literals, we can choose a $\mu$ such that $n-\lambda\delta+\mu\delta \in [1, \delta]$.

Let $j* = n - \lambda\delta + \mu\delta$. Then $I \models F{-\infty}[j*]$, so that $I \models F5$.

# Outline



1) Motivating example
2) Formal Description
3) Cooper's method
4) Ferrante and Rackoff's Method
5) Summary

# Ferrante & Rackoff's method



QE for the theory of rationals $T_Q$ is simpler than for $T_Z$. Recall that TQ has the following signature:

$\Sigma_Q : \{0, 1, +, -, =, \geq\}$ , where

- 0 and 1 are constants;
- + is a binary function;
- – is a unary function;
- and = and ≥ are binary predicates.

To be consistent with our presentation of Cooper's method, we switch from weak inequality ≥ to strict inequality >.

$x \geq y \Leftrightarrow x > y \vee x = y$ and $x > y \Leftrightarrow x \geq y \wedge \neg(x = y)$ .

# Ferrante & Rackoff's method



- Given a $\Sigma Q$-formula $\exists x.\ F[x]$ as input, where F is quantifier-free, the algorithm proceeds according to the following (4) steps.
- Step 1
- Put $F[x]$ in NNF. The output $\exists x.\ F_1[x]$ is TQ-equivalent to $\exists x.\ F[x]$ and is such that $F_1$ is a positive Boolean combination (only $\wedge$ and $\vee$) of literals.

# Ferrante & Rackoff's method



- Step 2

- Replace literals according to the following TQ-equivalences, applied from left to right:

$\neg(s < t) \Leftrightarrow t < s \vee t = s$

$\neg(s = t) \Leftrightarrow t < s \vee t > s$

- The output $\exists x.\ F2[x]$ is TQ-equivalent to $\exists x.\ F[x]$ and does not contain any negations.

# Ferrante & Rackoff's method



- Step 3
- Solve for x in each atom of $F2[x]$: for example, replace the atom $t < cx$ , where $c \in Z \setminus \{0\}$ and t is a term not containing x, with $t/c < x$ .
- Atoms in the output $\exists x.\ F3[x]$ now have the form

(A) $x < a$

(B) $b < x$

(C) $x = c$

- where a, b, c are terms that do not contain x. $\exists x.\ F3[x]$ is $T_Q$-equivalent to $\exists x.\ F[x]$.

# Ferrante & Rackoff's method



- Step 4
- Construct the left infinite projection $F_{-\infty}$ from $F_3[x]$ by replacing

(A) atoms $x < a$ by $\top$ ,

(B) atoms $b < x$ by $\bot$ , and

(C) atoms $x = c$ by $\bot$ .

- Construct the right infinite projection $F_{+\infty}$ from $F_3[x]$ by replacing

(A) atoms $x < a$ by $\bot$ ,

(B) atoms $b < x$ by $\top$ , and

(C) atoms $x = c$ by $\bot$ .

# Ferrante & Rackoff's method



- The left (right) infinite projection captures the case when small (large) $n \in Q$ satisfy $F3[n]$.
- Let S be the set of a, b, and c terms from the (A), (B), and (C) atoms.
- Construct the final output
- $F4 : F_{-\infty} \vee F_{+\infty} \vee \bigvee_{s,t \in S} F3\,[(s + t)/2]$
- which is TQ-equivalent to $\exists x.\ F[x]$.

# Ferrante & Rackoff's method (example)



- Consider the ΣQ-formula
- $\exists x.\ 2x = y$ : $F[x]$

- In Step 3, solving for x produces

$F' : \exists x.\ x = y/2$

- so that $S = \{y/2\}$.
- The left $F_{-\infty}$ and right $F_{+\infty}$ infinite projections are both $\perp$, as $F'$ contains a single (C) atom.

# Ferrante & Rackoff's method (example)



- Hence, simplifying

- F4 : $\bigvee_{s,t \in S} [(s + t)/2 = y/2]$

- reveals the TQ-equivalent quantifier-free formula $y/2 = y/2$ , or $\top$. Therefore, $\exists x.\ F[x]$ is TQ-valid.

# Ferrante & Rackoff's method (example 2)



- Consider the $\Sigma_Q$-formula
- $\exists x.\ 3x + 1 < 10 \wedge 7x - 6 > 7\}$ : $F[x]$

- Solving for x gives
- $F'$ : $\exists x.\ x < 3 \wedge x > 13 / 7$ : $F3[x]$
- and $S = \{3, 13/7\}$.

# Ferrante & Rackoff's method (example 2)



- Since $x < 3$ is an (A) atom and $x > 13/7$ is a (B) atom, both $F_{-\infty}$ and $F_{+\infty}$ simplify to $\perp$, leaving

$$F4 : \bigvee_{s,t \in S} [(s+t)/2 < 3 \wedge (s+t)/2 > 13/7]$$

- $(s+t)/2$ takes on three expressions: $3$, $13/7$ , and $(13/7+3)/2$ .

- The first two expressions arise when s and t are the same terms. F3[3] and F3[ 13/7 ] both simplify to $\perp$ since the inequalities are strict;

# Ferrante & Rackoff's method (example 2)



- however,

- F3 [ (13/7 + 3)/2] :

$(13/7 + 3)/2 < 3 \wedge (13/7 + 3)/2 > 13/7$ simplifies to $\top$.

- Thus, F4 : $\top$ is $T_Q$-equivalent to $\exists x.\ F[x]$, so $\exists x.\ F[x]$ is $T_Q$-valid.

# Ferrante & Rackoff's method (example 3)



- Consider the $\Sigma_Q$-formula G :
- $\forall x.\ x < y$ .
- To eliminate x, consider the subformula F of
- $G' : \neg(\exists x.\ \neg(x < y) \mid \{z\}$ : $F[x]$

- Step 2 rewrites F as
- $\exists x.\ y < x \vee y = x$ .

# Ferrante & Rackoff's method (example 3)



- The literals are already in solved form for x in Step 3.

$F_{-\infty} : \bot \vee \bot$ and $F_{+\infty} : \top \vee \bot$

- simplify to $\bot$ and $\top$, respectively.
- Since $F_{+\infty}$ is $\top$, we need not consider the rest of Step 4, but instead declare that

$\exists x.\ F[x]$ is $T_Q$-equivalent to $F_4 : \top$.

- Then $G'$ is $\neg\top$, so that $G$ is $T_Q$-equivalent to $\bot$.

# Ferrante & Rackoff's method (Theorem)



- **Theorem 2** : Given $\Sigma_Q$-formula $\exists x.\ F[x]$ in which F is quantifier-free, Ferrante and Rackoff's method returns a $T_Q$-equivalent quantifier-free formula.

(Proof very similar to proof of Cooper's method)

- **Theorem 3 :** On a ΣQ-formula of length n, Ferrante and Rackoff's method requires deterministic time $2^{2^{pn}}$ for some fixed constant $p > 0$.

# Outline



1) Motivating example
2) Formal Description
3) Cooper’s method
4) Ferrante & Rackoff’s method
5) Summary

# Summary : Complexity



- Fischer and Rabin proved the following lower bounds.
  The length n of a formula is the number of symbols.
- **Theorem** ($T_Z$ Lower Bound). There is a fixed constant c > 0 such that for all sufficiently large n, there is a $\Sigma_Z$-formula of length n that requires at least 2^2^cn steps to decide its validity.
- **Theorem** ($T_Q$ Lower Bound). There is a fixed constant c > 0 such that for all sufficiently large n, there is a $\Sigma_Q$-formula of length n that requires at least 2^cn steps to decide its validity.

# Summary : Complexity



Oppen analyzed Cooper's method to prove the following upper bound.

- **Theorem** ($T_Z$ Upper Bound). On a $\Sigma_Z$-formula of length n, Cooper's method requires deterministic time 2^2^2^pn for some fixed constant p > 0.

Ferrante and Rackoff proved the following upper bound.

- **Theorem** ($T_Q$ Upper Bound). On a $\Sigma_Q$-formula of length n, Ferrante and Rackoff's method requires deterministic time 2^2^pn for some fixed constant p > 0.

# Summary



- Quantifier elimination is a standard technique for reasoning about theories in which satisfiability is decidable even with arbitrary quantification.
- Based on structural induction, one only needs to consider the special case of formulae of the form $\exists x.\ F[x]$, in which $F$ is quantifier-free but may contain free variables in addition to $x$; arbitrary formulae may then be treated compositionally.
- Closing the gap between the lower and upper bounds would require answering long-standing open questions in complexity theory.

# Summary



- Elimination over integers, $T_Z$. ($\exists x.\ 2x = y$ ? $2 \mid y$ )

- The basic theory of integers does not admit quantifier elimination; it must be augmented with divisibility predicates. This situation, in which additional predicates are required to develop a quantifier elimination procedure, is common. The main idea of the procedure is to identify intervals with periodic behavior induced by the divisibility predicates.

- Elimination over rationals, $T_Q$.
  The main idea of the procedure is to partition the rationals into a finite number of points and intervals.

# Thank you

## for your attention !