Small Proofs from Congruence Closure

Oliver Flatt, Samuel Coward, Max Willsey,
Zachary Tatlock, Pavel Panchekha
Solvers and Proofs
Solvers and Proofs

Problem
Is \(a+b = b+a\)?

Solver
Solvers and Proofs

Problem
Is $a + b = b + a$?

Solver

Solution
YES
Solvers and Proofs

Problem
Is a+b = b+a?

Solver

Solution
YES

Complex
Solvers and Proofs

Problem
Is \(a+b = b+a\)?

Solver

Solution
YES

Proof
\(a+b = b+a\) by commutativity

Complex
Solvers and Proofs

Problem
Is \( a+b = b+a? \)

Solver
Complex

Solution
YES

Proof
\( a+b = b+a \) by commutativity

Proof Checker
Solvers and Proofs

Problem
Is $a+b = b+a$?

Solver

Solution
YES

Proof
$a+b = b+a$
by commutativity

Proof Checker
Simple
Solvers and Proofs

Problem:
Is a+b = b+a?

Solver:

Solution:
YES

Proof:

Proof Checker:

a+b = b+a by commutativity

Complex

Simple
Proofs are Useful
Proofs are Useful

Checking
Proofs are Useful

Checking

Can we trust the solver?
Proofs are Useful

- Checking
  - Can we trust the solver?
- Debugging
Proofs are Useful

Checking

Can we trust the solver?

Debugging

How did we prove 0 = 1?
Proofs are Useful

Checking
Can we trust the solver?

Debugging
How did we prove $0 = 1$?

CDCL
Proofs are Useful

Checking
Can we trust the solver?

Debugging
How did we prove $0 = 1$?

CDCL
What facts led to this result?
Proofs are Useful

Checking
Can we trust the solver?

Debugging
How did we prove 0 = 1?

CDCL
What facts led to this result?

...And More
Fuzzing
Optimization
Proofs can be Long

Checking
Can we trust the solver?

Debugging
How did we prove 0 = 1?

CDCL
What facts led to this result?

...And More
Fuzzing
Optimization
Proofs can be Long

Checking
Okay

Debugging
How did we prove $0 = 1$?

CDCL
What facts led to this result?

...And More
Fuzzing
Optimization
Proofs can be Long

Checking
Okay

Debugging
Confusing

CDCL
What facts led to this result?

...And More
Fuzzing
Optimization
Proofs can be Long

- Checking
  - Okay

- Debugging
  - Confusing

- CDCL
  - Too Specific

- And More
  - Fuzzing
  - Optimization
Proofs can be Long

Checking
- Okay

Debugging
- Confusing

CDCL
- Too Specific

...And More
- Slow
Proofs can be Long

This Talk:
Finding smaller proofs from congruence closure
Why Congruence Closure?
Why Congruence Closure?

**Congruence Closure** forms the basis of many solvers
Why Congruence Closure?

**Congruence Closure** forms the basis of many solvers.

Generates all proofs of **equality**.
Why Congruence Closure?

**Congruence Closure** forms the basis of many solvers

Generates all proofs of **equality**

Enables **equality saturation**

Optimization and synthesis
Why Congruence Closure?

**Congruence Closure** forms the basis of many solvers.

Generates all proofs of *equality*.

Enables *equality saturation*.

Optimization and synthesis.

Our library: egg 🥚.
Why Congruence Closure?

**Congruence Closure** forms the basis of many solvers

Generates all proofs of **equality**

Enables **equality saturation**

Optimization and synthesis

Our library: egg 🥚
Motivation

Congruence Closure

Proofs from Congruence Closure

Finding Small Proofs
Motivation

Congruence Closure

Proofs from Congruence Closure

Finding Small Proofs

27.2% smaller proofs!
Congruence Closure
Congruence Closure

Input: equalities between terms

\[ a = b \]
\[ f(a) = f(b) \]
\[ b = c \]
Congruence Closure

Input: equalities between terms
a = b
f(a) = f(b)
b = c

Output: equivalence relation
stored in an e-graph data structure
Ask: is a = c?
**Congruence Closure**

**Input:** equalities between terms
- \( a = b \)
- \( f(a) = f(b) \)
- \( b = c \)

**Output:** equivalence relation stored in an **e-graph** data structure
**Ask:** is \( a = c \)?

The relation is also closed under **congruence**

\[
\forall x, y: x = y \Rightarrow f(x) = f(y)
\]
E-Graph Example

A graph with 3 kinds of edges
E-Graph Example

A graph with 3 kinds of edges

Inputs:

\[ f(a) \]

\[ f(b) \]

\[ a = b \]
E-Graph Example

A graph with 3 kinds of edges

Inputs:

- f(a)
- f(b)
- a = b
E-Graph Example

A graph with 3 kinds of edges

Inputs:

f(a)  →  f
f(b)  →  a = b

f  →  a
f  →  b
E-Graph Example

A graph with 3 kinds of edges

Inputs:
- f(a)
- f(b)
- a = b
E-Graph Example

A graph with 3 kinds of edges

Inputs:
- f(a)
- f(b)
- a = b
E-Graph Example

A graph with 3 kinds of edges

Inputs:
- f(a)
- f(b)
- a = b

Diagram:
- Congruence edge: f \rightarrow f
given congr(a, b)
- Equality edge: a \rightarrow b
- Parent-child edge: parent-child edge
E-Graph Example

A graph with 3 kinds of edges

equality edge from congruence

congr(a, b)

Inputs:

f(a)

f(b)

a = b

parent-child edge

equality edge
A Bigger E-Graph Example
A Bigger E-Graph Example

Inputs:

\[
\begin{align*}
& f(c) \\
& f(a) = f(b) \\
& a = b \\
& b = c \\
& a = c
\end{align*}
\]
A Bigger E-Graph Example

Inputs:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \]
A Bigger E-Graph Example

Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$
A Bigger E-Graph Example

Inputs:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \]
A Bigger E-Graph Example

Inputs:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \]
A Bigger E-Graph Example

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$
A Bigger E-Graph Example

Inputs:
- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \)  [unnecessary]
A Bigger E-Graph Example

Inputs:
- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \) **unnecessary**

Are \( a \) and \( c \) equal? **Yes!**

Are \( f(a) \) and \( a \)? **No!**

Are \( f(a) \) and \( f(c) \)? **Yes!**
A Bigger E-Graph Example

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ (unnecessary)

Are $a$ and $c$ equal? Yes!
Are $f(a)$ and $a$? No!
Are $f(a)$ and $f(c)$? Yes!

Key idea: equality edges form equivalence classes of terms
Example:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \text{ unnecessary} \]
Example:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \text{ unnecessary} \]

Reality:
Example:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \]

Reality:
Motivation

Congruence Closure

Proofs from Congruence Closure

Finding Small Proofs

27.2% smaller proofs!
Congruence Proofs
Congruence Proofs

Answer the question "how are these two terms equal?"
Congruence Proofs

Answer the question "how are these two terms equal?"

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$  **unnecessary**
Congruence Proofs

Answer the question "how are these two terms equal?"

Inputs:
- \(f(c)\)
- \(f(a) = f(b)\)
- \(a = b\)
- \(b = c\)
- \(a = c\) [unnecessary]

Prove \(a\) and \(c\) are equal:
Congruence Proofs

Answer the question "how are these two terms equal?"

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ **unnecessary**

Prove $a$ and $c$ are equal:
- $a = b$
Congruence Proofs

Answer the question "how are these two terms equal?"

Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ [unnecessary]

Prove $a$ and $c$ are equal:

- $a = b$
- $b = c$
Congruence Proofs

Answer the question "how are these two terms equal?"

Inputs:
- $f(c)
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ (unnecessary)

Prove $a$ and $c$ are equal:
- $a = b$
- $b = c$
- done!
Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ **unnecessary**

Prove $f(a)$ and $f(c)$ are equal:
Inputs:

\[ f(c) \]
\[ f(a) = f(b) \]
\[ a = b \]
\[ b = c \]
\[ a = c \] \textcolor{red}{\text{unnecessary}}

Prove \( f(a) \) and \( f(c) \) are equal:

\[ f(a) = f(b) \]
Inputs:

- f(c)
- f(a) = f(b)
- a = b
- b = c
- a = c  **unnecessary**

Prove f(a) and f(c) are equal:

- f(a) = f(b)

Prove f(b) = f(c) by congruence:
Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ \textbf{unnecessary}

Prove $f(a)$ and $f(c)$ are equal:

- $f(a) = f(b)$

Prove $f(b) = f(c)$ by congruence:

- $b = c$
Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ **unnecessary**

Prove $f(a)$ and $f(c)$ are equal:

- $f(a) = f(b)$

Prove $f(b) = f(c)$ by congruence:

- $b = c$

done!
Inputs:
- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \) **unnecessary**

Prove \( f(a) \) and \( f(c) \) are equal:

- \( f(a) = f(b) \)

Prove \( f(b) = f(c) \) by congruence:

- \( b = c \)

**done!**
Prove $f(a)$ and $f(c)$ are equal:

$$f(a) = f(b)$$

Prove $f(b) = f(c)$ by congruence:

$$b = c$$

done!
We define **proof size** as the number of **unique** equalities in the proof.

Prove \( f(a) \) and \( f(c) \) are equal:

\[
f(a) = f(b)
\]

Prove \( f(b) = f(c) \) by congruence:

\[
b = c
\]

done!
We define **proof size** as the number of **unique** equalities in the proof.

This proof: size 2

Prove $f(a)$ and $f(c)$ are equal:

- $f(a) = f(b)$
- Prove $f(b) = f(c)$ by congruence:
- $b = c$

Done!
We define **proof size** as the number of **unique** equalities in the proof.

This proof: size 2

Prove $f(a)$ and $f(c)$ are equal:

- $f(a) = f(b)$
- Prove $f(b) = f(c)$ by congruence:
  - $b = c$
  - done!

We can do better!
Leveraging Additional Equalities

Inputs:

- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \) [unnecessary]
Leveraging Additional Equalities

Inputs:
- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \) useful
Leveraging Additional Equalities

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$  

![Diagram showing relationships between variables and functions]
Leveraging Additional Equalities

Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$  

The diagram shows the relationships:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$  

Useful relationships highlighted.
Leveraging Additional Equalities

Inputs:
- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \) useful

Prove \( f(a) \) and \( f(c) \) are equal:
Leveraging Additional Equalities

Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ (useful)

Prove $f(a)$ and $f(c)$ are equal:

Prove $f(a) = f(c)$ by congruence:
Leveraging Additional Equalities

Inputs:
- f(c)
- f(a) = f(b)
- a = b
- b = c
- a = c **useful**

Prove f(a) and f(c) are equal:

Prove f(a) = f(c) by congruence:
- a = c
Leveraging Additional Equalities

Inputs:
- \( f(c) \)
- \( f(a) = f(b) \)
- \( a = b \)
- \( b = c \)
- \( a = c \) \text{ useful}

Prove \( f(a) \) and \( f(c) \) are equal:

Prove \( f(a) = f(c) \) by congruence:
- \( a = c \)

done!
Leveraging Additional Equalities

Inputs:

\[
\begin{align*}
f(c) \\
f(a) &= f(b) \\
a &= b \\
b &= c \\
a &= c \text{ useful}
\end{align*}
\]

Prove \( f(a) \) and \( f(c) \) are equal:

Prove \( f(a) = f(c) \) by congruence:

\[
\begin{align*}
a &= c
\end{align*}
\]

done!

Proof size: 1 😊
Leveraging Additional Equalities

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$  
  **useful**

Prove $f(a)$ and $f(c)$ are equal:

Prove $f(a) = f(c)$ by congruence:

- $a = c$

done!

Proof size: 1 🤖
Leveraging Additional Equalities

Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ [useful]

Prove $f(a)$ and $f(c)$ are equal:

Prove $f(a) = f(c)$ by congruence:

- $a = c$

done!

Proof size: 1 🧪

Key idea:
Try alternate path

new!
The Crux of The Problem

Prove $a = e$

Diagram:

- $a$
- $f$
- $f$
- $b$
- $c$
- $d$
- $e$
The Crux of The Problem

Prove $a = e$
The Crux of The Problem

Prove $a = e$
The Crux of The Problem

Prove $a = e$

Bottom: size 4

Diagram:
- $a$ connected to $f$ via $congr(c, d)$
- $f$ connected to $congr(d, c)$
- $congr(d, c)$ connected to $c$
- $c$ connected to $d$
- $d$ connected to $f$
- $f$ connected to $e$

Bottom nodes: $a$, $b$, $c$, $d$, $e$
The Crux of The Problem

Prove $a = e$

Bottom: size 4

Top: ???

Diagram:
- $a$ connected to $b$ and $c$
- $b$ connected to $d$ and $e$
- $c$ connected to $d$ and $f$
- $d$ connected to $f$
- $f$ connected to $c$ and $d$

Relations:
- $\text{congr}(c, d)$
- $\text{congr}(d, c)$
The Crux of The Problem

Prove \( a = e \)

Bottom: size 4

Top: size 3
The Crux of The Problem

Prove $a = e$

Bottom: size 4

Top: size 3

Proof re-use is free
The Crux of The Problem

Prove \( a = e \)

Bottom: size 4

Top: size 3

Proof re-use is free

Subproofs can be complex
The Crux of The Problem

Prove $a = e$

Bottom: size 4

Top: size 3

Proof re-use is free

Subproofs can be complex

Exponential paths in graph
The Crux of The Problem

Prove $a = e$

Bottom: size 4

Top: size 3

Proof re-use is free

Subproofs can be complex

NP-Hard Problem (Fellner Et al.)

Exponential paths in graph
Motivation

Congruence Closure

Proofs from Congruence Closure

Finding Small Proofs

27.2% smaller proofs!
Idea: Shortest Path?

Inputs:
- f(c)
- f(a) = f(b)
- a = b
- b = c
- a = c (useful)
Idea: Shortest Path?

Inputs:
- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$ [useful]

Diagram:
- $f$ to $b$ with $\text{congr}(b, c)$
- $f$ to $a$ with $\text{congr}(a, c)$
- $a$ to $b$ with $\text{congr}(a, c)$
- $f$ to $f$

Question: shortest path?
Idea: Shortest Path?

Inputs:

- $f(c)$
- $f(a) = f(b)$
- $a = b$
- $b = c$
- $a = c$

Problem: how big are congruence edges?

![Diagram showing relationships between nodes and edges with labels such as congr(b, c) and congr(a, c).]
Proof Size Estimation

Inputs:

\[ a = b \]
\[ g(b) = c \]
Proof Size Estimation

Inputs:

\[ a = b \]
\[ g(b) = c \]
Proof Size Estimation

Inputs:

\[ a = b \]

\[ g(b) = c \]
Proof Size Estimation

Inputs:

\( a = b \)

\( g(b) = c \)
Proof Size Estimation

Inputs:

\[ a = b \]

\[ g(b) = c \]
Proof Size Estimation

Inputs:

\[ a = b \]

\[ g(b) = c \]

\[ \text{congr}(g(a), c) \]

\[ \text{congr}(a, b) \]

Key idea: compute estimates bottom-up
Proof Size Estimation

Inputs:
\[ a = b \]
\[ g(b) = c \]

\[
\begin{align*}
& \text{congr}(g(a), c) \\
& \text{congr}(a, b) \quad 1 \\
& a \quad 1 \\
& b \\
& g \\
& g \\
& f \quad f
\end{align*}
\]

Key idea: compute estimates bottom-up
Proof Size Estimation

Inputs:

\[ a = b \]

\[ g(b) = c \]

Key idea: compute estimates bottom-up
Proof Size Estimation

Inputs:

\[ a = b \]
\[ g(b) = c \]

Key idea: compute estimates bottom-up
Proof Size Estimation

Inputs:

- $a = b$
- $g(b) = c$

Key idea: compute estimates bottom-up
Proof Size Estimation

Inputs:

\[ a = b \]
\[ g(b) = c \]

Key idea: compute estimates bottom-up
Proof Size Estimation

Inputs:

\[ a = b \]
\[ g(b) = c \]

Key idea: compute estimates bottom-up
Putting it All Together

\begin{align*}
\text{c} & \congr \text{g(a), c} \\
\text{g} & \congr \text{a, b} \\
\text{a} & \text{b}
\end{align*}
Putting it All Together

1. Compute size estimates

Diagram:
- Node f connected to g by congr(g(a), c)
- Node g connected to a by congr(a, b)
- Node g connected to b
- Node f connected to c
Putting it All Together

1. Compute size estimates
   - congr(g(a), c)
   - congr(a, b)

2. Find shortest path
   - f
   - g
   - a
   - b
   - c
Putting it All Together

1. Compute size estimates
2. Find shortest path
3. Output extracted proof

\[
\text{congr}(g(a), c)
\]

\[
\text{congr}(a, b)
\]
Results

Number of Benchmarks (Cumulative)

DAG Size of Proof Certificate

better
Results

![Graph showing comparison between different methods, labeled as Greedy, Z3, and Unoptimized. The graph indicates that Greedy is better than the others in terms of the number of benchmarks achieved for a given DAG size of proof certificate.](image)
Multi-operation circuit optimization and translation validation with egg 🥚
Intel Case Study

Multi-operation circuit optimization and translation validation with egg 🥚

4.7 hours -> 2.3 hours
Intel Case Study

Multi-operation circuit optimization and translation validation with egg
4.7 hours -> 2.3 hours

![Diagram of a circuit with 2 inputs and 5 outputs, showing operations and outputs like 5a + b, 4a + 2b, 3a + 3b, 2a + 4b, and a + 5b.](image)
Team and Acknowledgments

Oliver Flatt  Samuel Coward  Max Willsey  Zachary Tatlock  Pavel Panchekha

Special thanks to:

Theo Drane (Intel)
George A. Constantinides (Imperial College)
Leonardo de Moura (Microsoft)
Questions?

1. Compute size estimates
2. Find shortest path
3. Output extracted proof

congr(a, b)
congr(g(a), c)

oflatt@cs.washington.edu