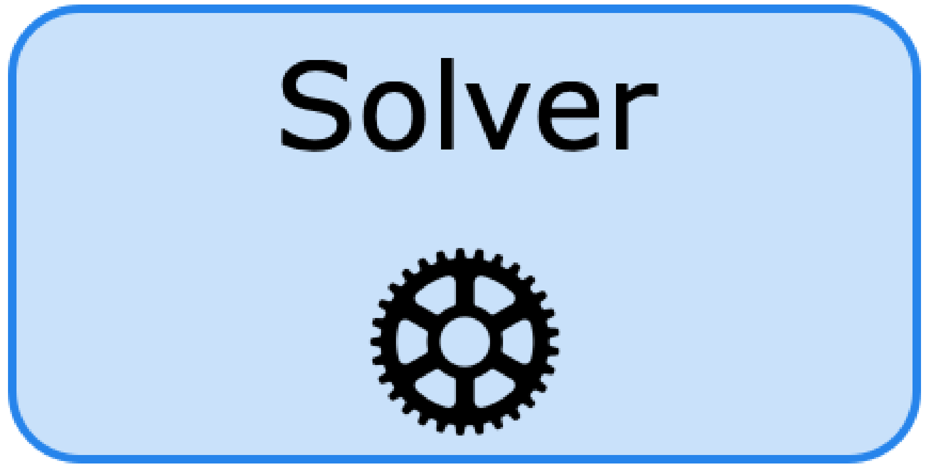
Small Proofs from Congruence Closure

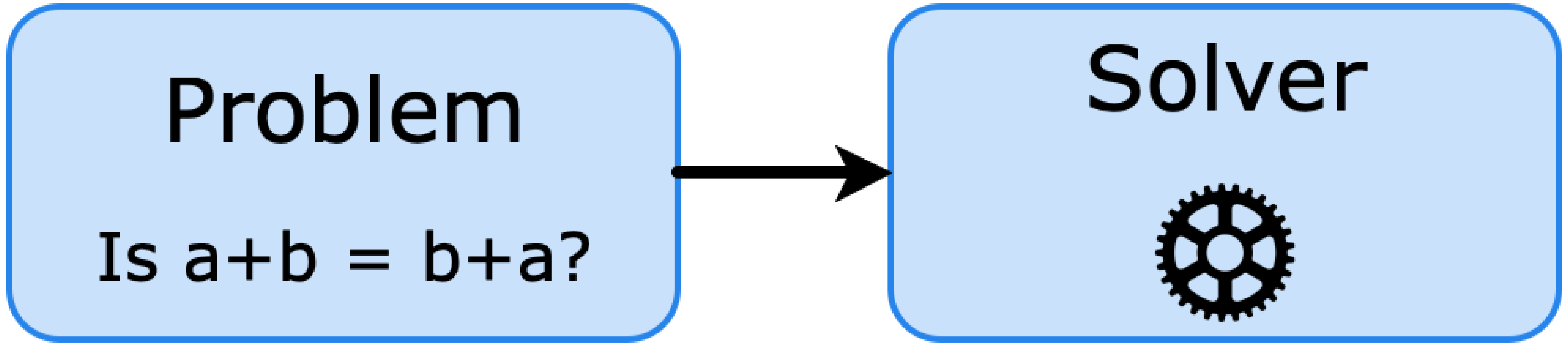
Oliver Flatt, Samuel Coward, Max Willsey,

Zachary Tatlock, Pavel Panchekha

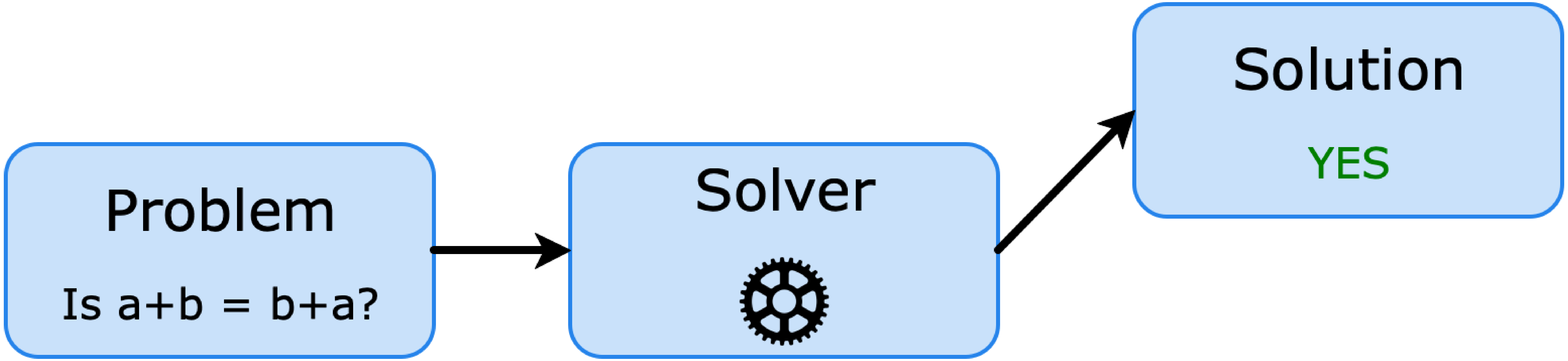
Solvers and Proofs



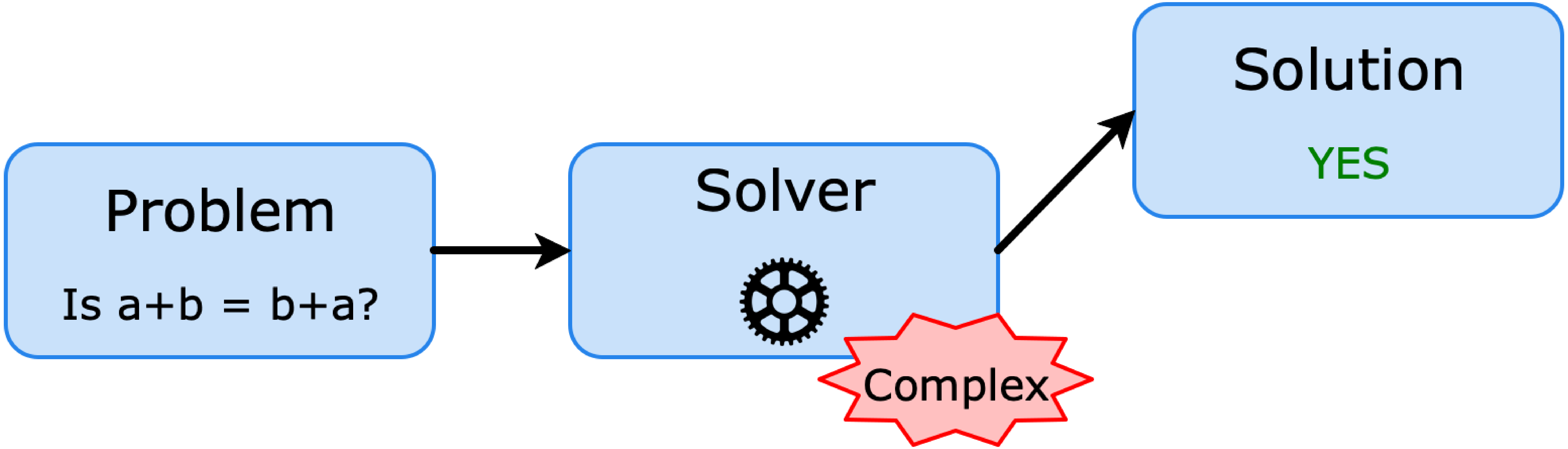
Solvers and Proofs



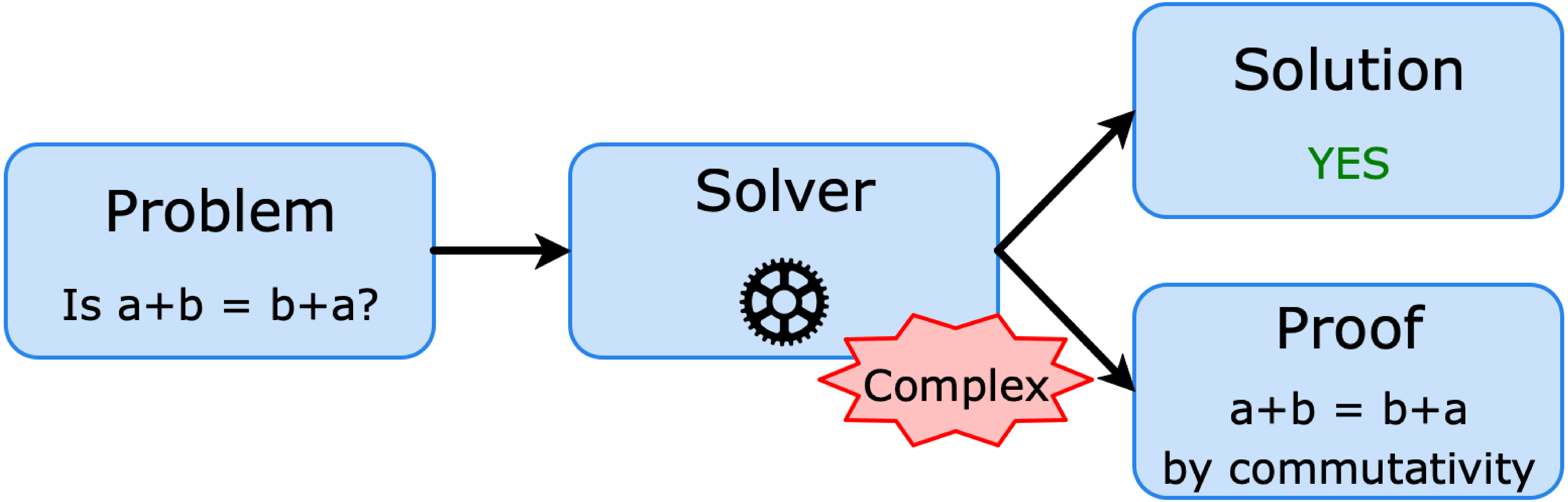
Solvers and Proofs



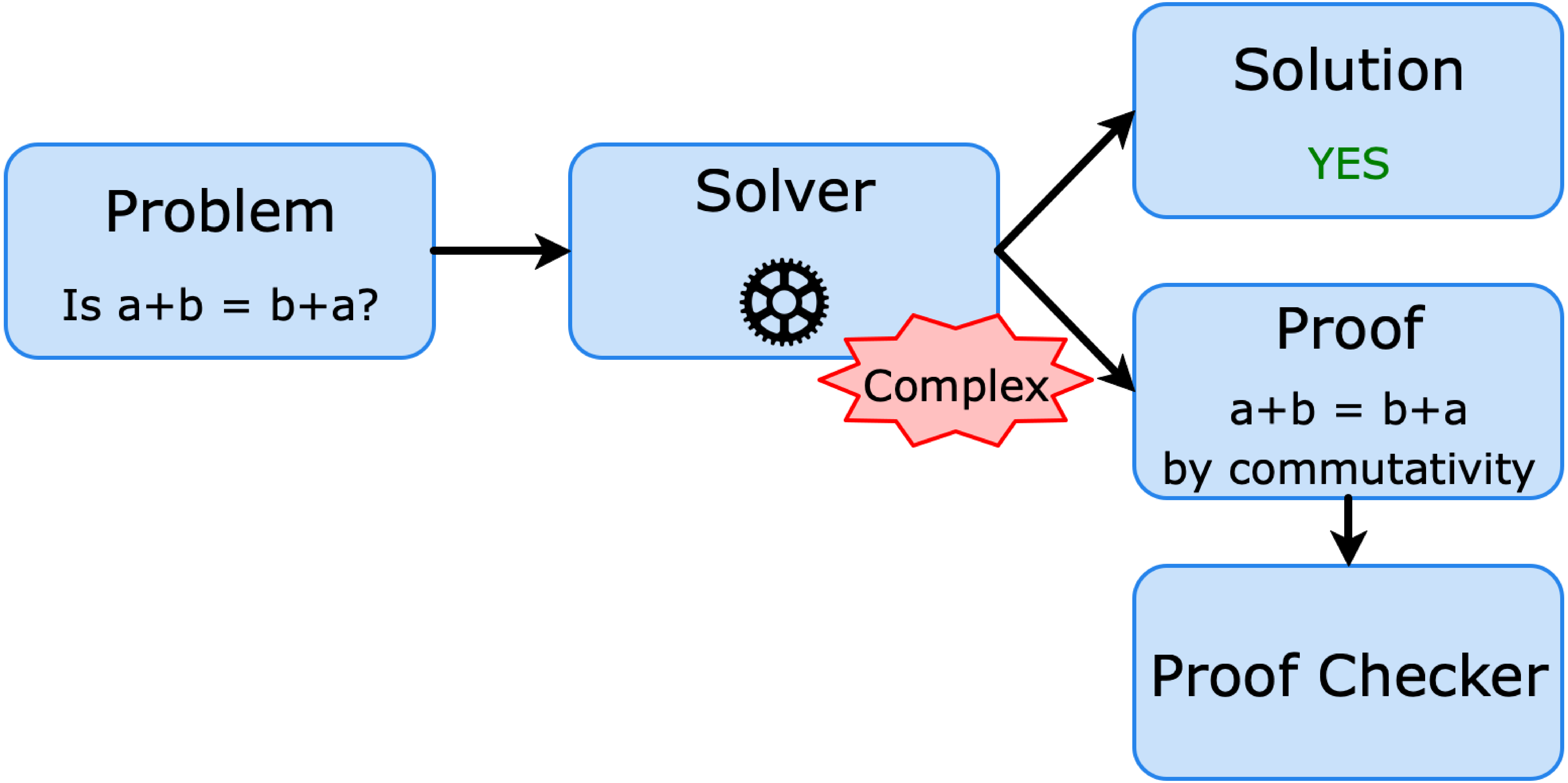
Solvers and Proofs



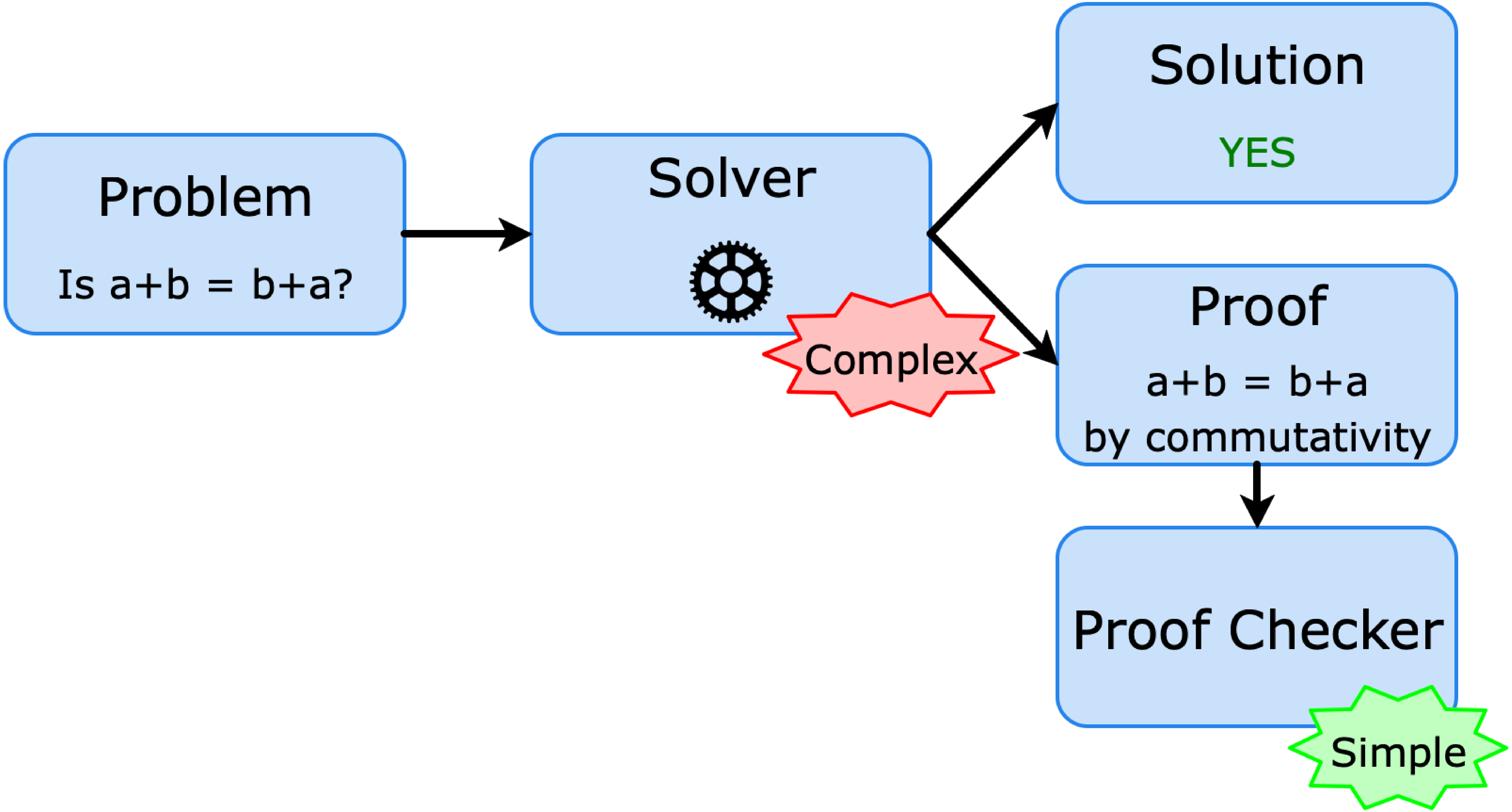
Solvers and Proofs



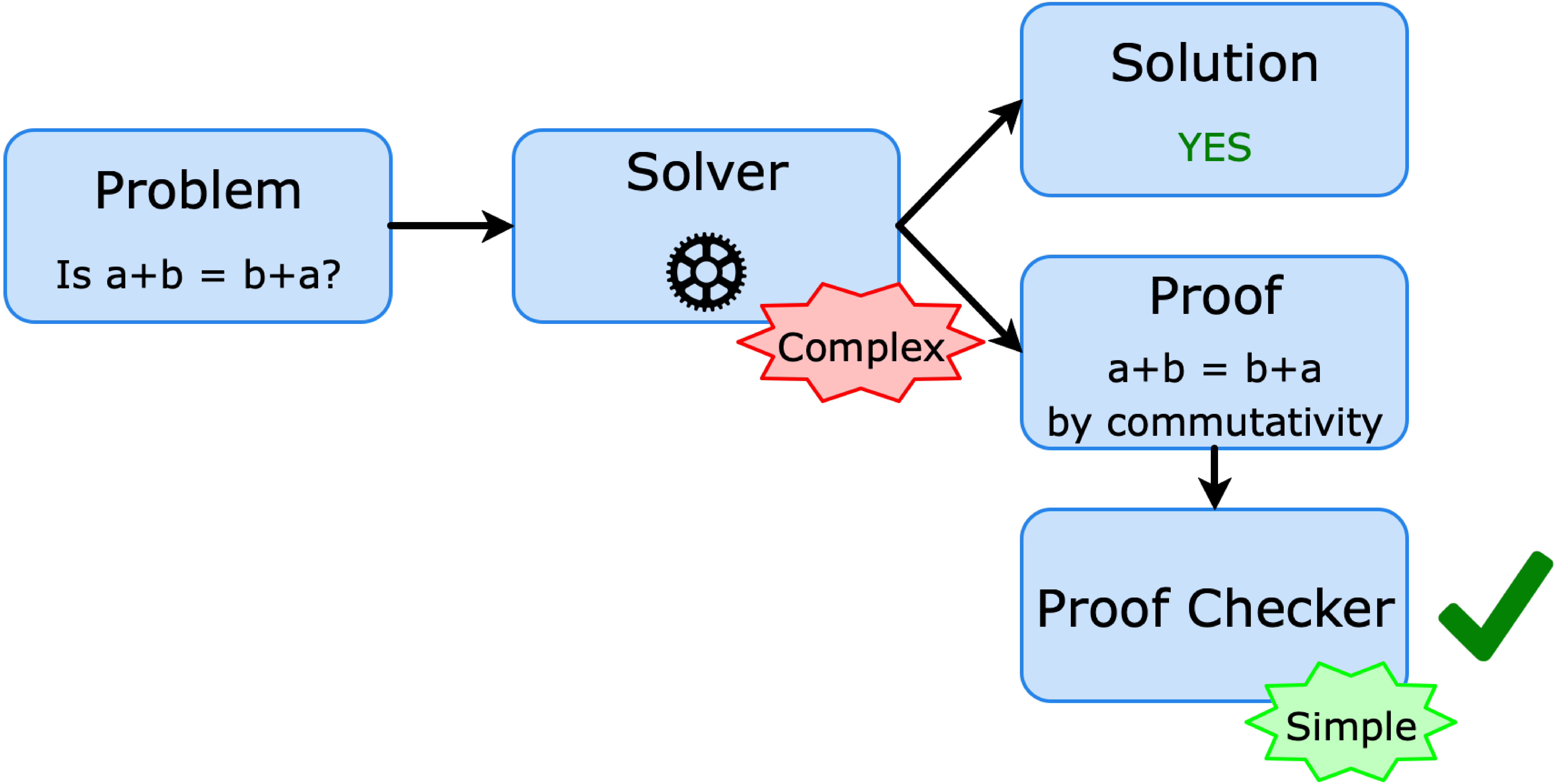
Solvers and Proofs



Solvers and Proofs



Solvers and Proofs



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

- [x] Motivation
- [ ] Congruence Closure
- [ ] Proofs from Congruence Closure
- [ ] Finding Small Proofs

- [x] 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

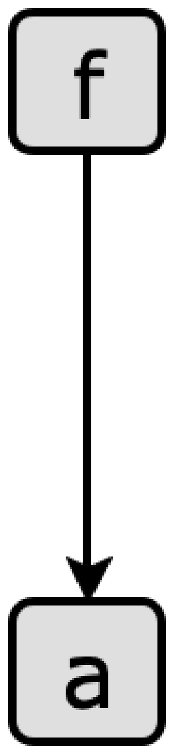
A graph with **3** kinds of edges

Inputs:

f(a)

f(b)

a = b



E-Graph Example

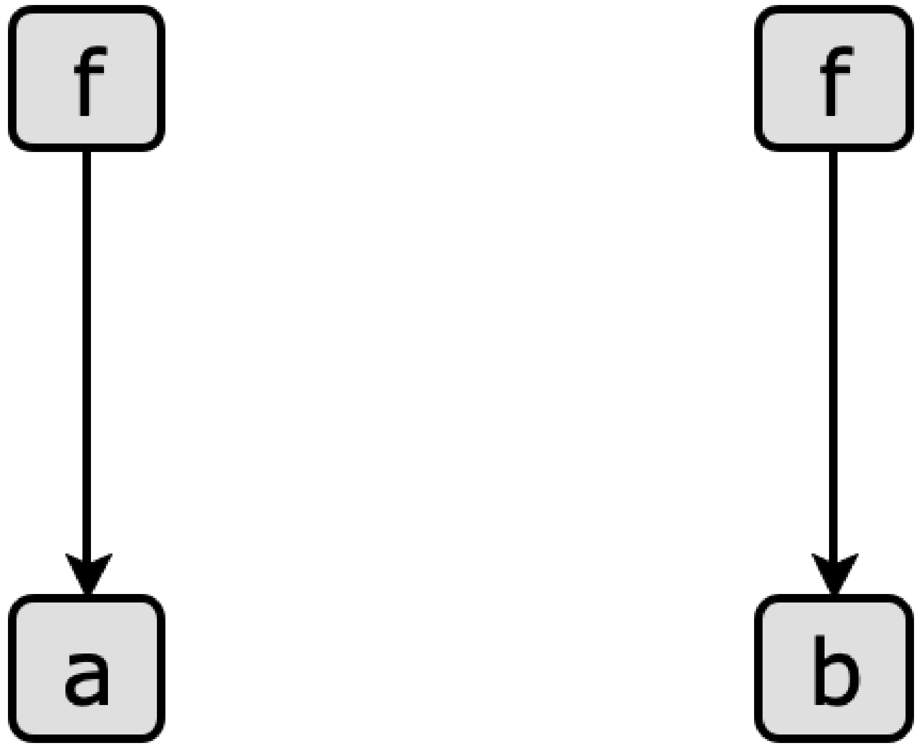
A graph with **3** kinds of edges

Inputs:

f(a)

f(b)

a = b



E-Graph Example

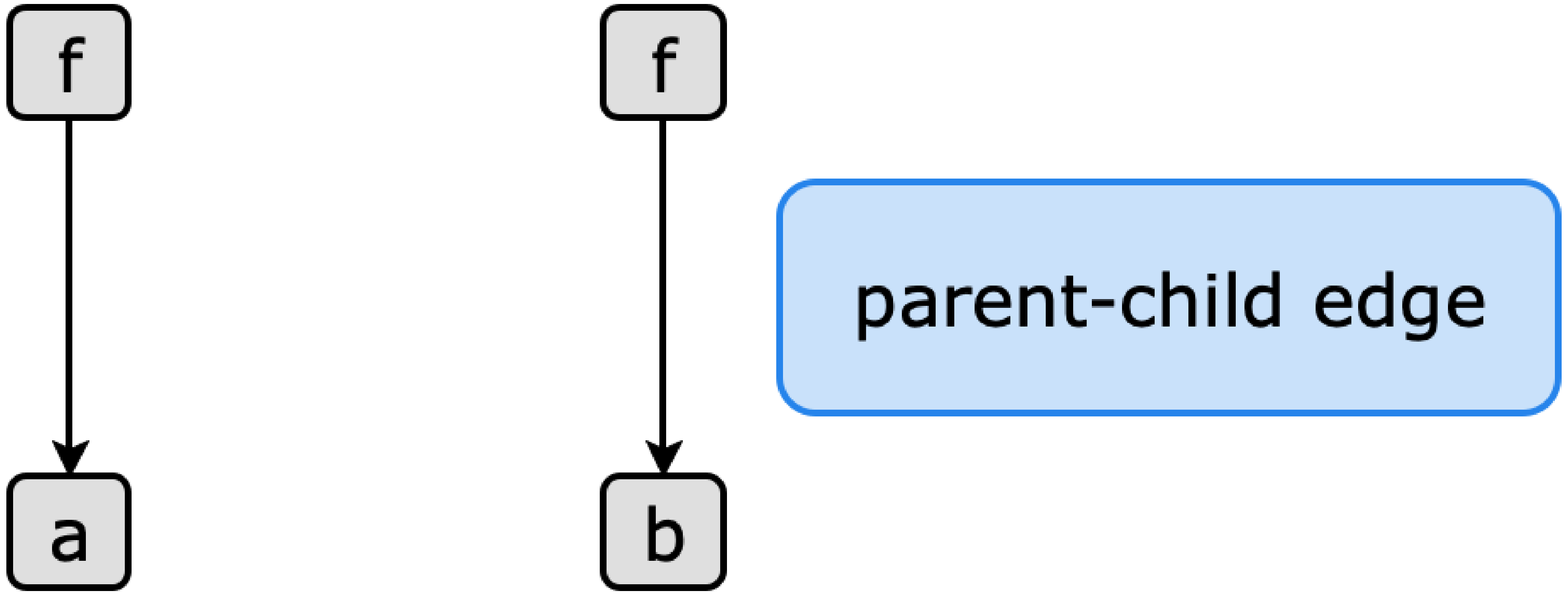
A graph with **3** kinds of edges

Inputs:

f(a)

f(b)

a = b



E-Graph Example

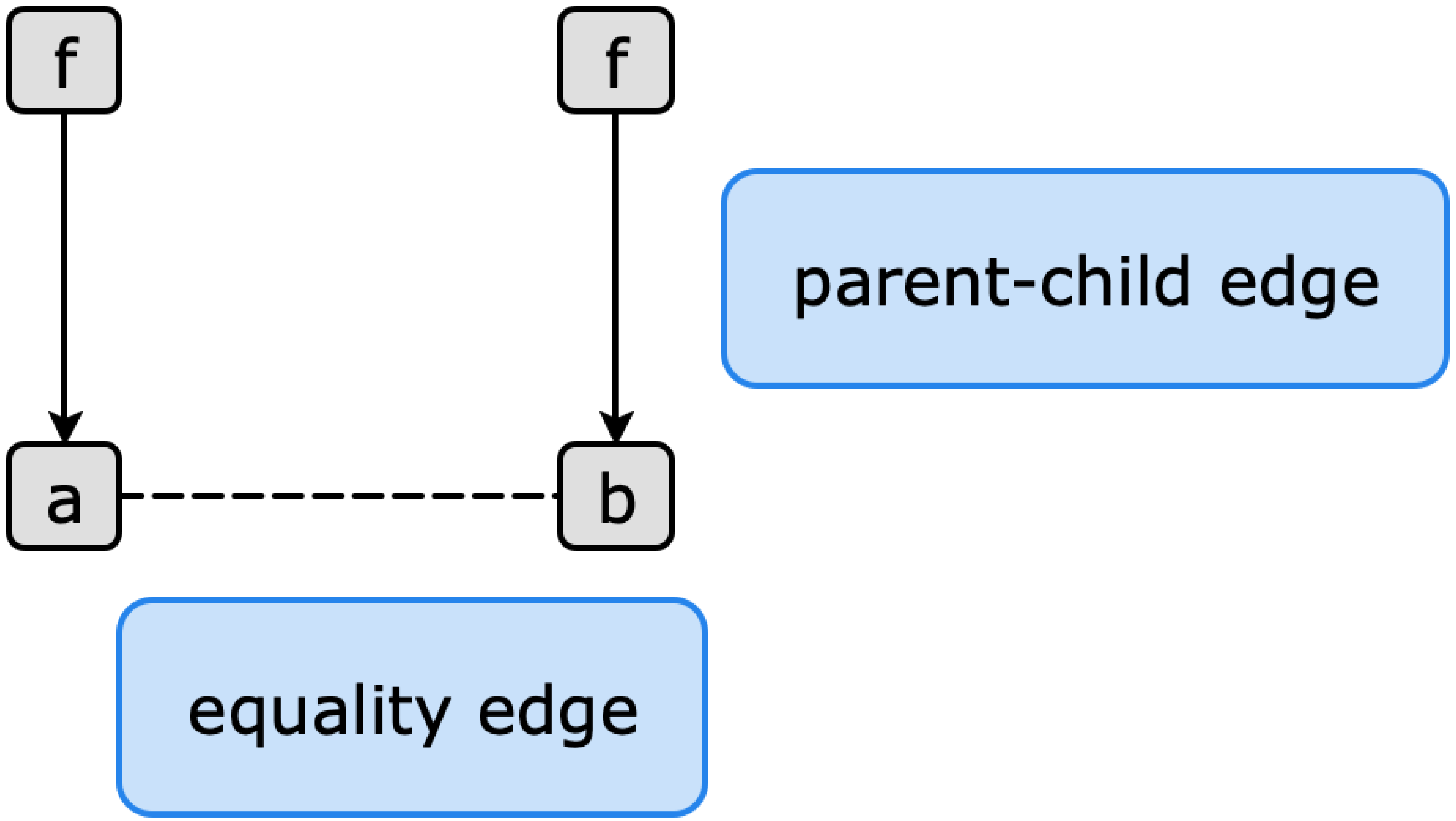
A graph with **3** kinds of edges

Inputs:

f(a)

f(b)

a = b



E-Graph Example

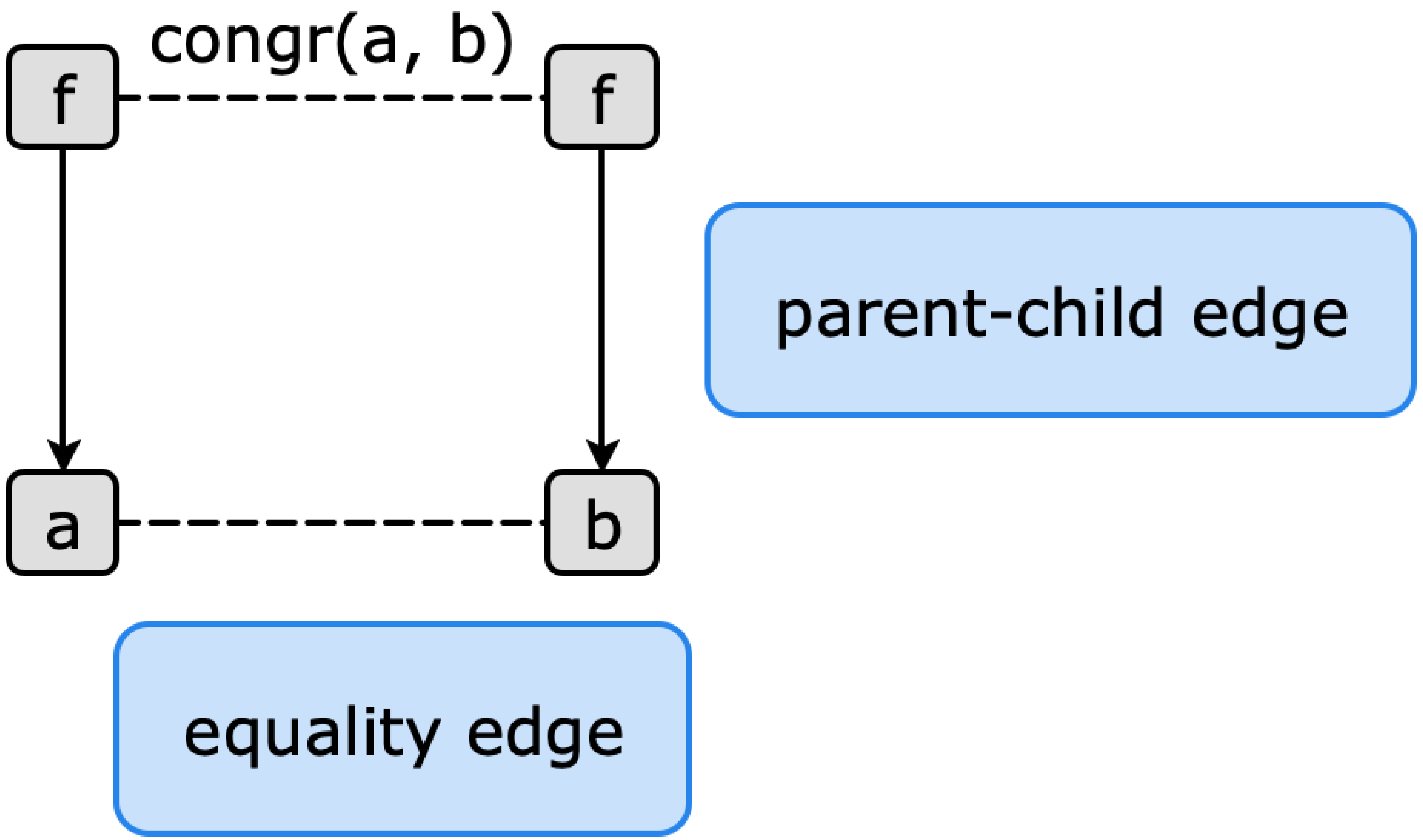
A graph with **3** kinds of edges

Inputs:

f(a)

f(b)

a = b



E-Graph Example

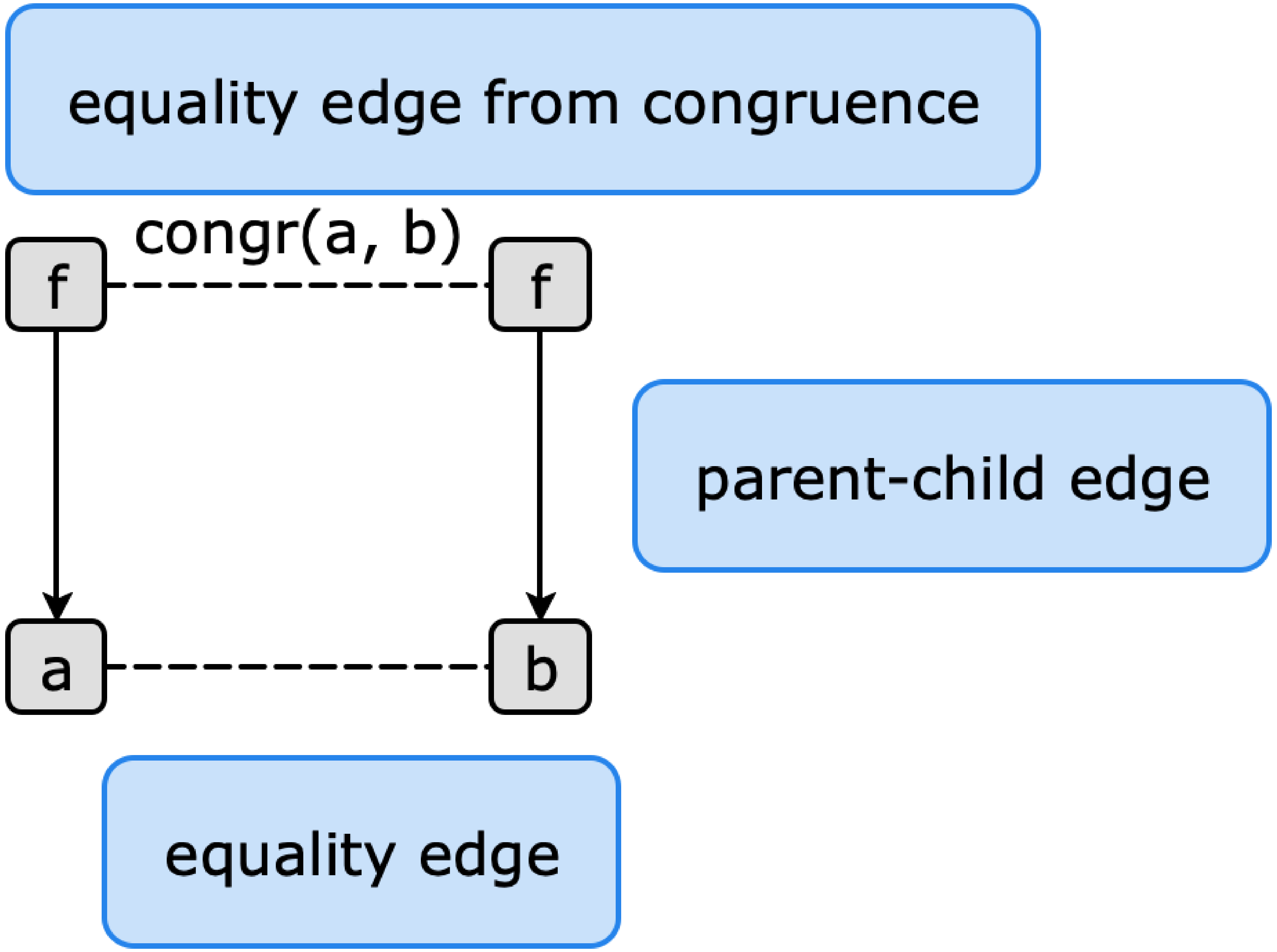
A graph with **3** kinds of edges

Inputs:

f(a)

f(b)

a = b

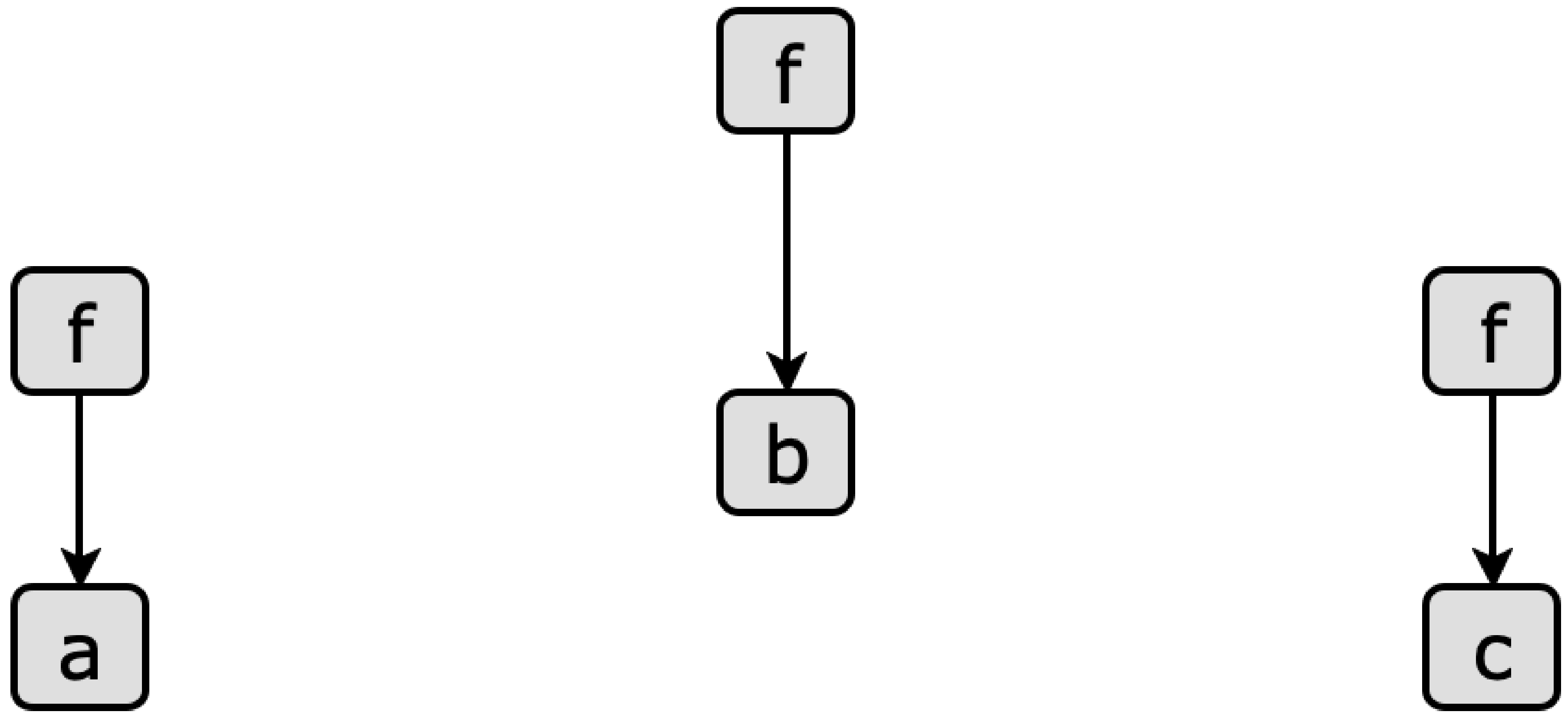


A Bigger E-Graph Example

A Bigger E-Graph Example

Inputs:

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



A Bigger E-Graph Example

Inputs:

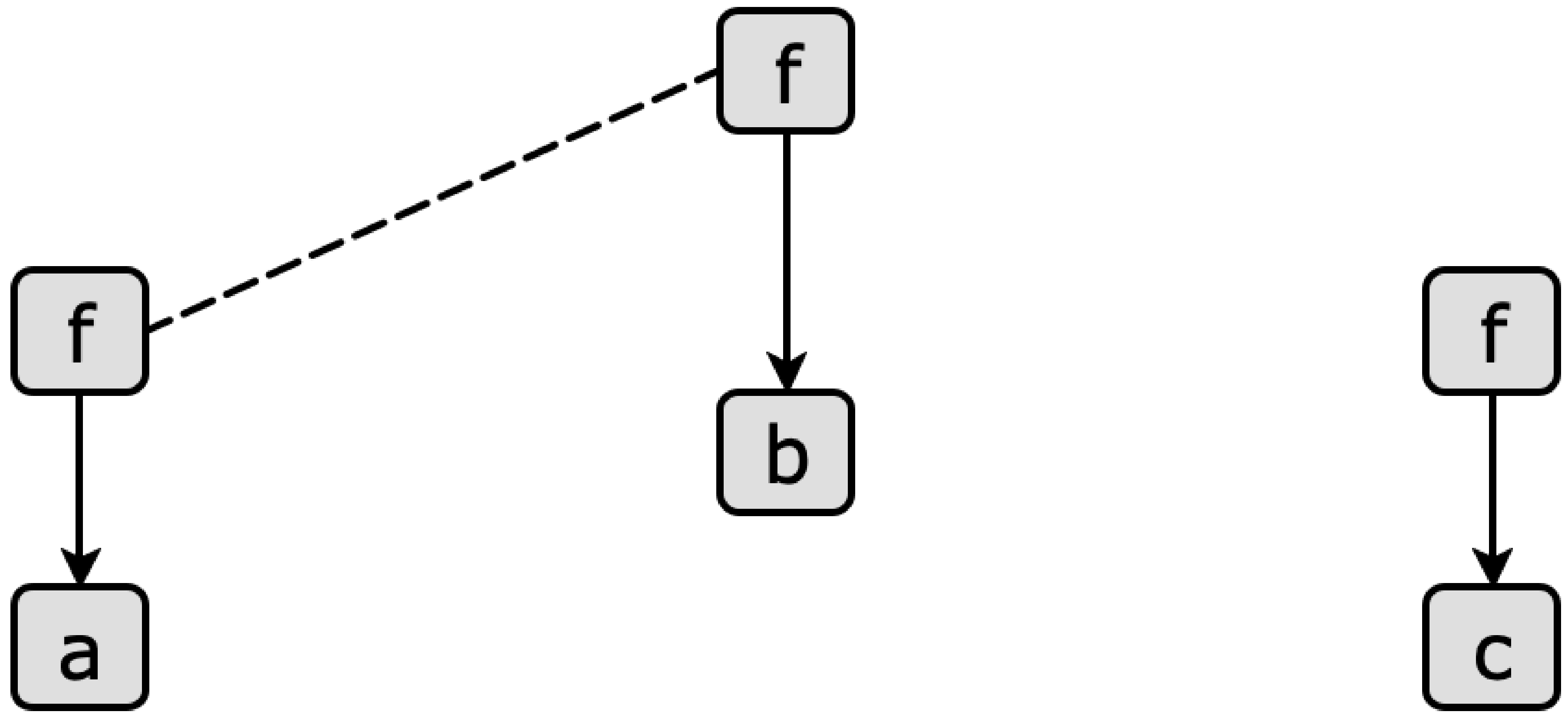
f(c)

f(a) = f(b)

a = b

b = c

a = c



A Bigger E-Graph Example

Inputs:

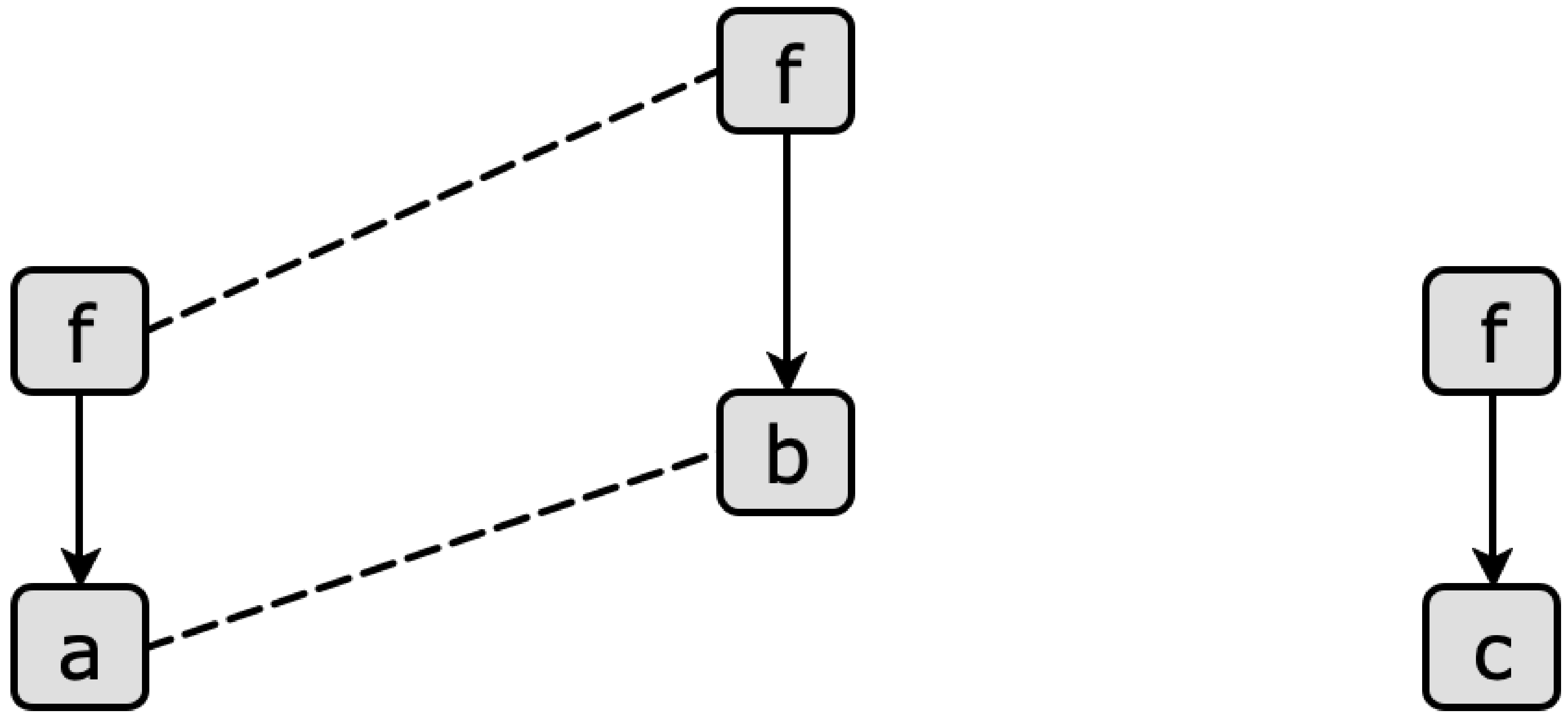
f(c)

f(a) = f(b)

a = b

b = c

a = c



A Bigger E-Graph Example

Inputs:

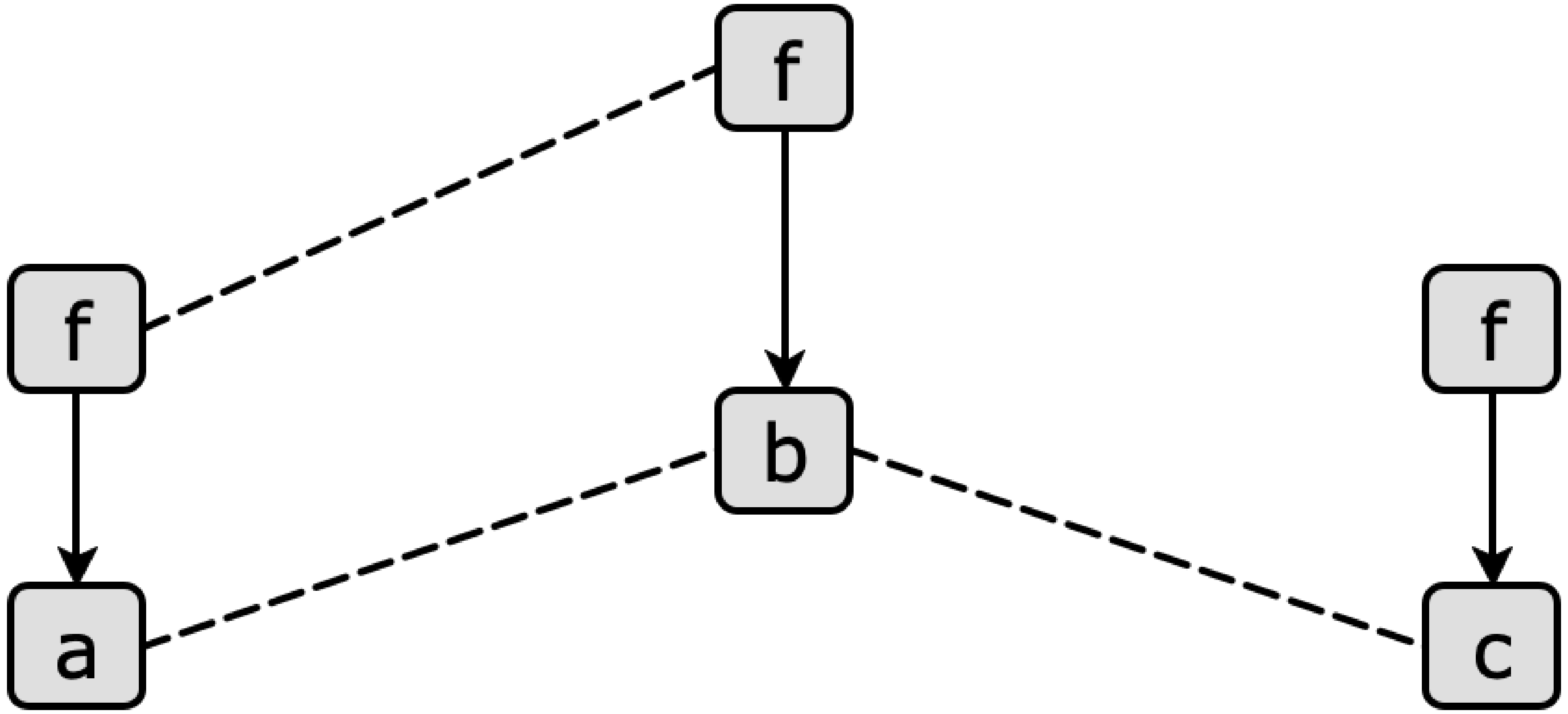
f(c)

f(a) = f(b)

a = b

b = c

a = c



A Bigger E-Graph Example

Inputs:

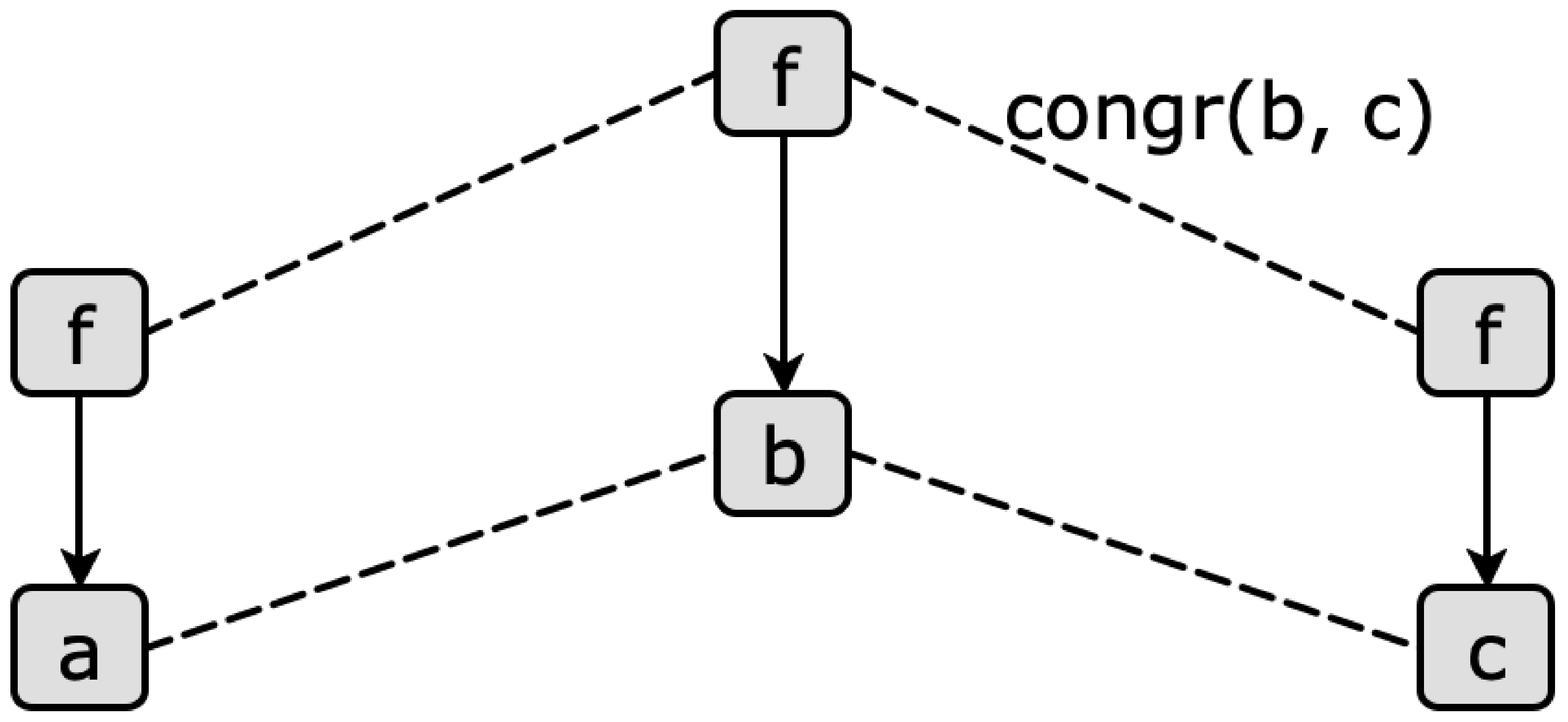
f(c)

f(a) = f(b)

a = b

b = c

a = c



A Bigger E-Graph Example

Inputs:

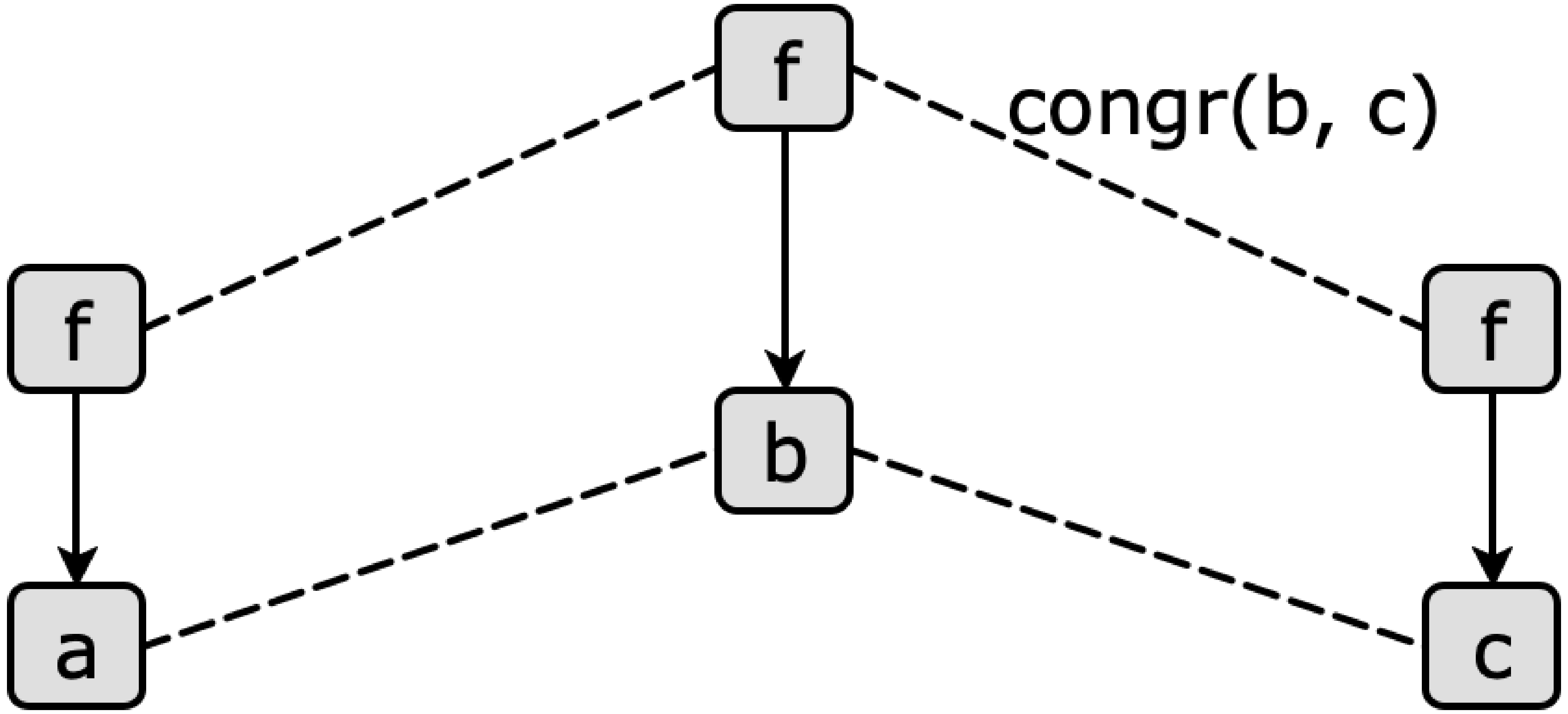
f(c)

f(a) = f(b)

a = b

b = c

a = c



A Bigger E-Graph Example

Inputs:

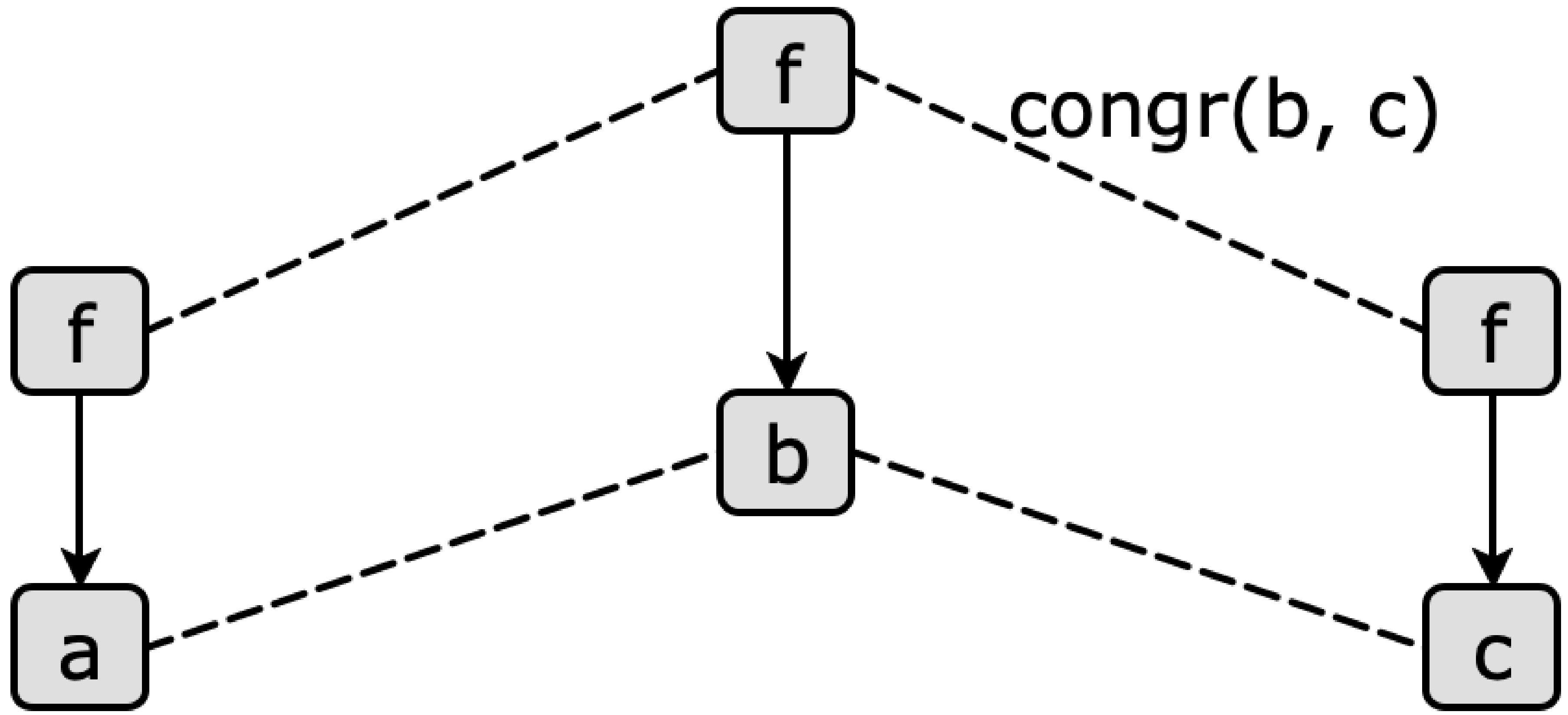
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



A Bigger E-Graph Example

Inputs:

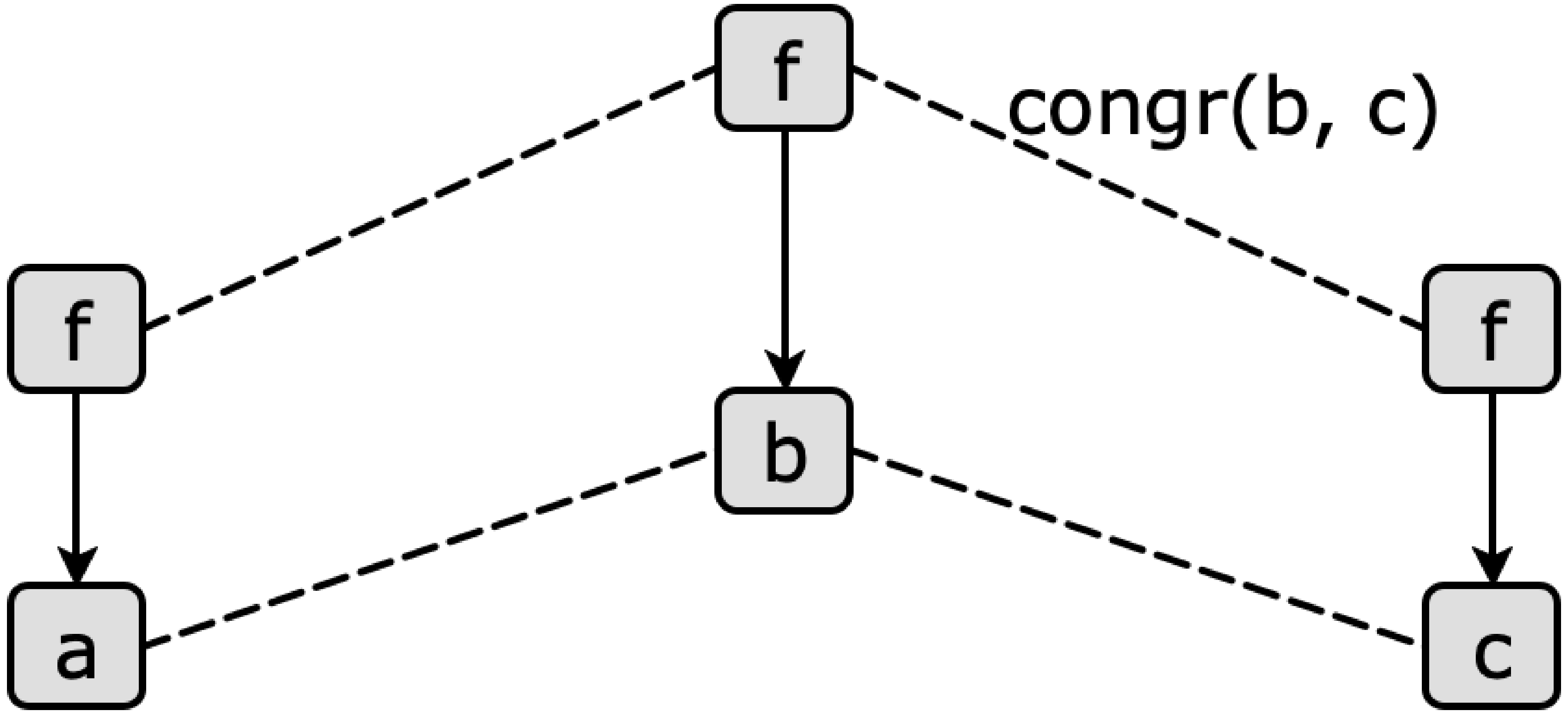
f(c)

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:

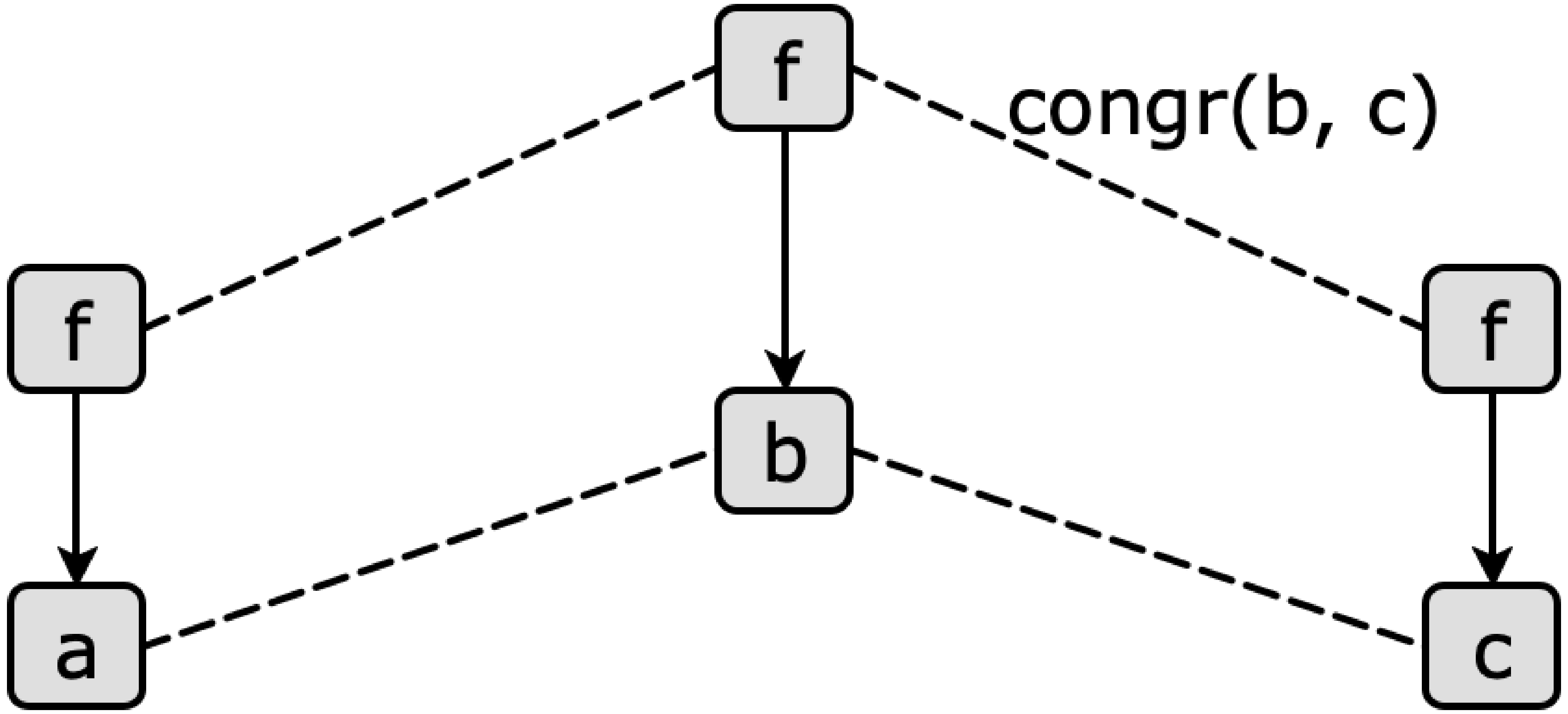
f(c)

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:

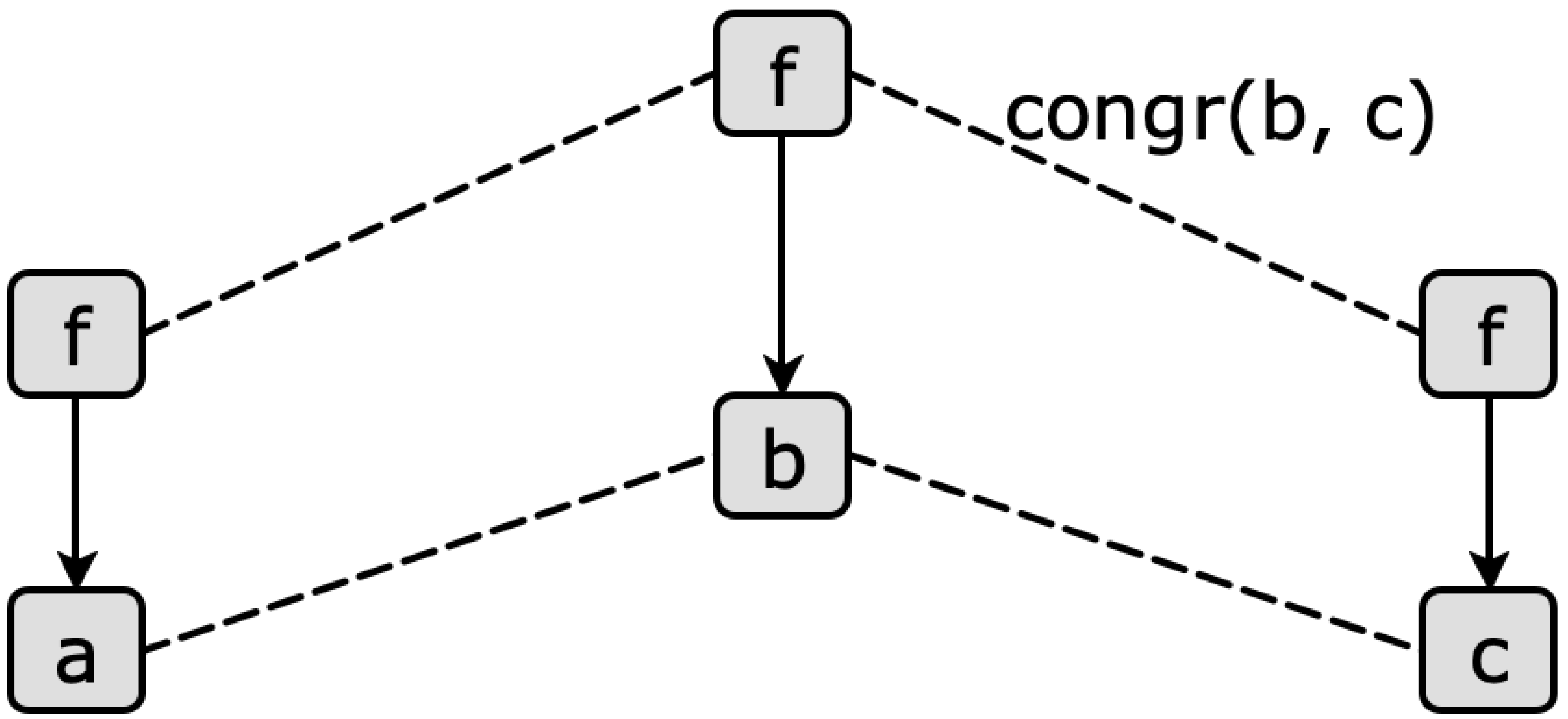
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



Example:

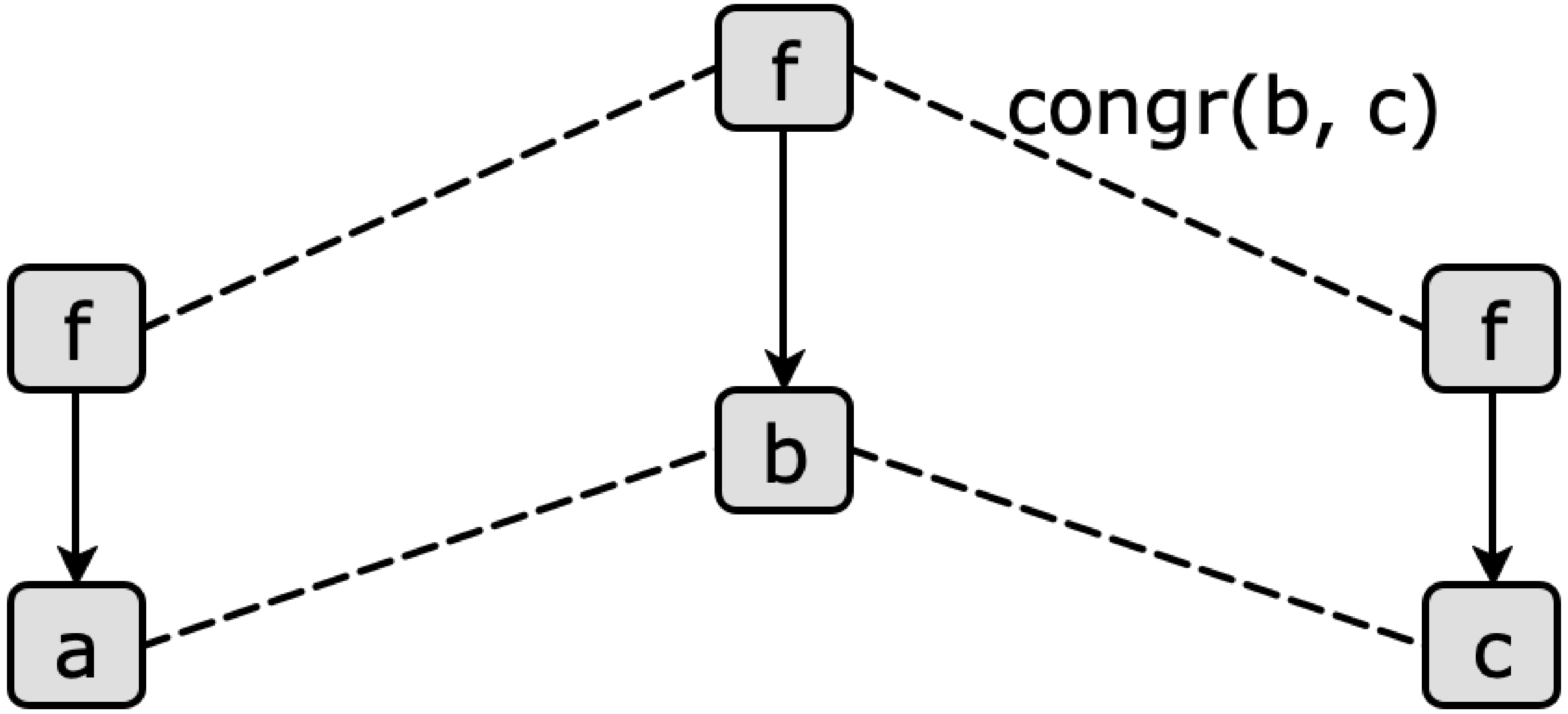
f(c)

f(a) = f(b)

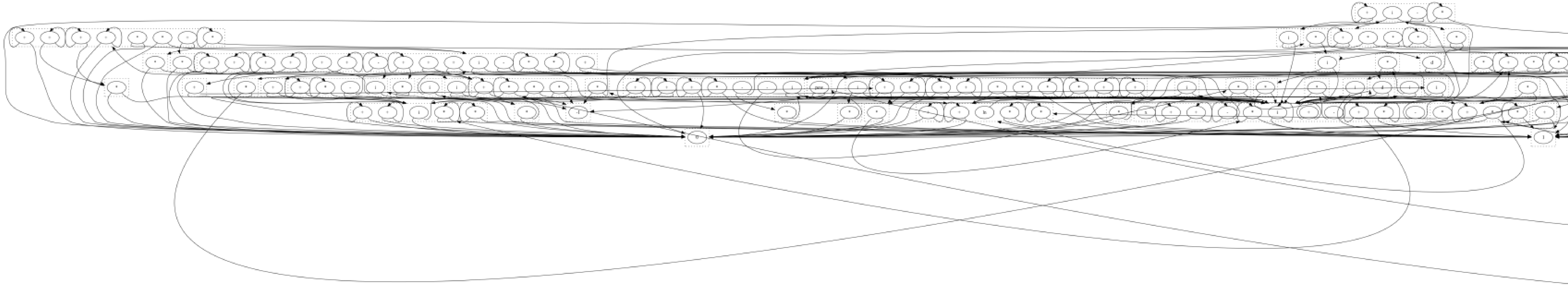
a = b

b = c

a = c unnecessary



Reality:



Example:

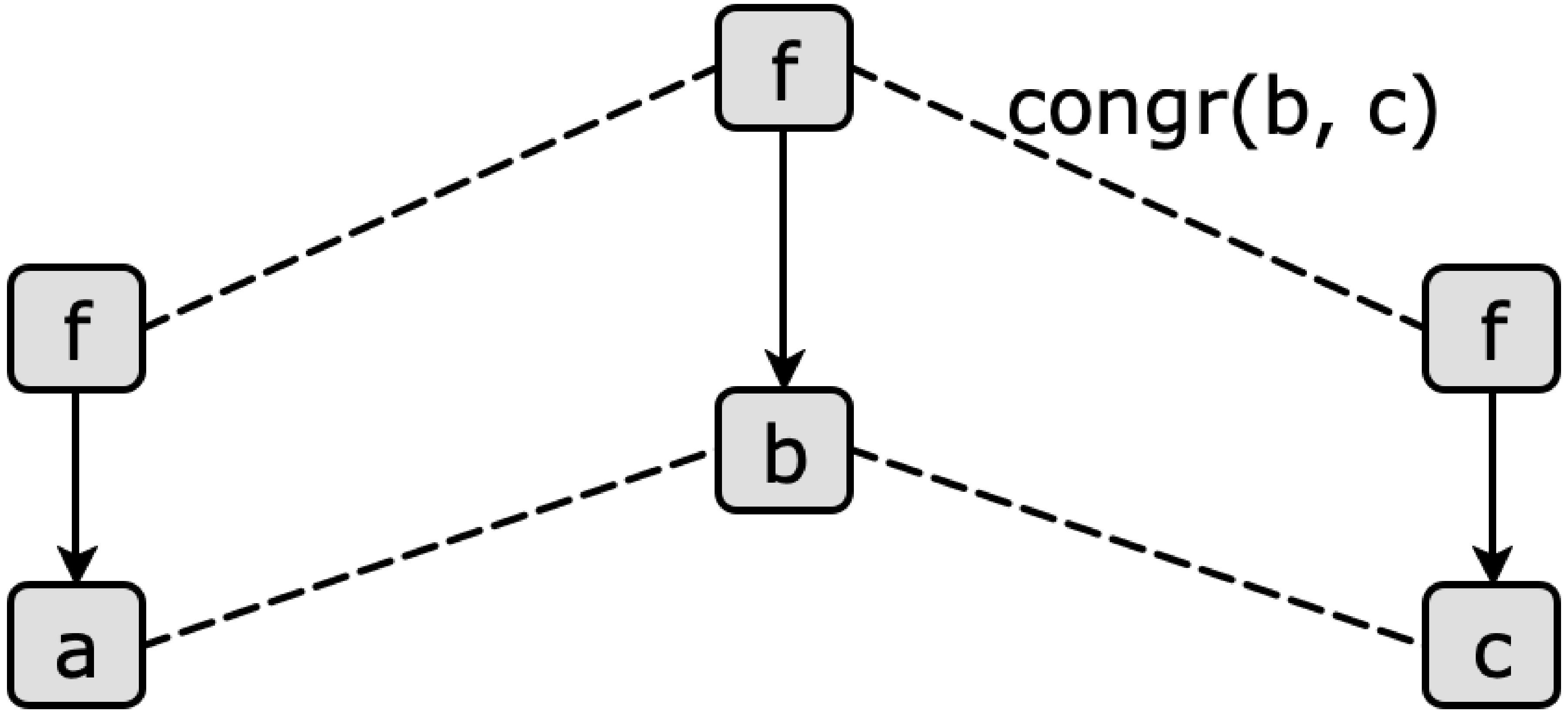
f(c)

f(a) = f(b)

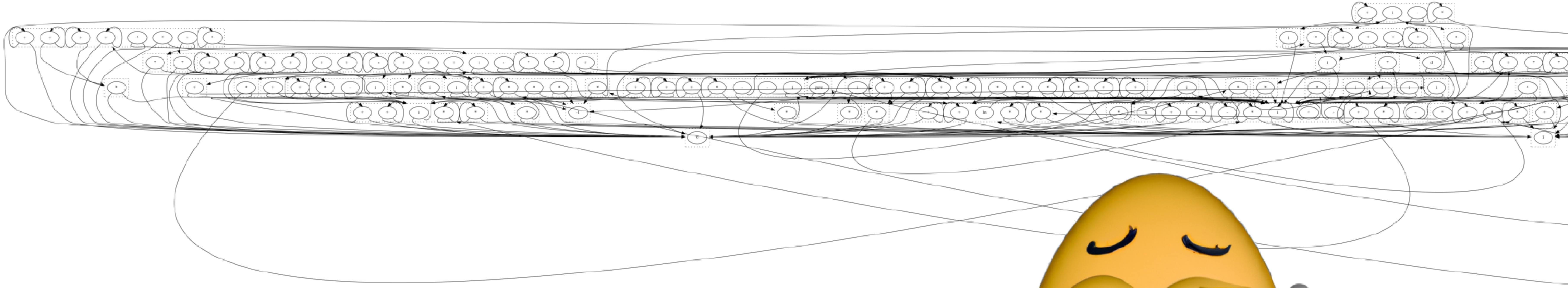
a = b

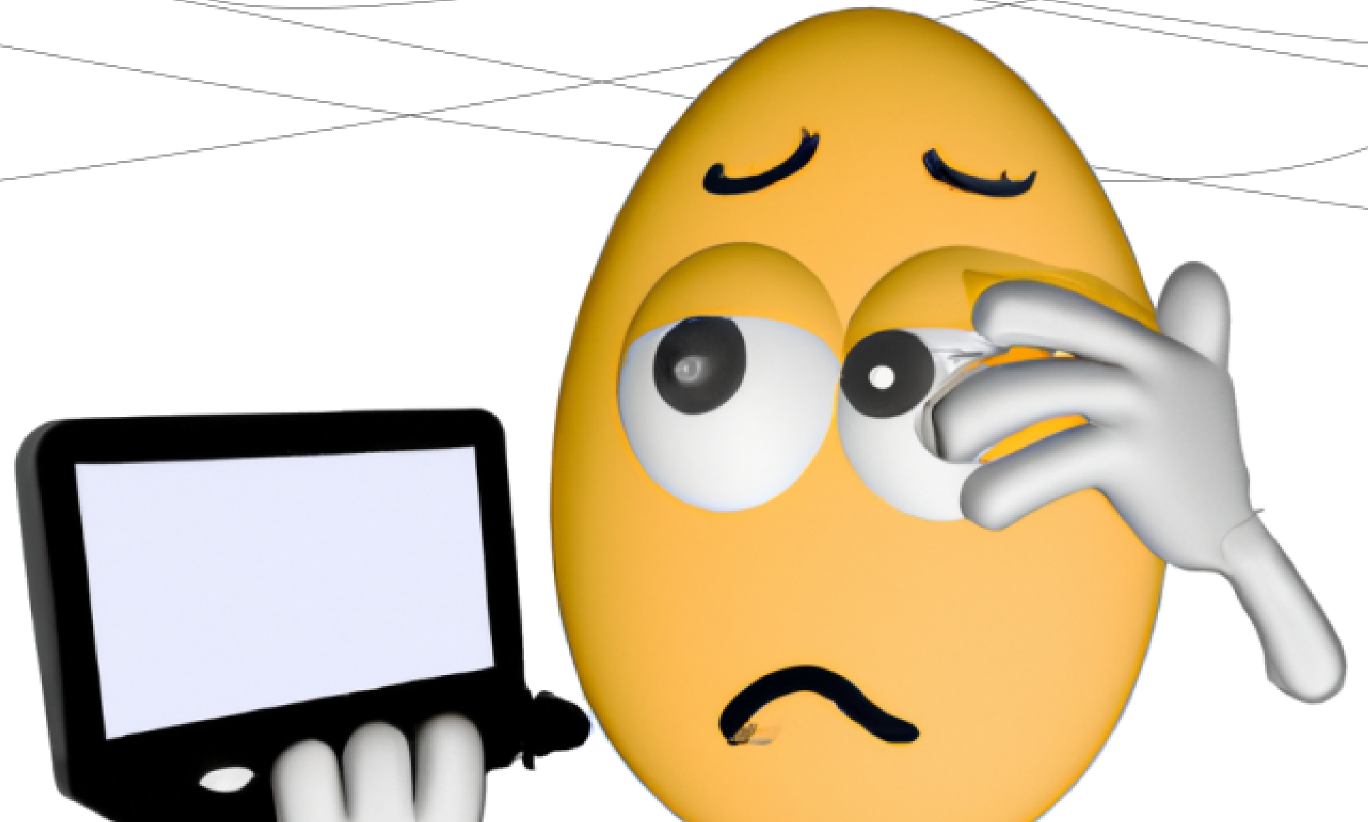
b = c

a = c unnecessary



Reality:





- [x] Motivation
- [x] 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:

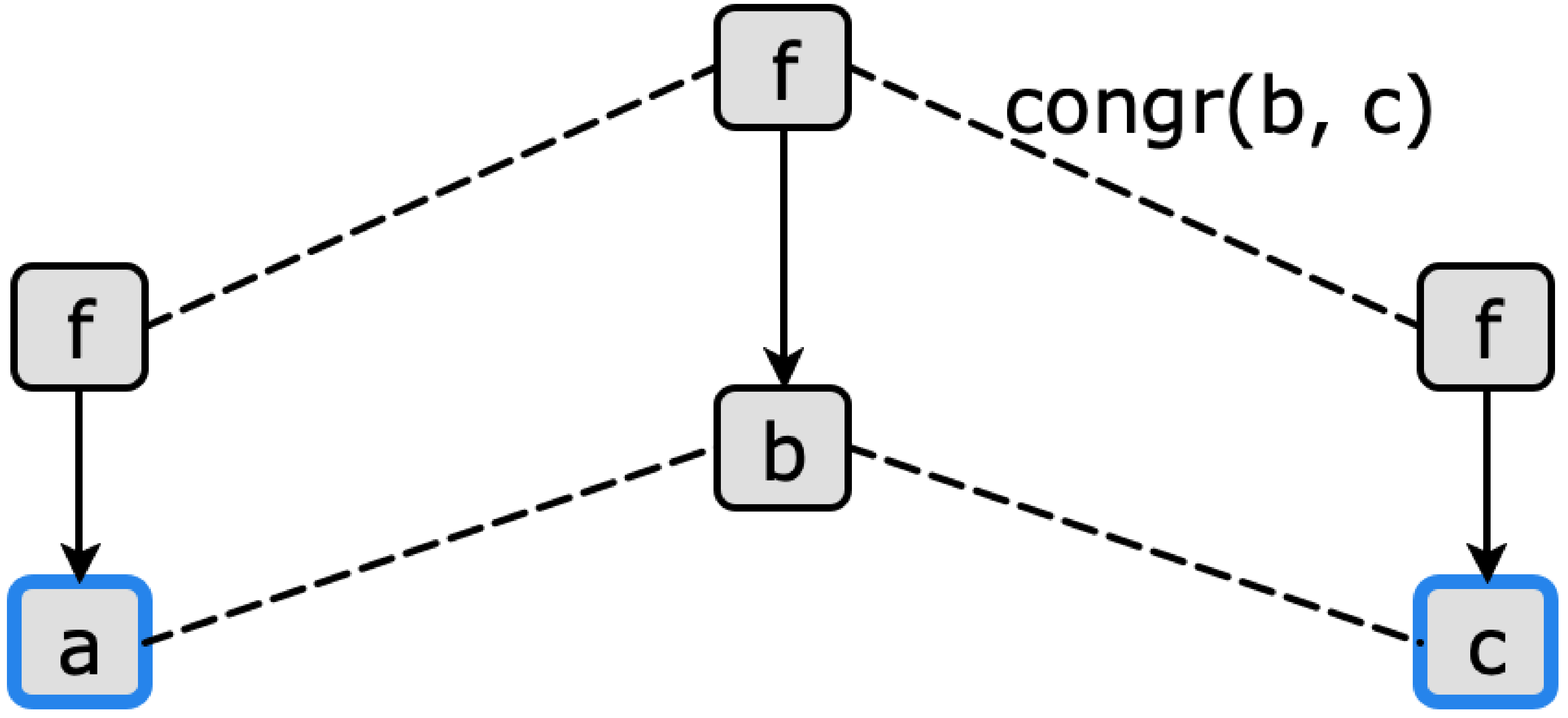
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



Congruence Proofs

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

Inputs:

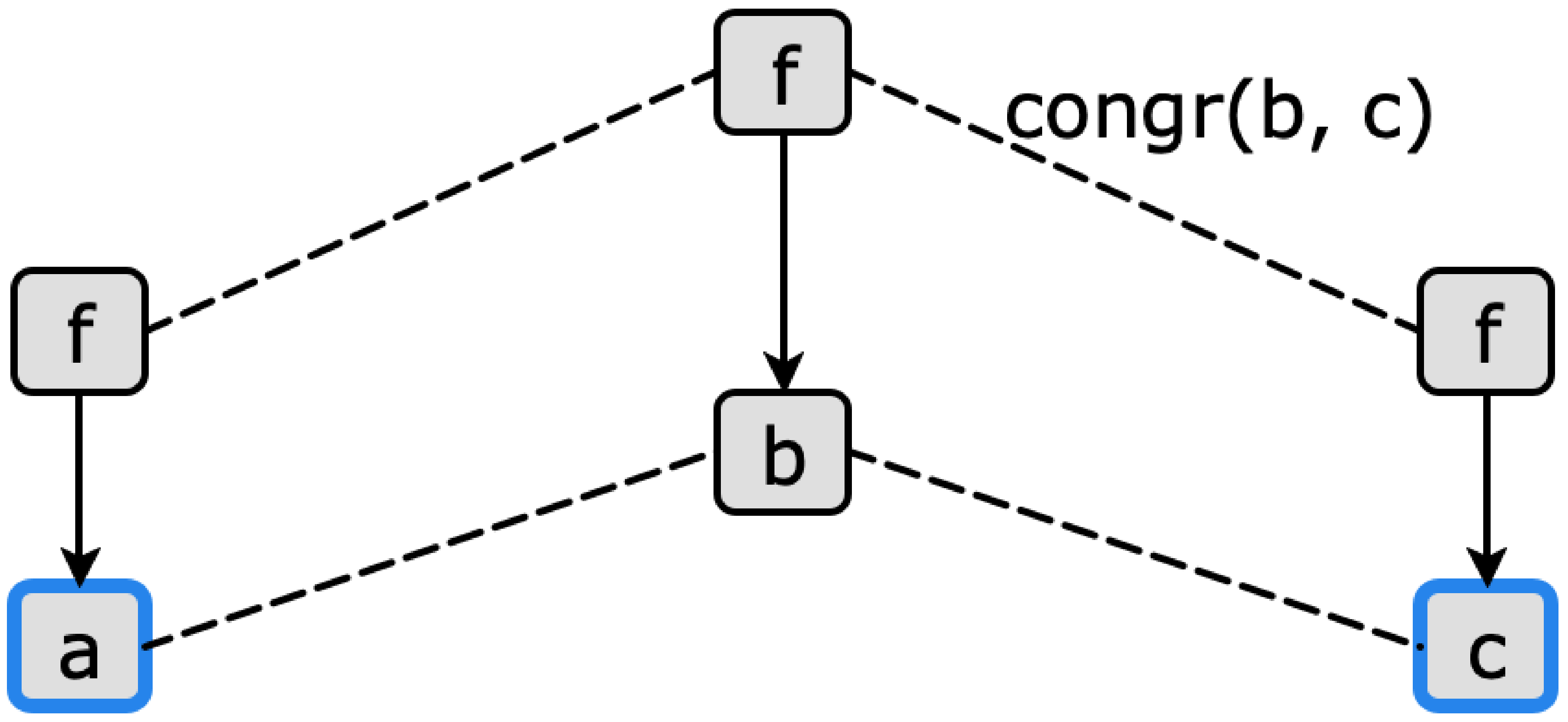
f(c)

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:

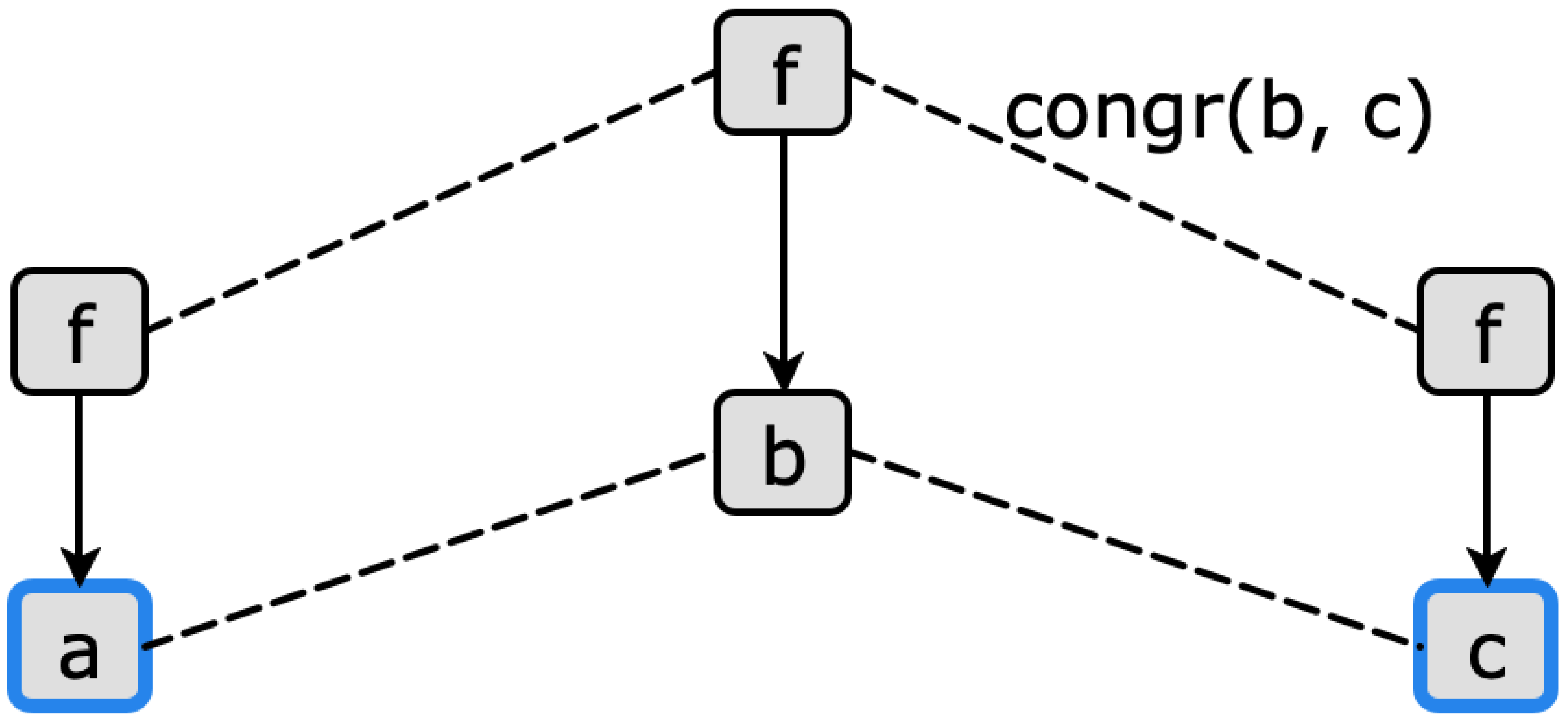
f(c)

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:

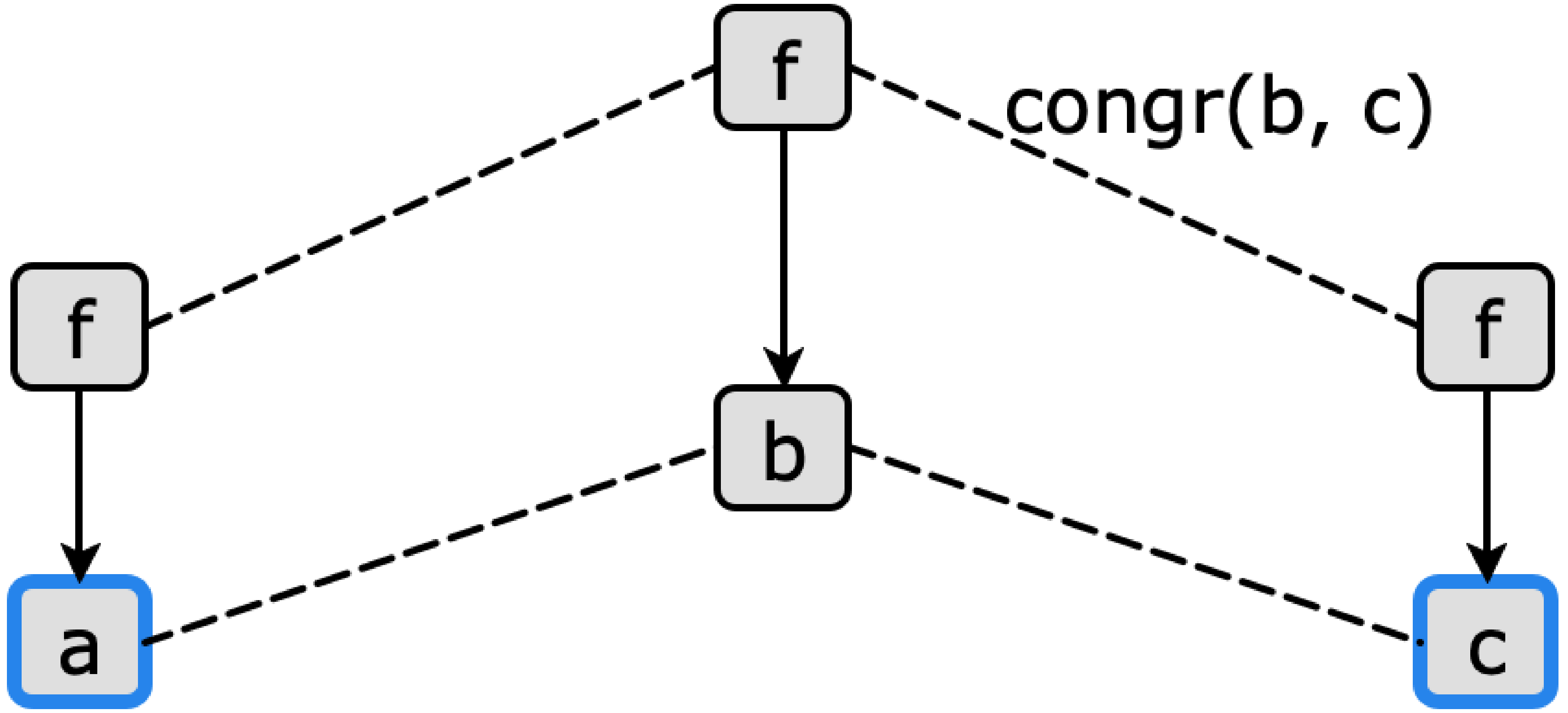
f(c)

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:

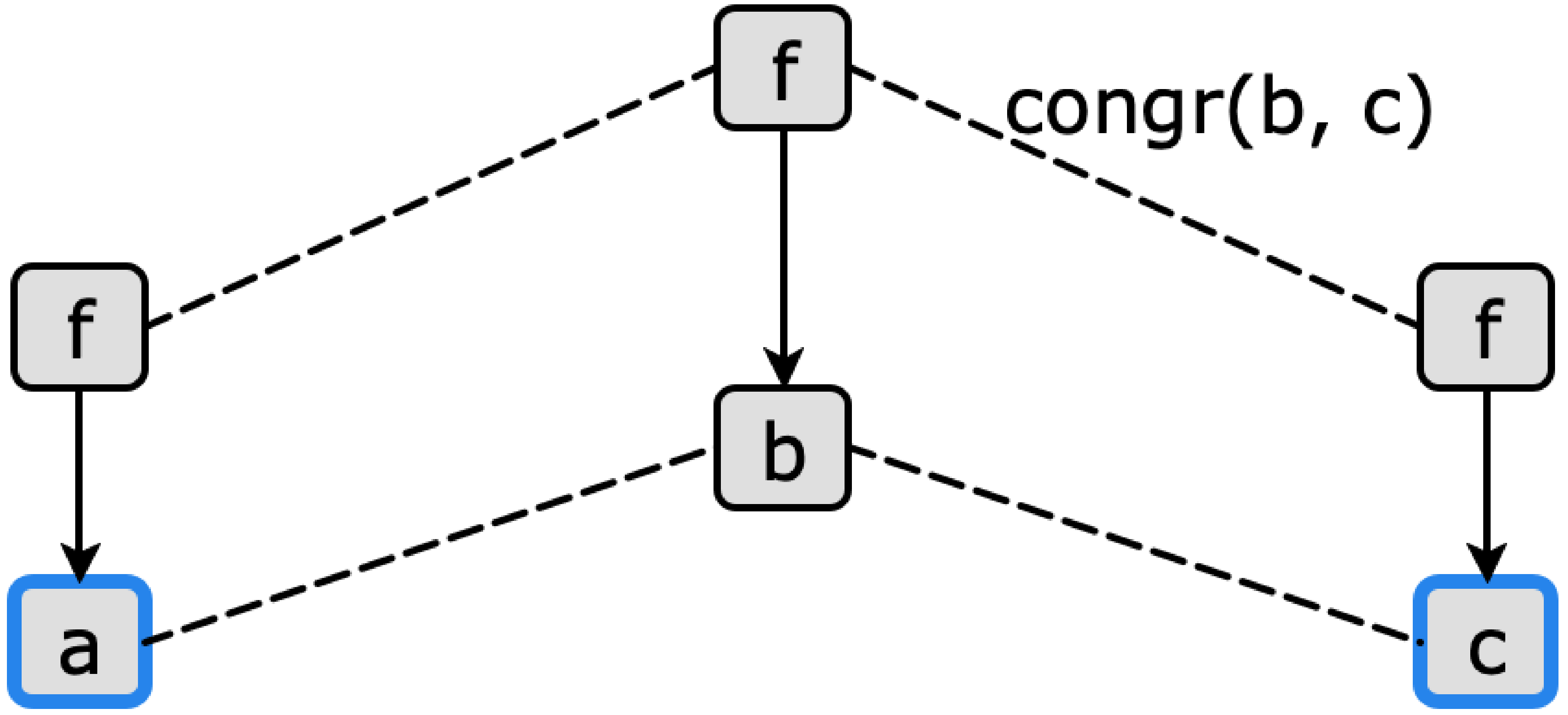
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



Prove a and c are equal:

a = b

b = c

done!

Inputs:

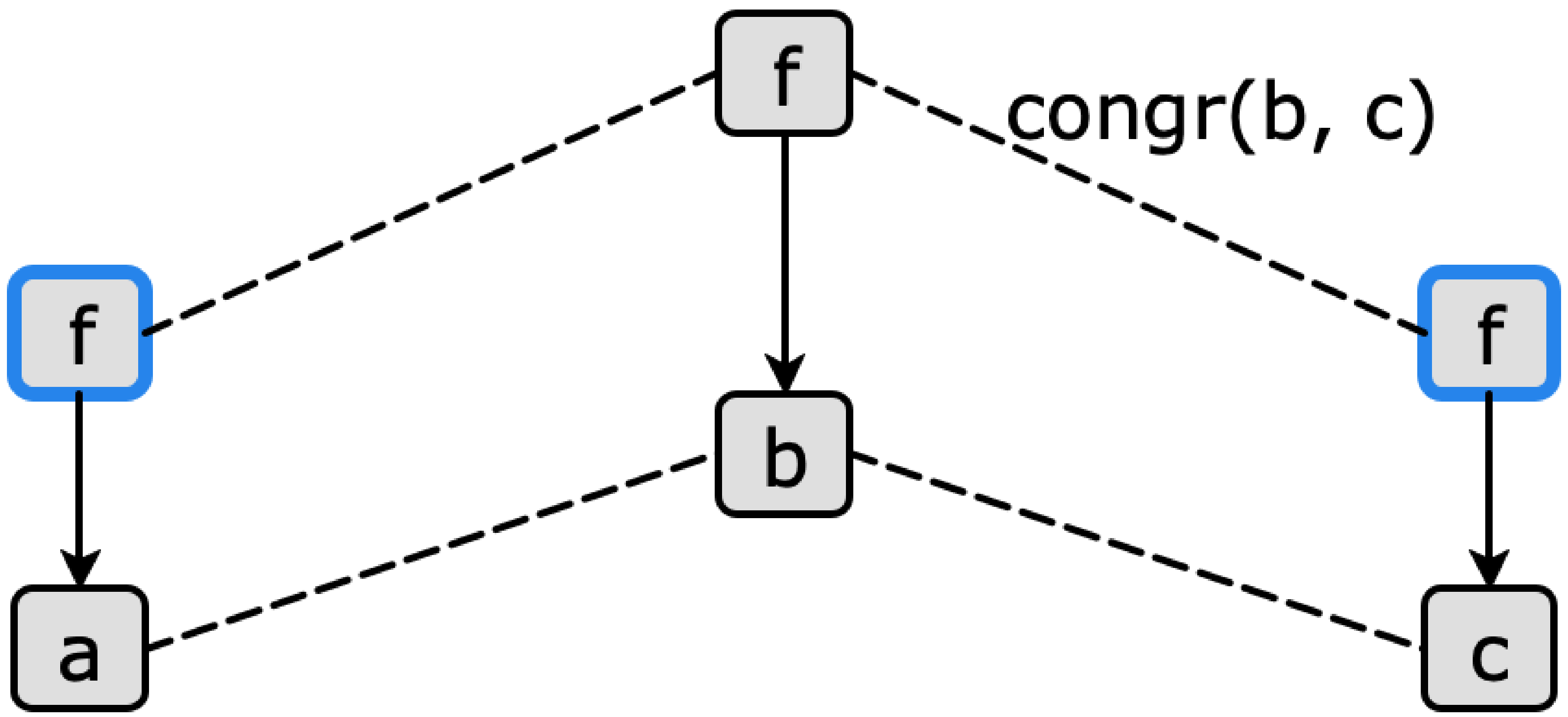
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



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

Inputs:

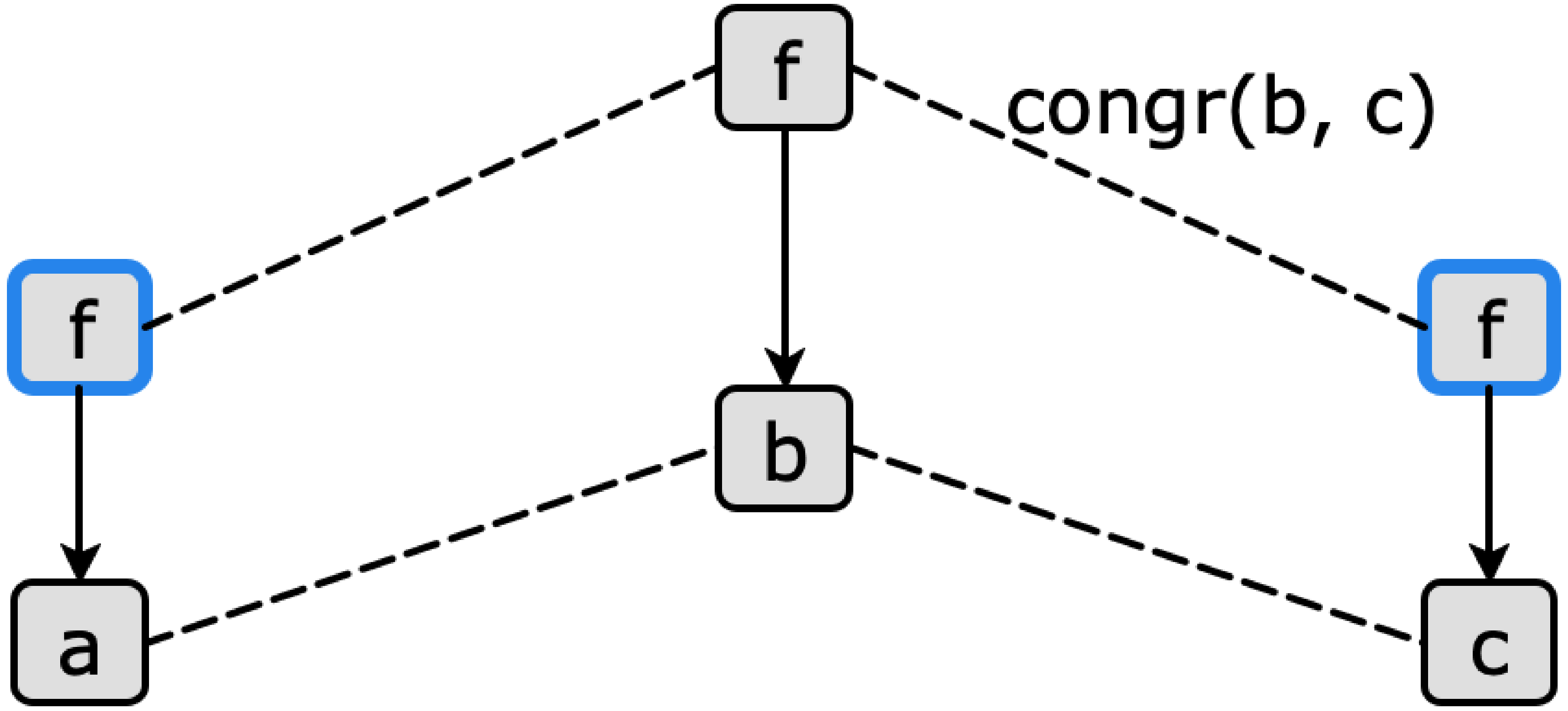
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



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

f(a) = f(b)

Inputs:

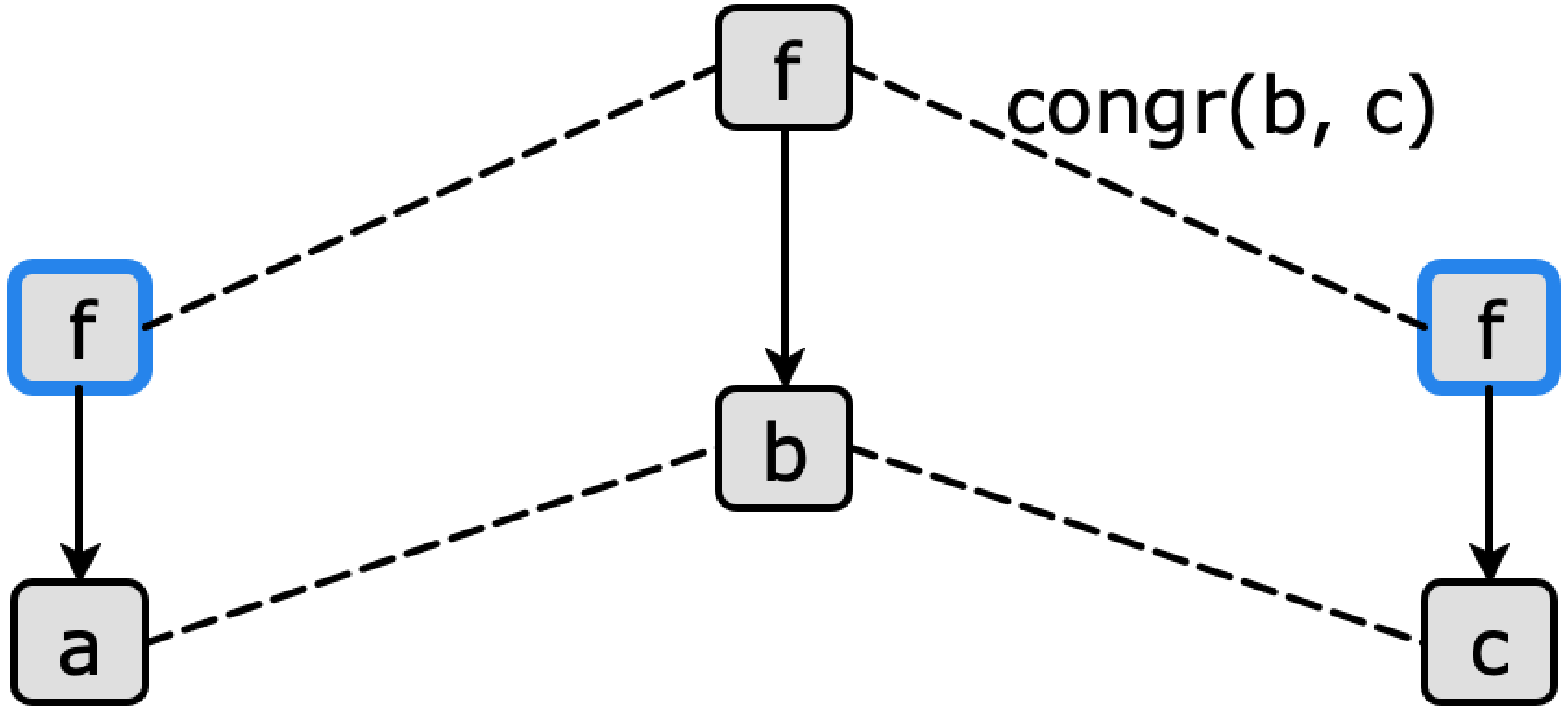
f(c)

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:

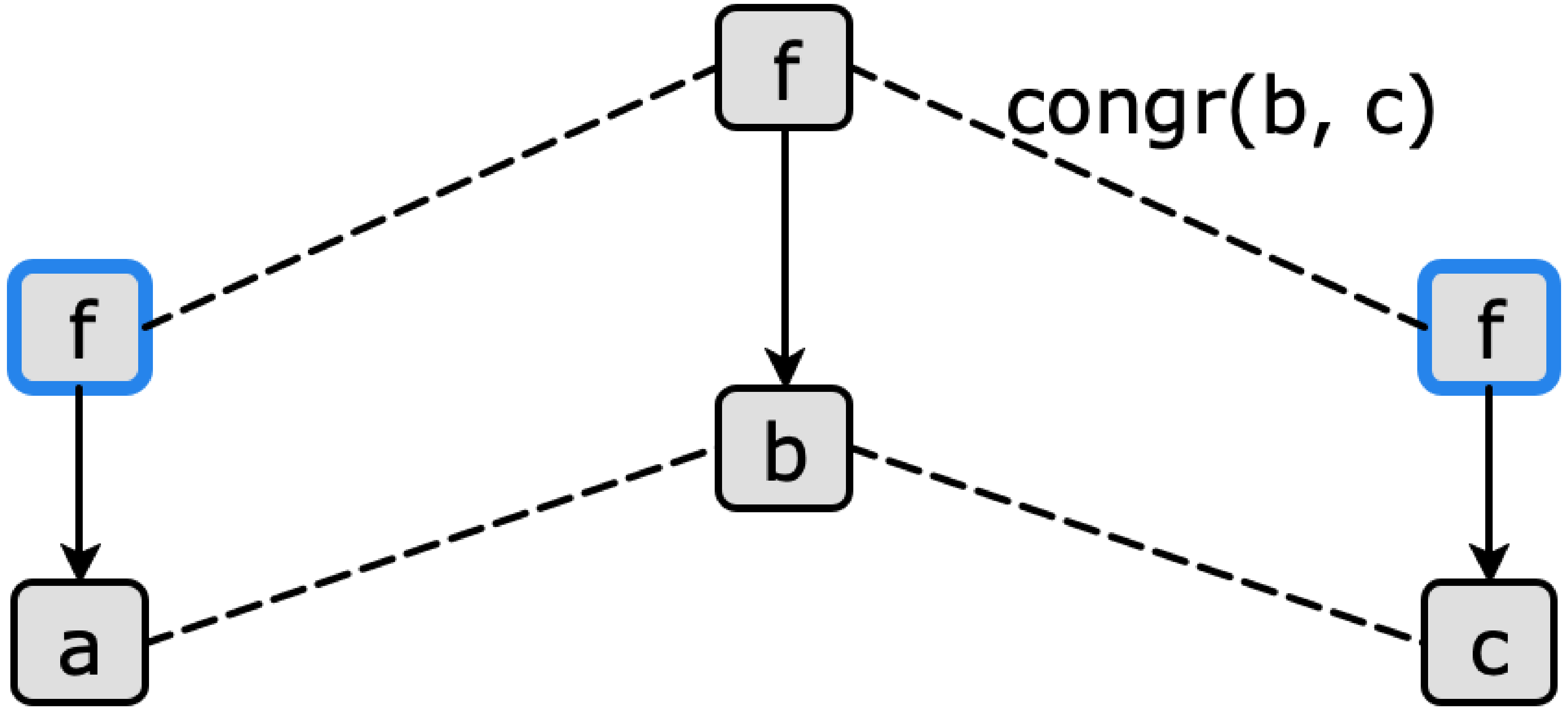
f(c)

f(a) = f(b)

a = b

b = c

a = c unnecessary



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

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

Inputs:

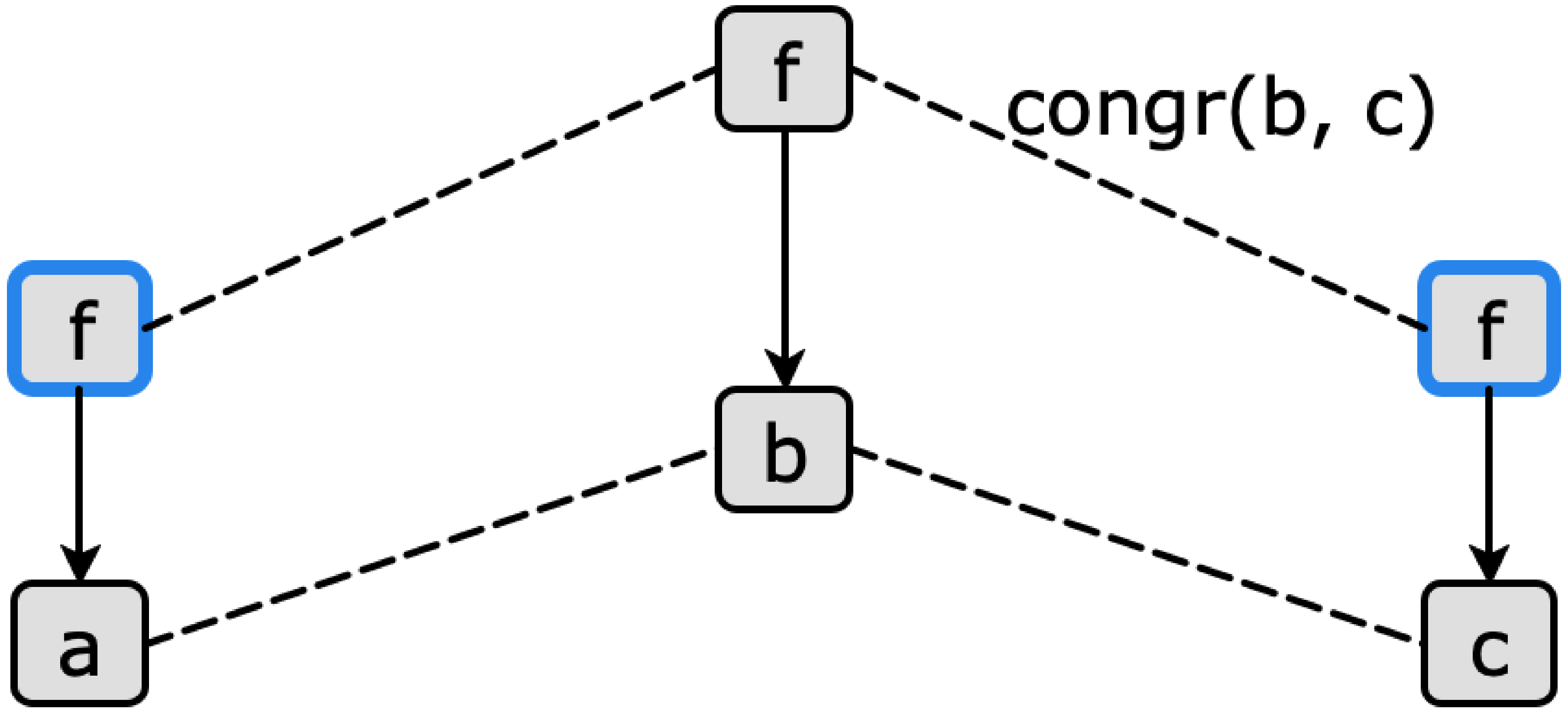
f(c)

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:

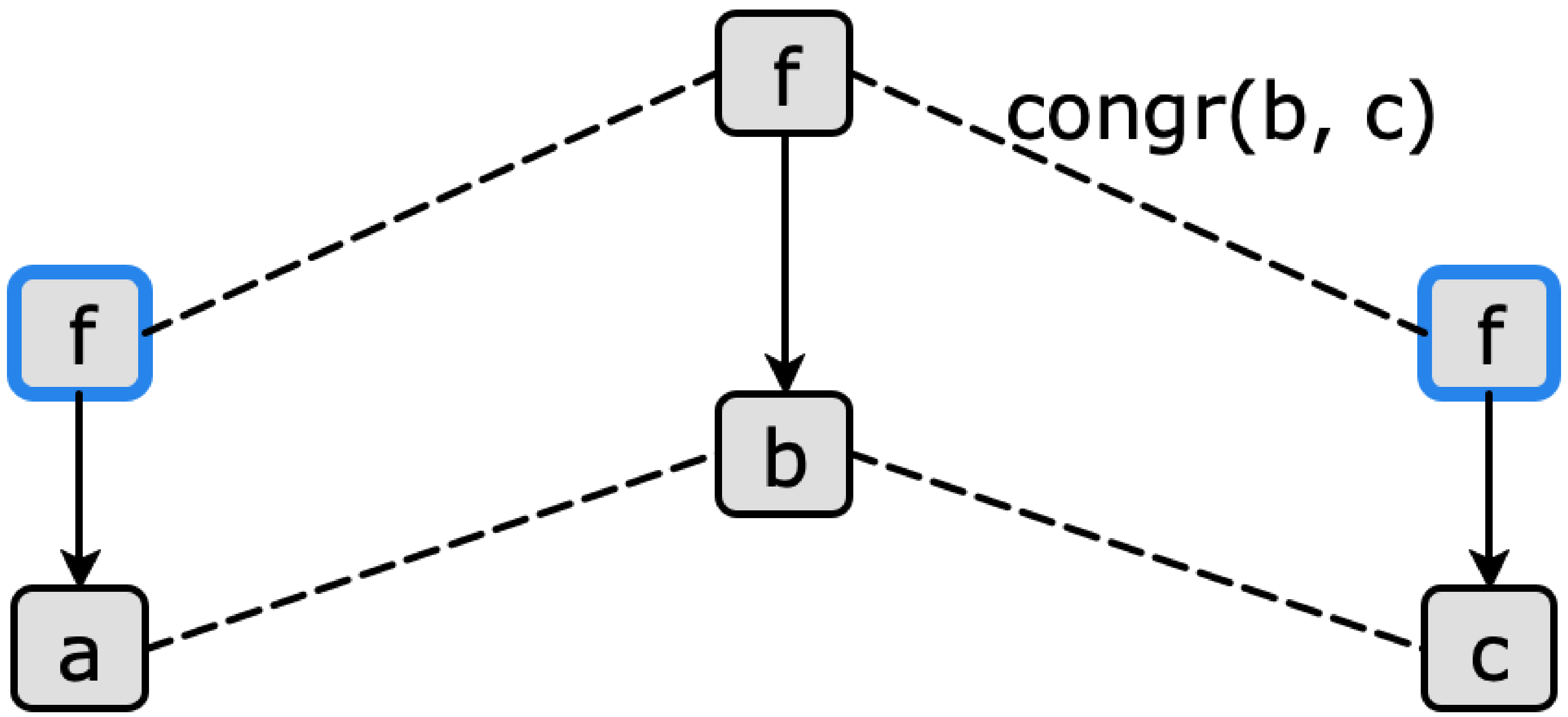
f(c)

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

We can do better!

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

f(a) = f(b)

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

b = c

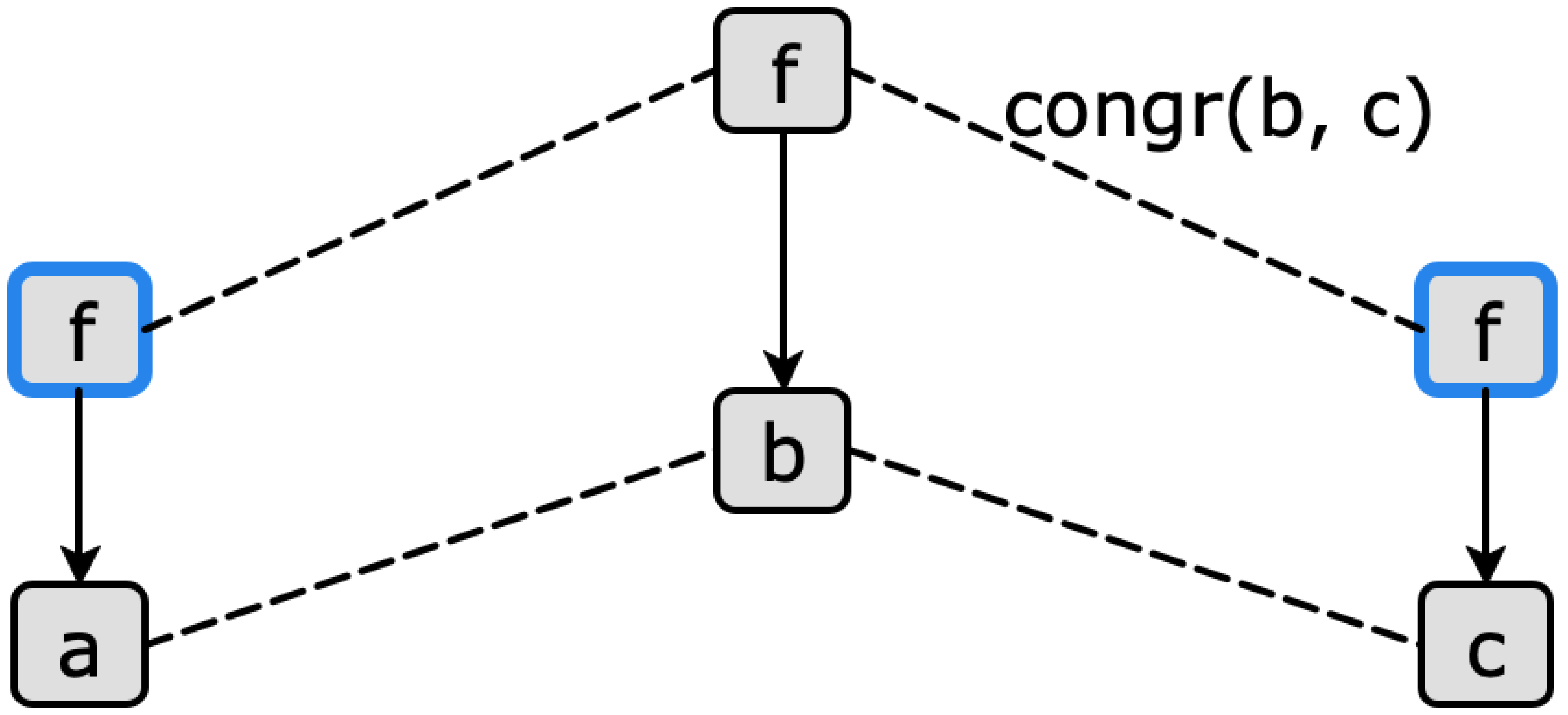
done!

Leveraging Additional Equalities

Inputs:

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

unnecessary



Leveraging Additional Equalities

Inputs:

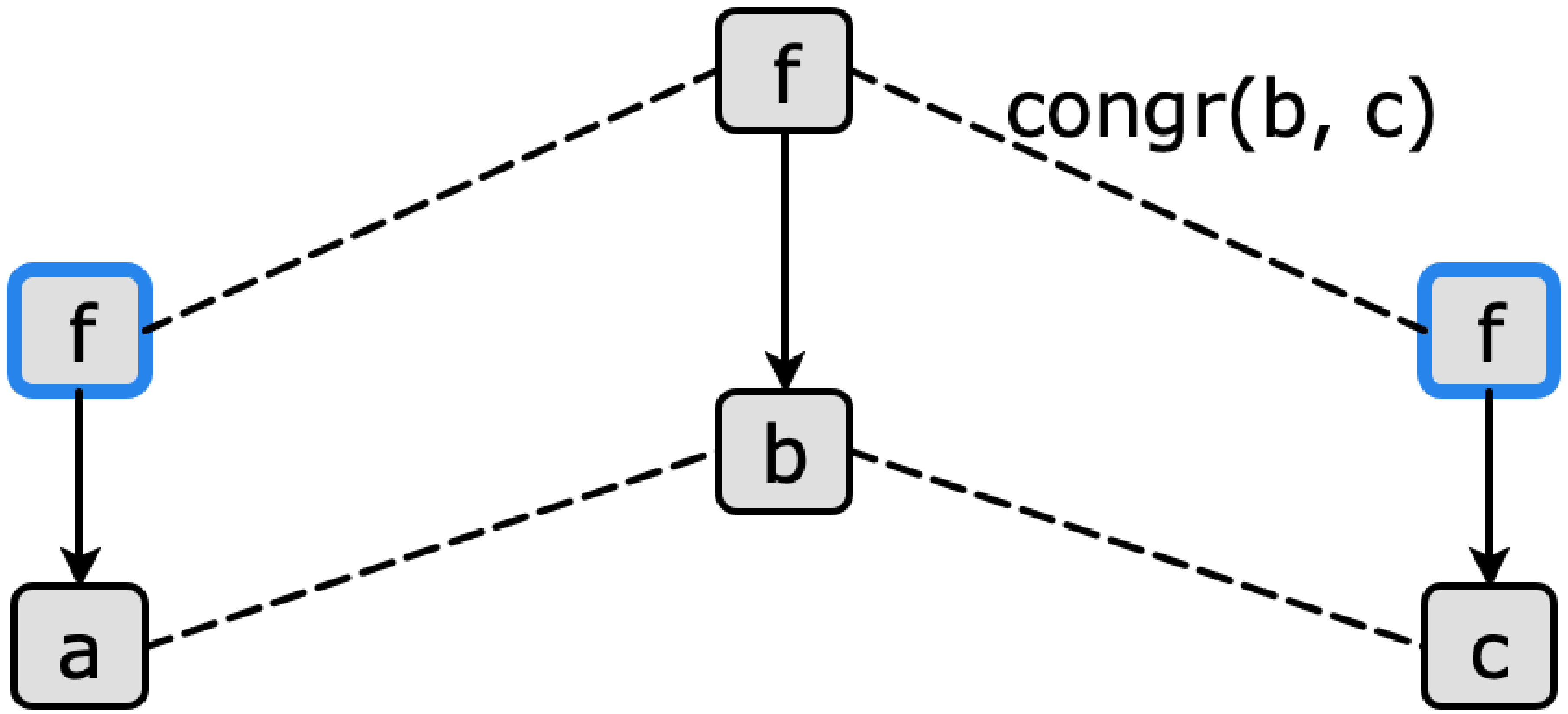
f(c)

f(a) = f(b)

a = b

b = c

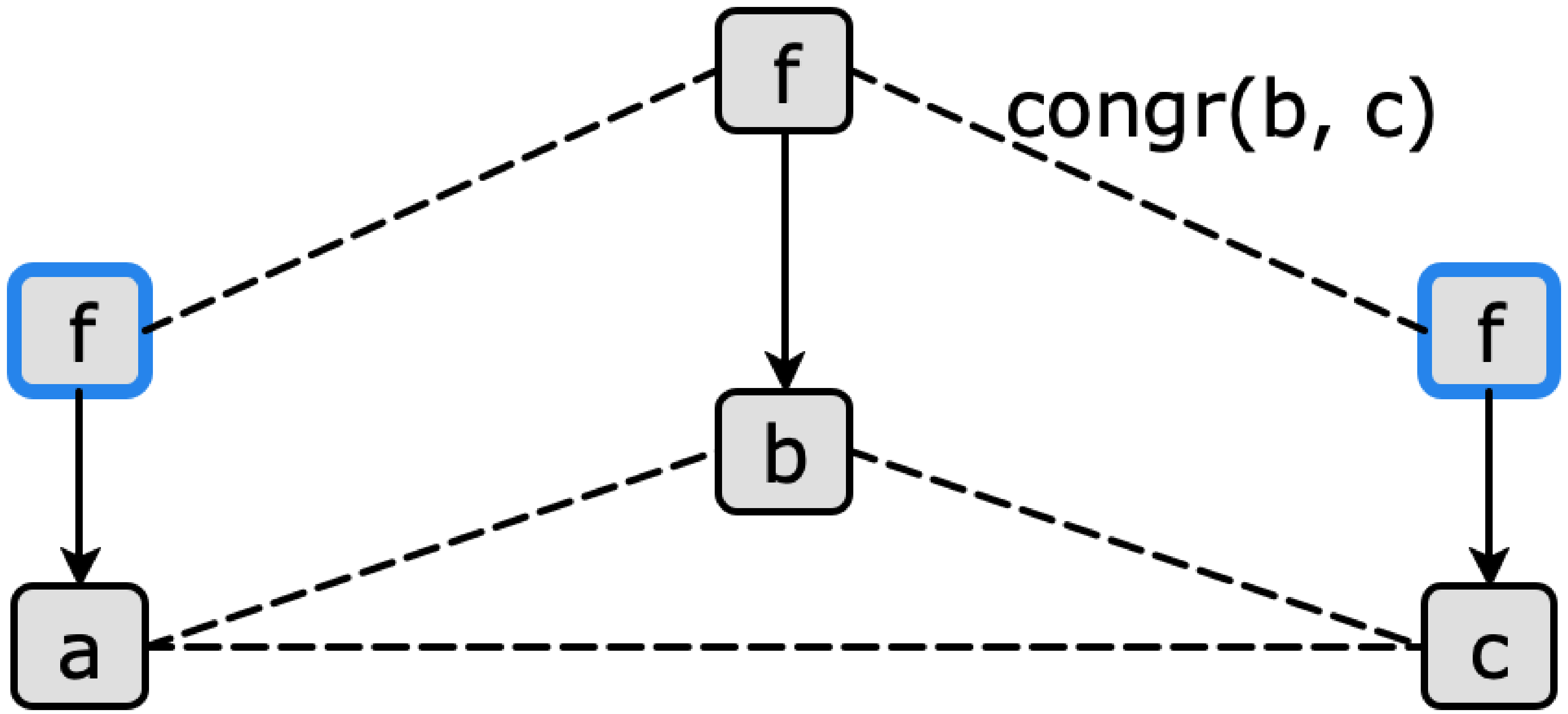
a = c useful



Leveraging Additional Equalities

Inputs:

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



Leveraging Additional Equalities

Inputs:

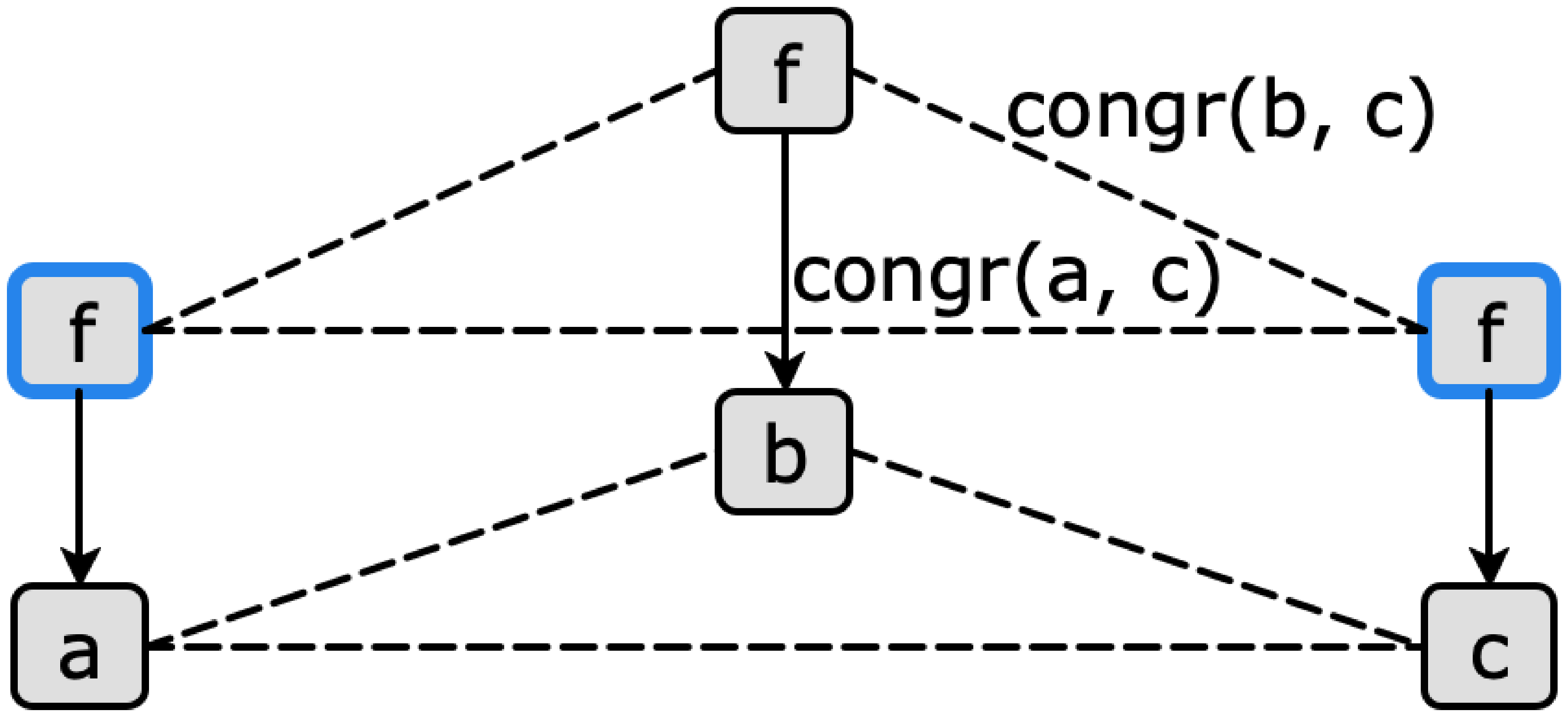
f(c)

f(a) = f(b)

a = b

b = c

a = c useful



Leveraging Additional Equalities

Inputs:

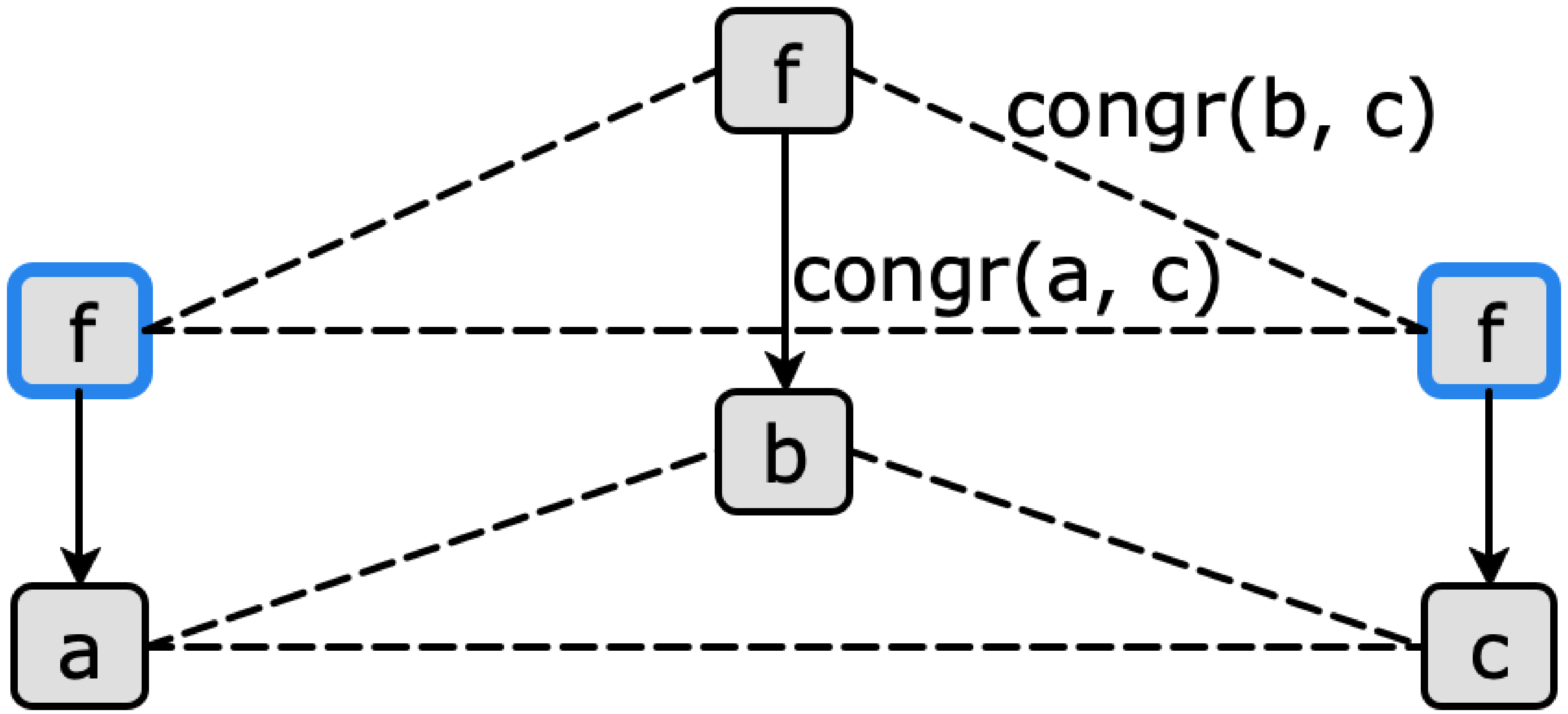
f(c)

f(a) = f(b)

a = b

b = c

a = c useful

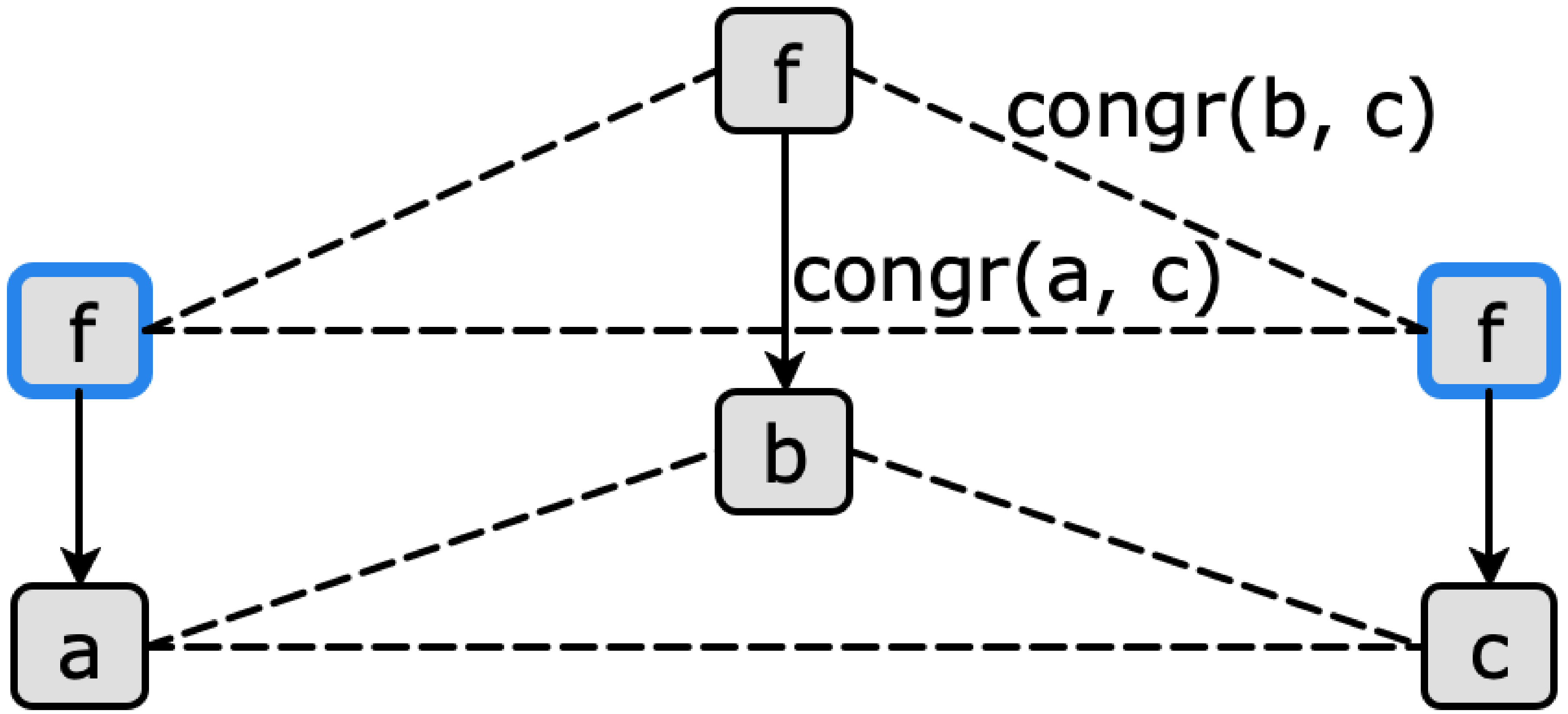


Prove f(a) and f(c) are equal:

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:

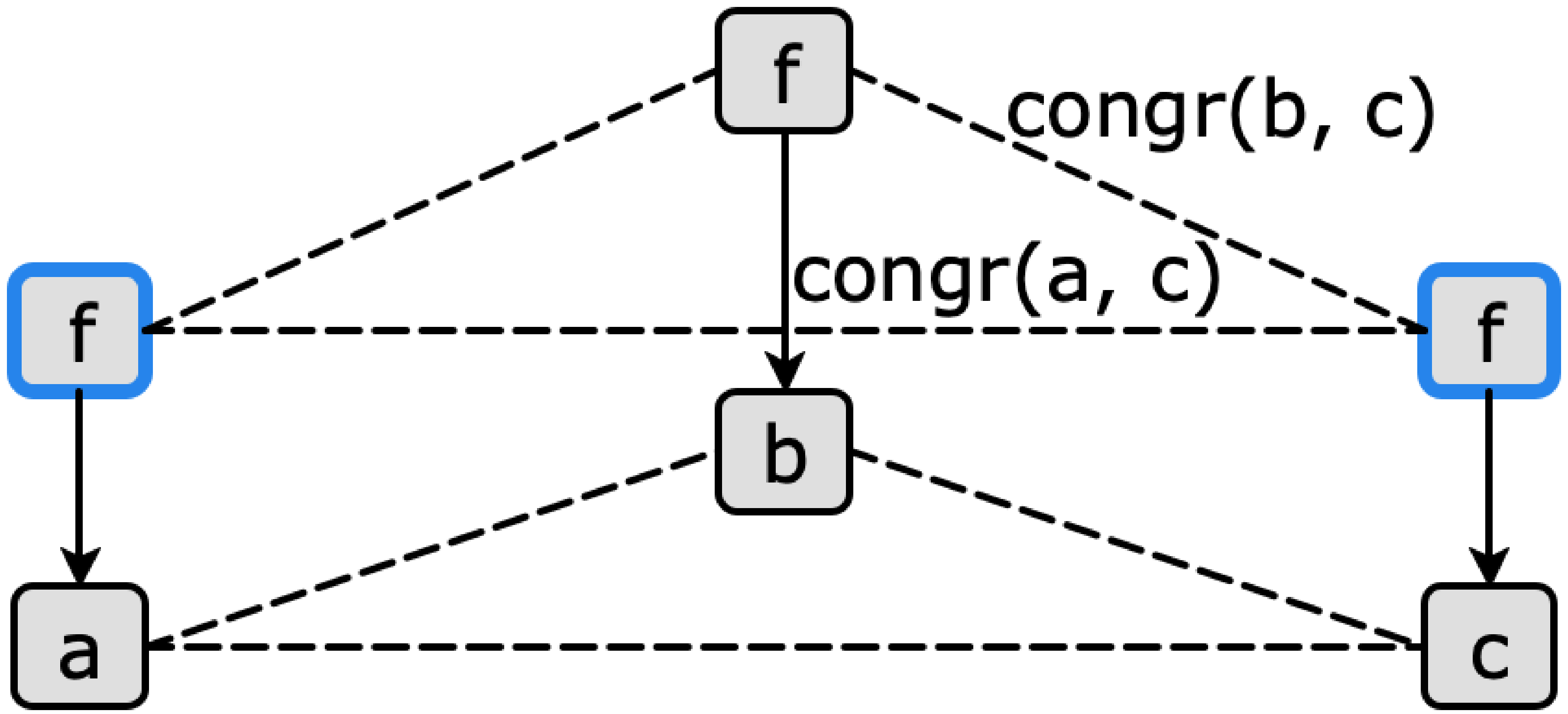
f(c)

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:

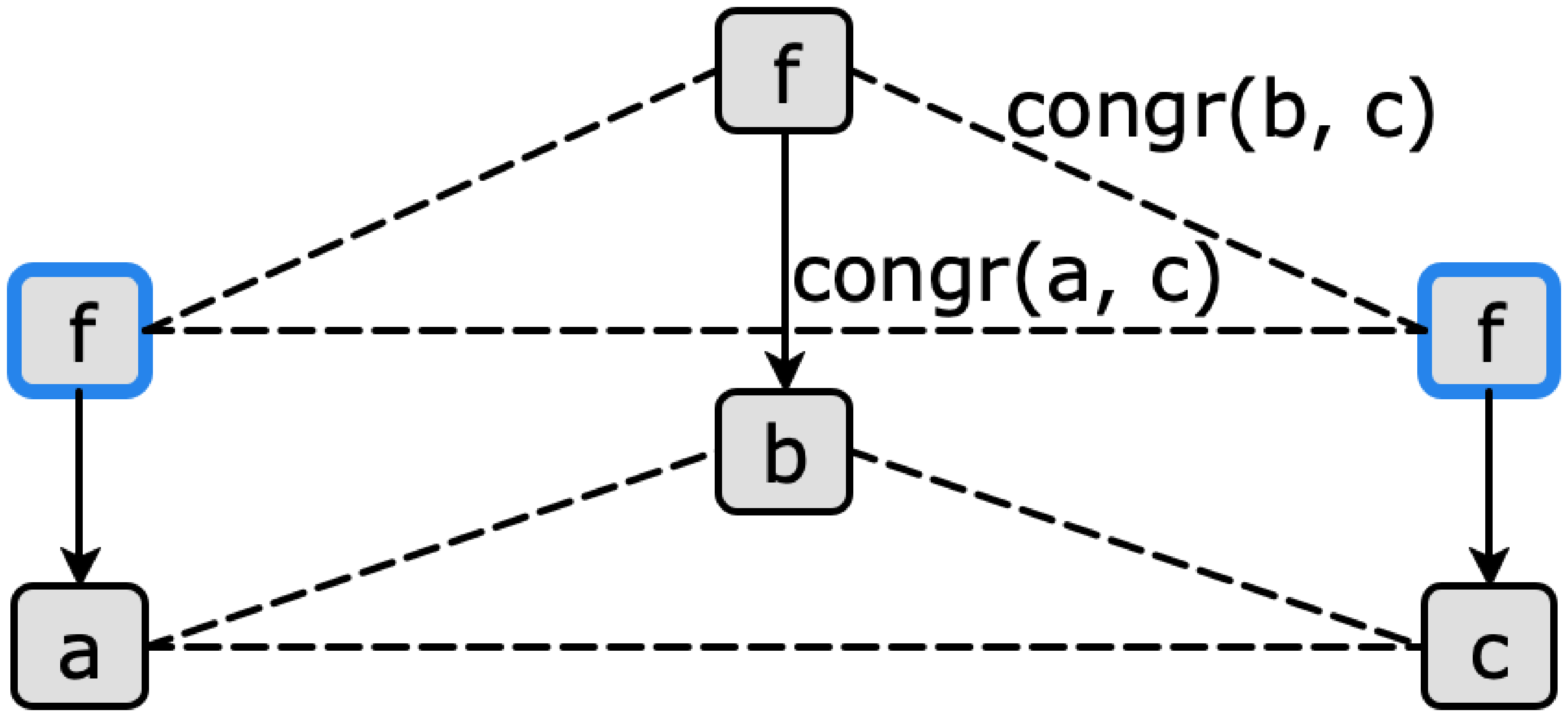
f(c)

f(a) = f(b)

a = b

b = c

a = c useful



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

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

a = c

done!

Leveraging Additional Equalities

Inputs:

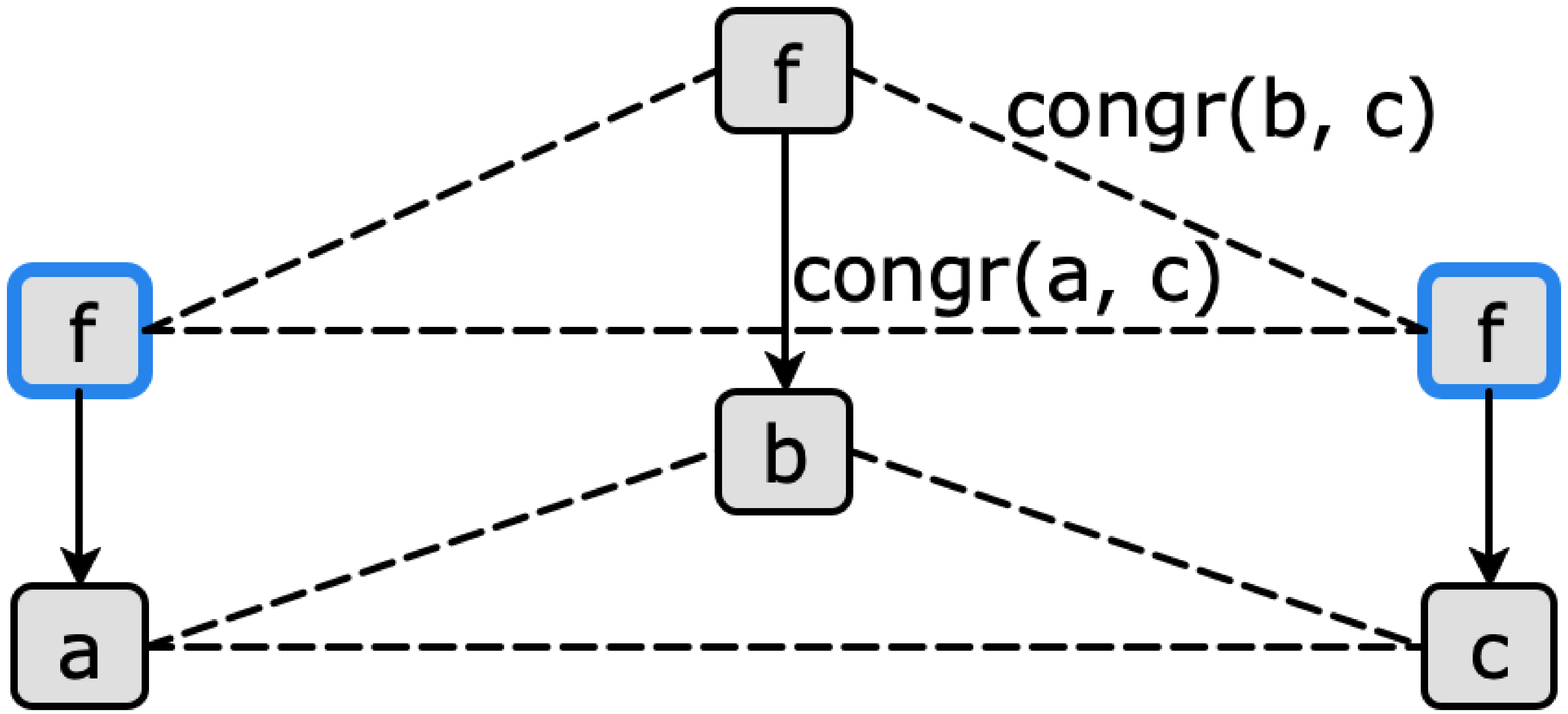
f(c)

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

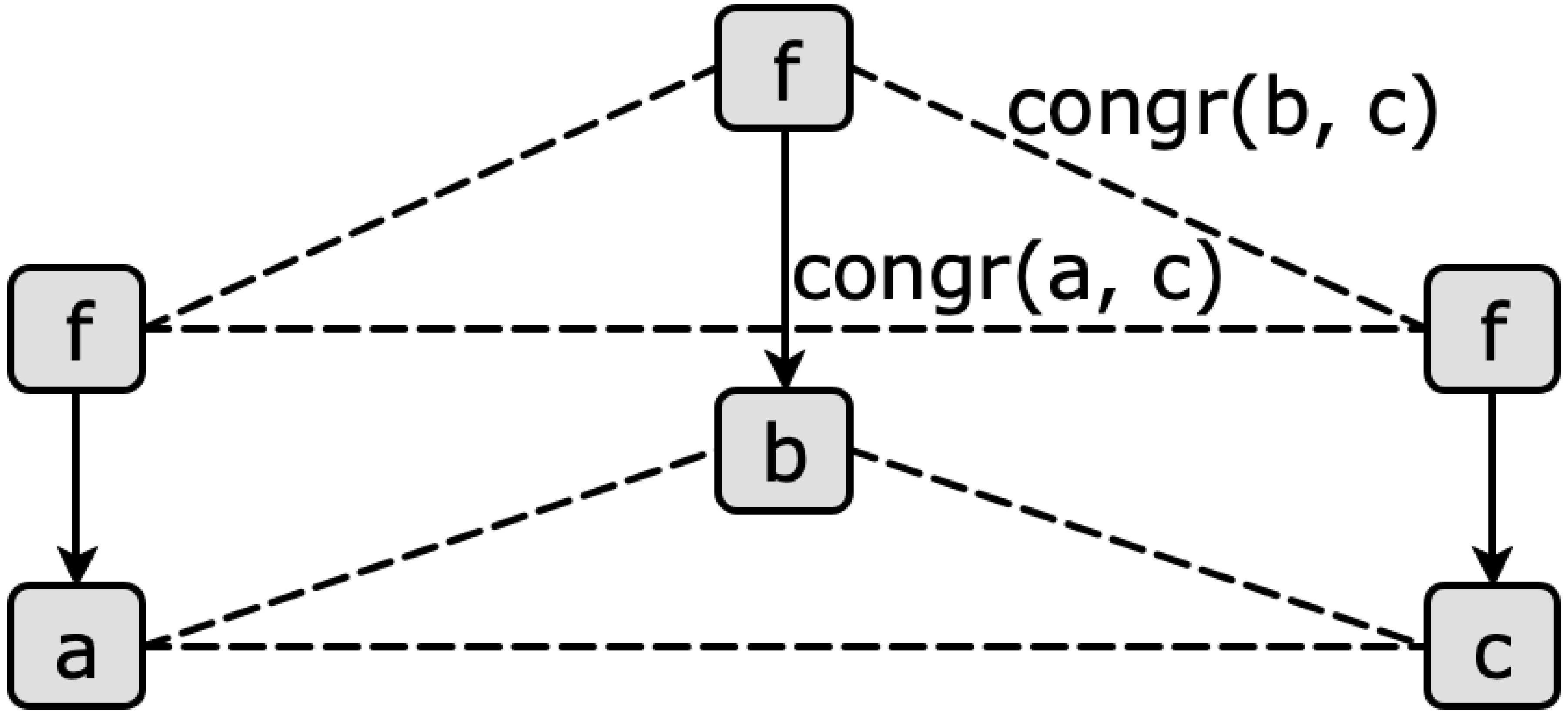
done!

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

Leveraging Additional Equalities

Inputs:

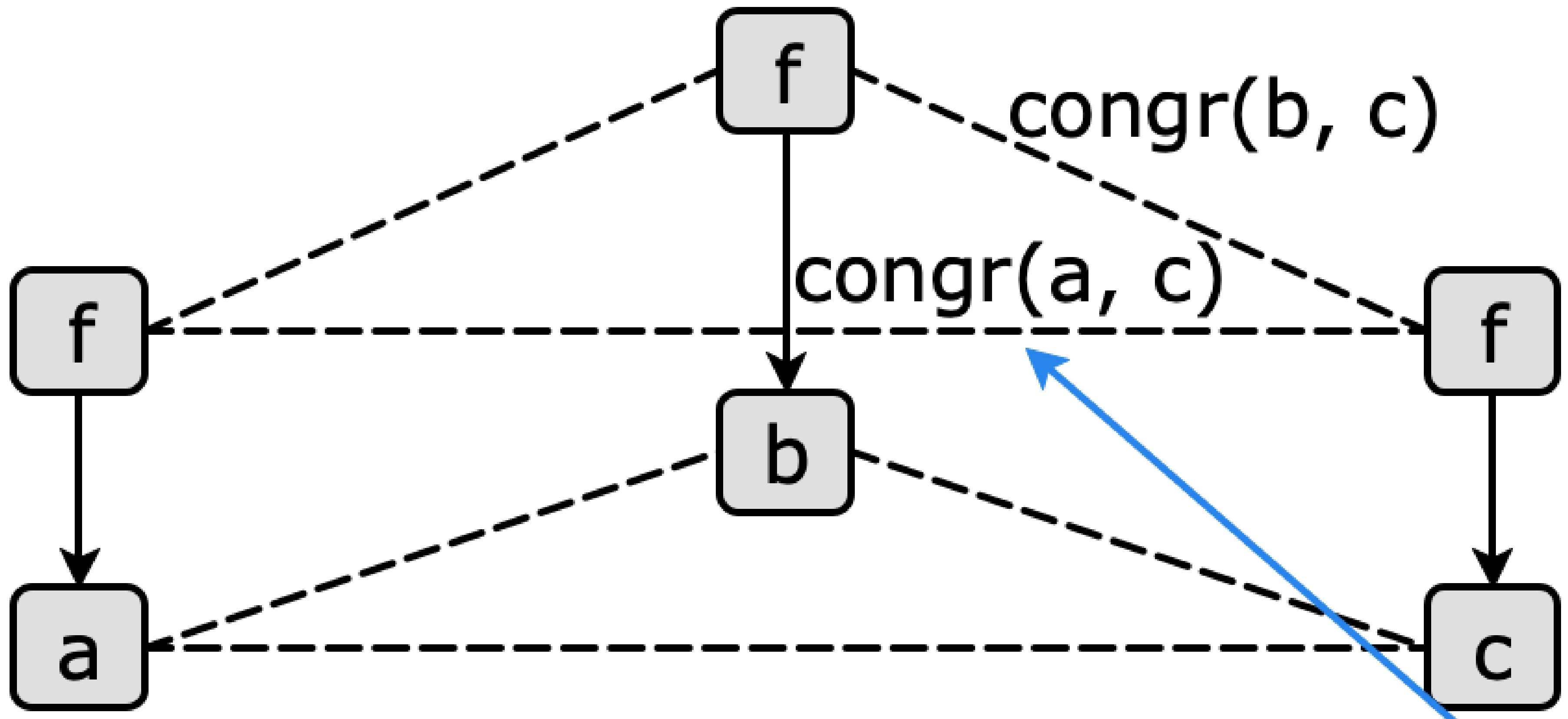
f(c)

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

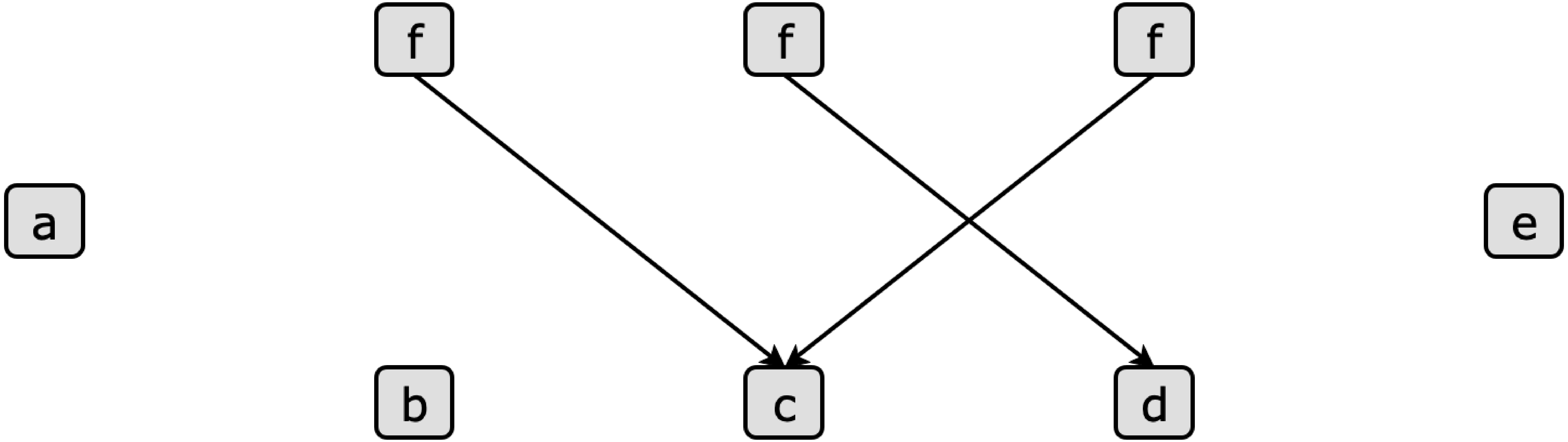
Key idea:

Try alternate path

new!

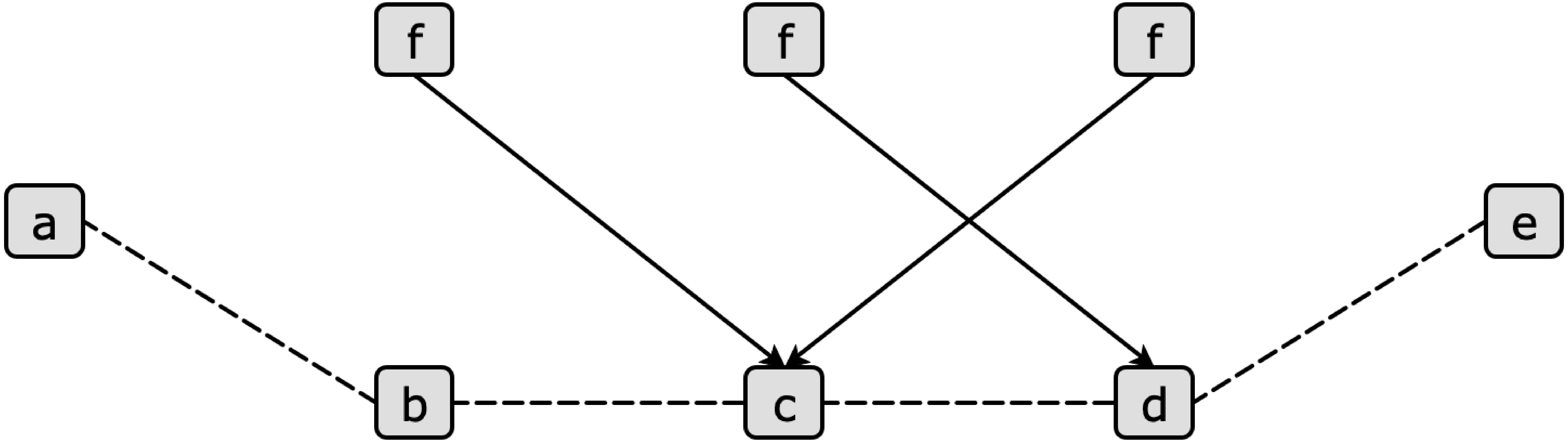
The Crux of The Problem

Prove a = e



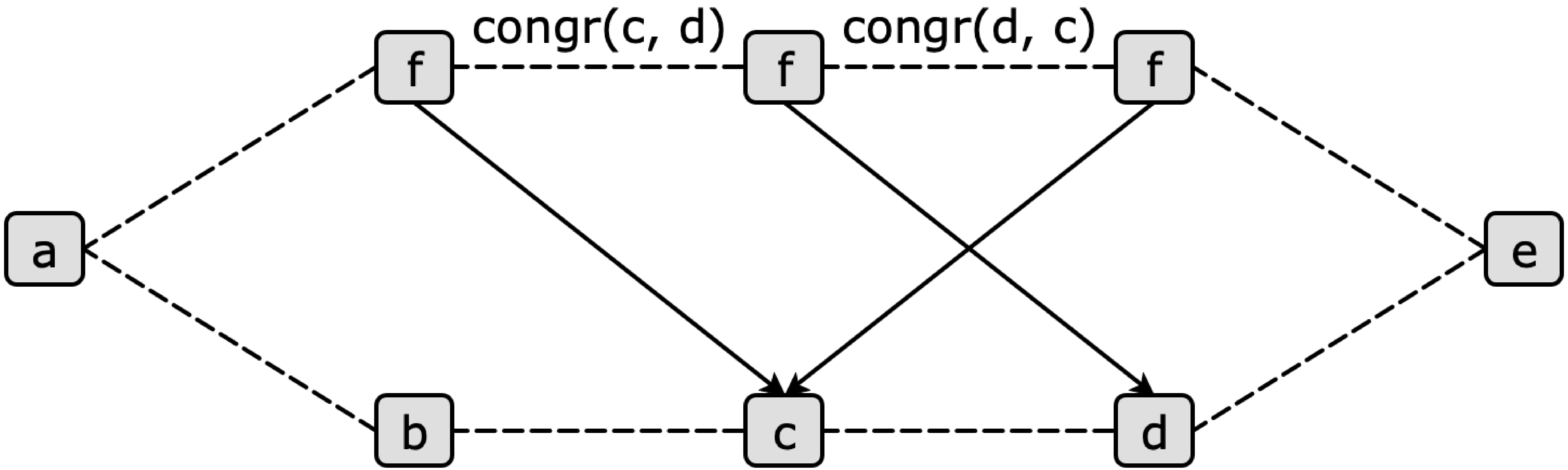
The Crux of The Problem

Prove a = e



The Crux of The Problem

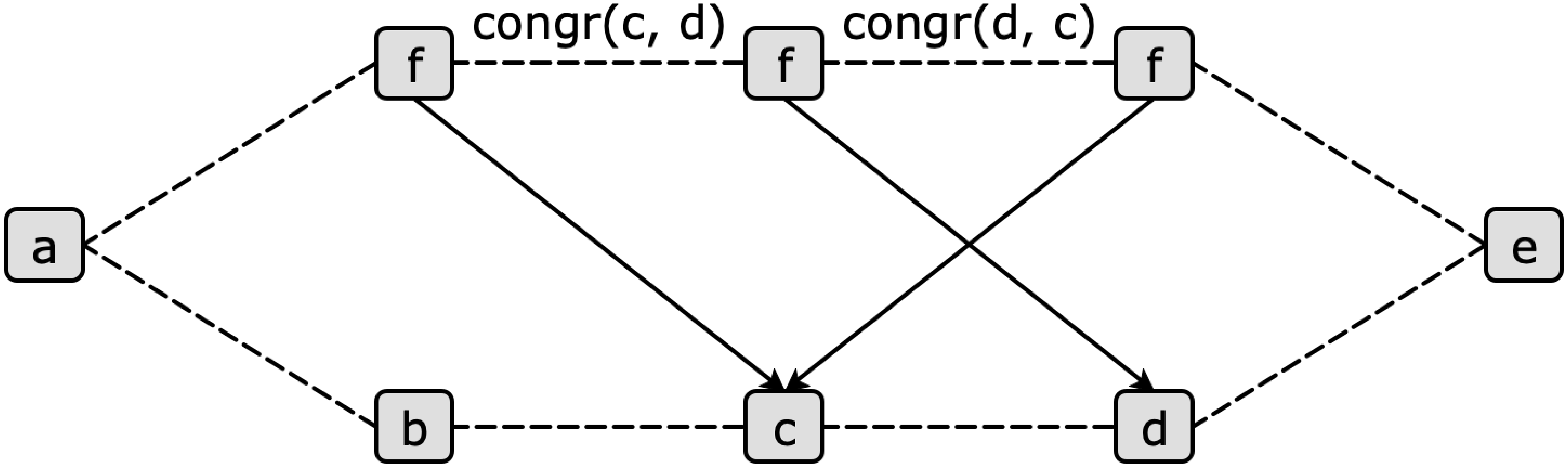
Prove a = e



The Crux of The Problem

Prove a = e

Bottom: size 4

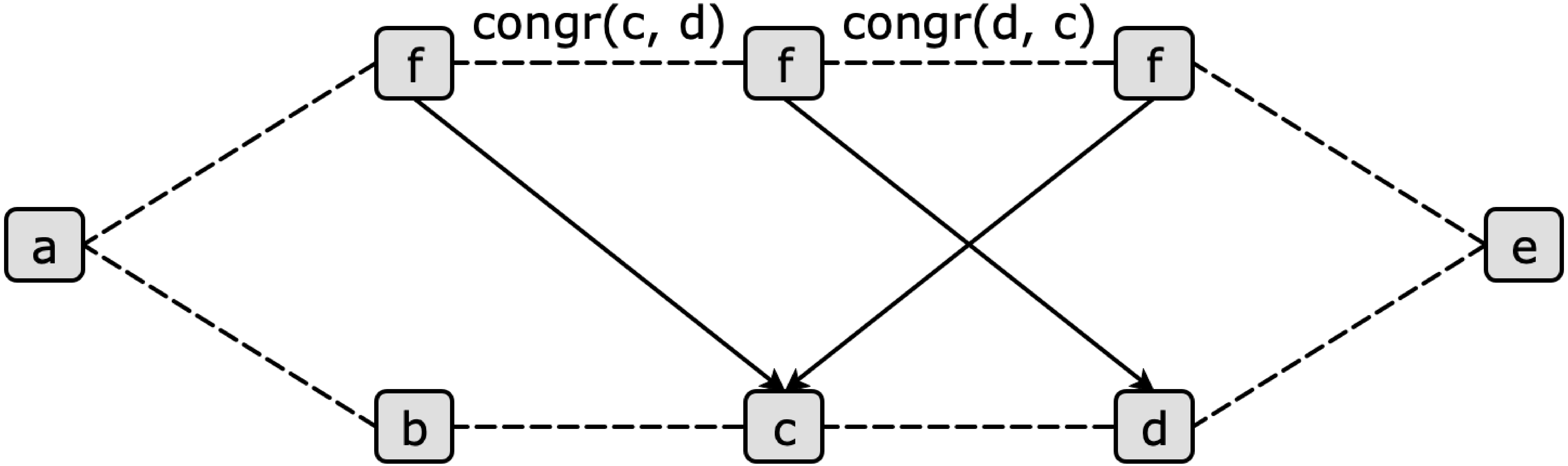


The Crux of The Problem

Prove a = e

Bottom: size 4

Top: ???

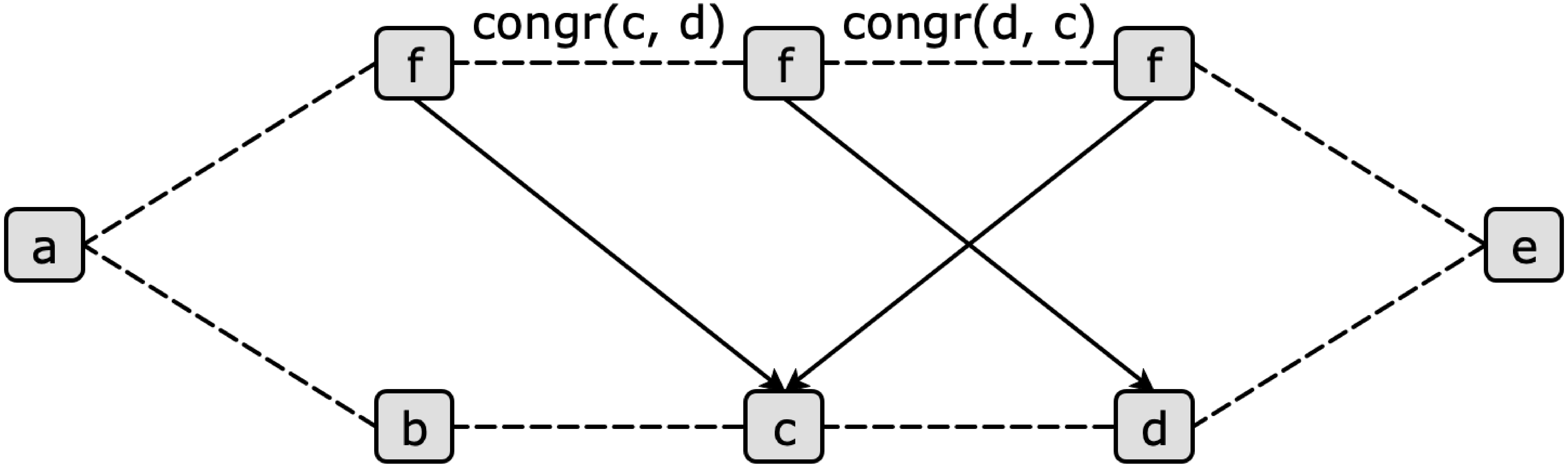


The Crux of The Problem

Prove a = e

Bottom: size 4

Top: size 3



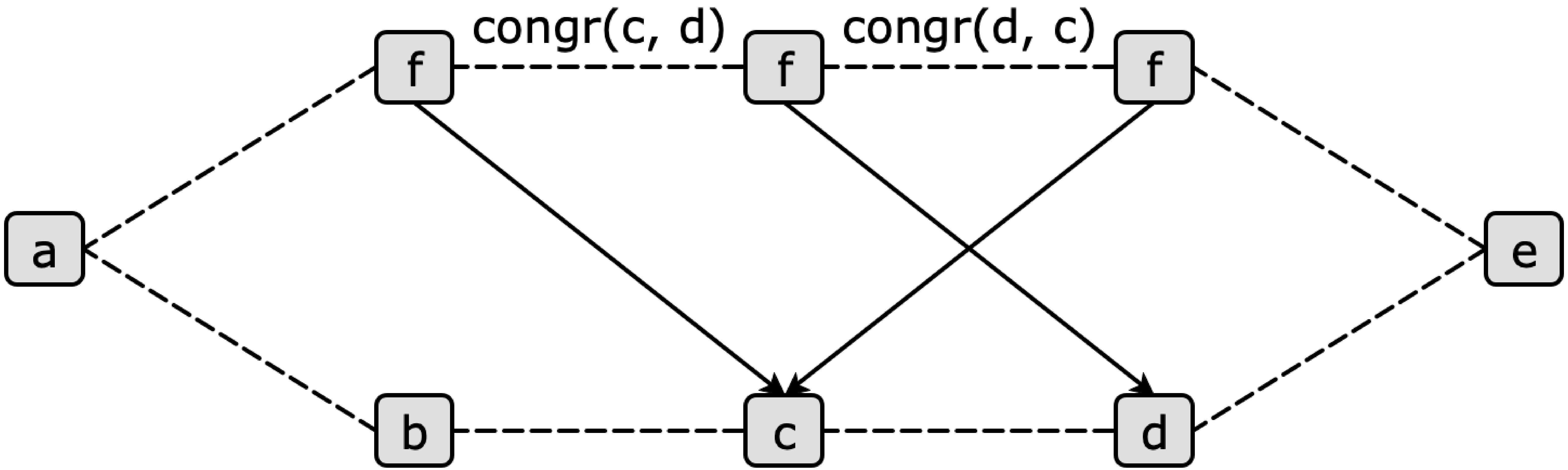
The Crux of The Problem

Prove a = e

Bottom: size 4

Top: size 3

Proof re-use is free

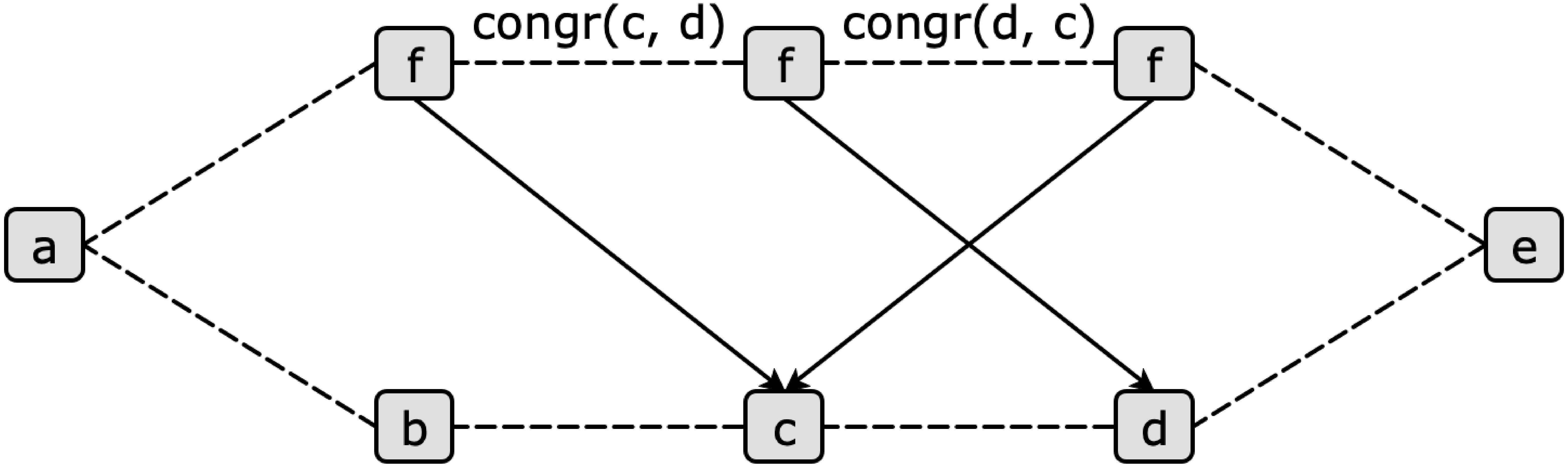


The Crux of The Problem

Prove a = e

Bottom: size 4

Top: size 3



Proof re-use is free

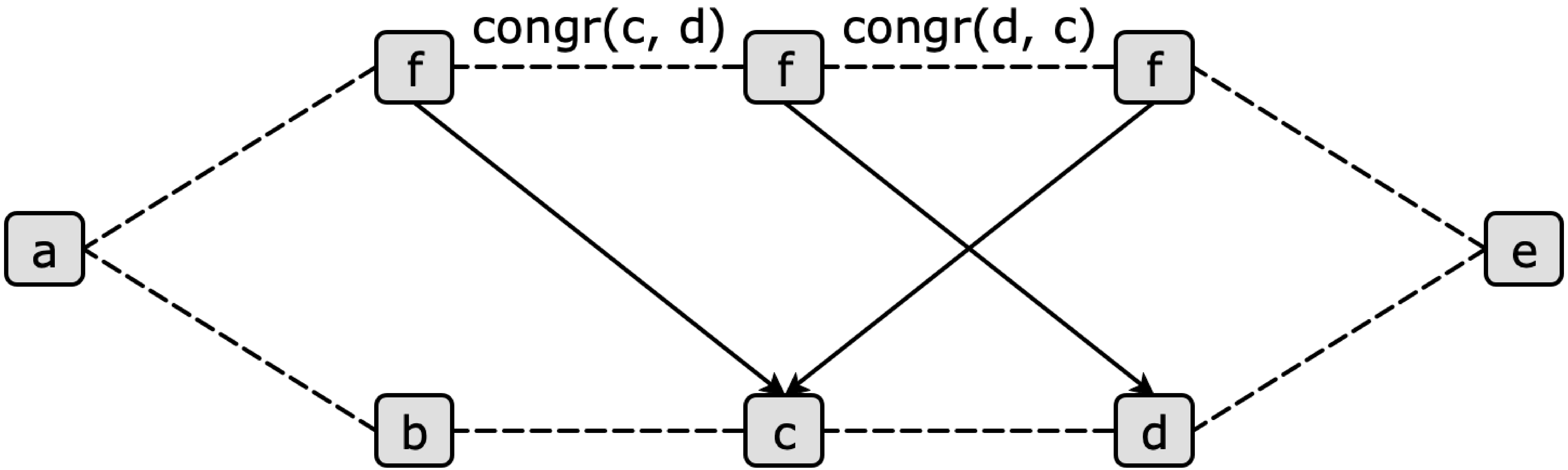
Subproofs can be complex

The Crux of The Problem

Prove a = e

Bottom: size 4

Top: size 3



Proof re-use is free

Subproofs can be complex

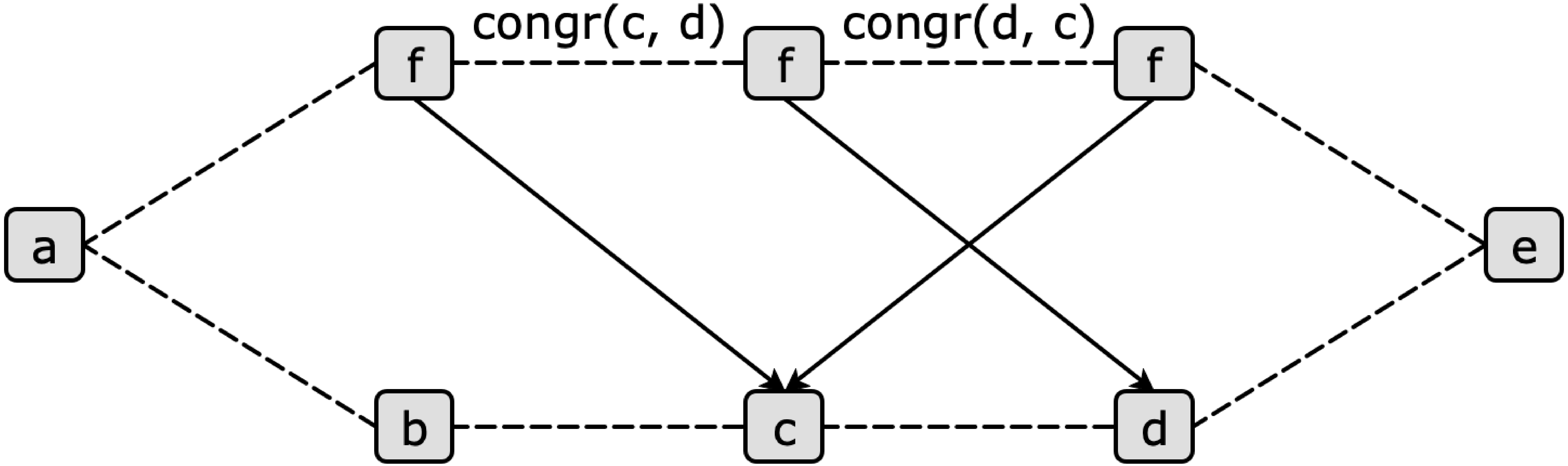
Exponential paths in graph

The Crux of The Problem

Prove a = e

Bottom: size 4

Top: size 3



Proof re-use is free

Subproofs can be complex

Exponential paths in graph

NP-Hard Problem (Fellner Et al.)

- [x] Motivation
- [x] Congruence Closure
- [x] Proofs from Congruence Closure
- [ ] Finding Small Proofs

27.2% smaller proofs!

Idea: Shortest Path?

Inputs:

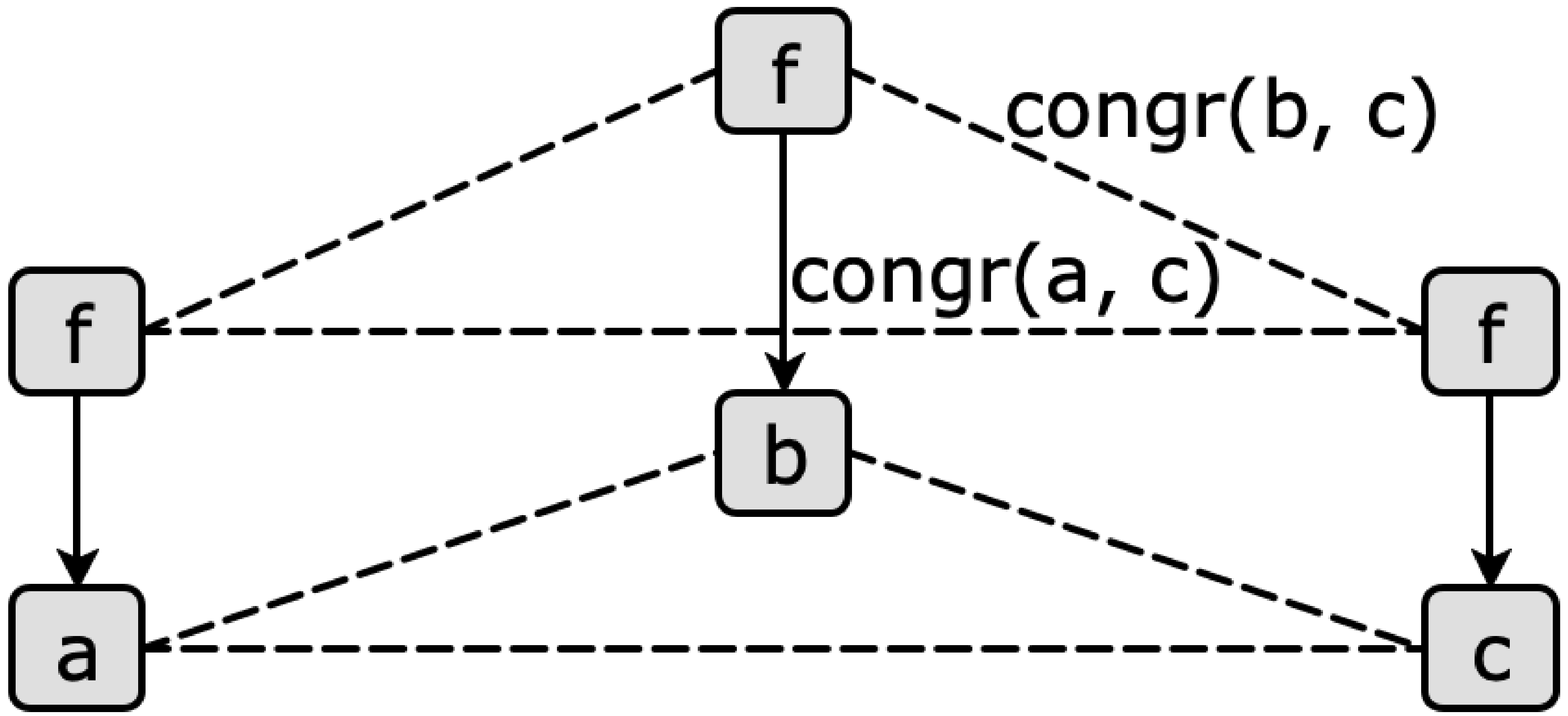
f(c)

f(a) = f(b)

a = b

b = c

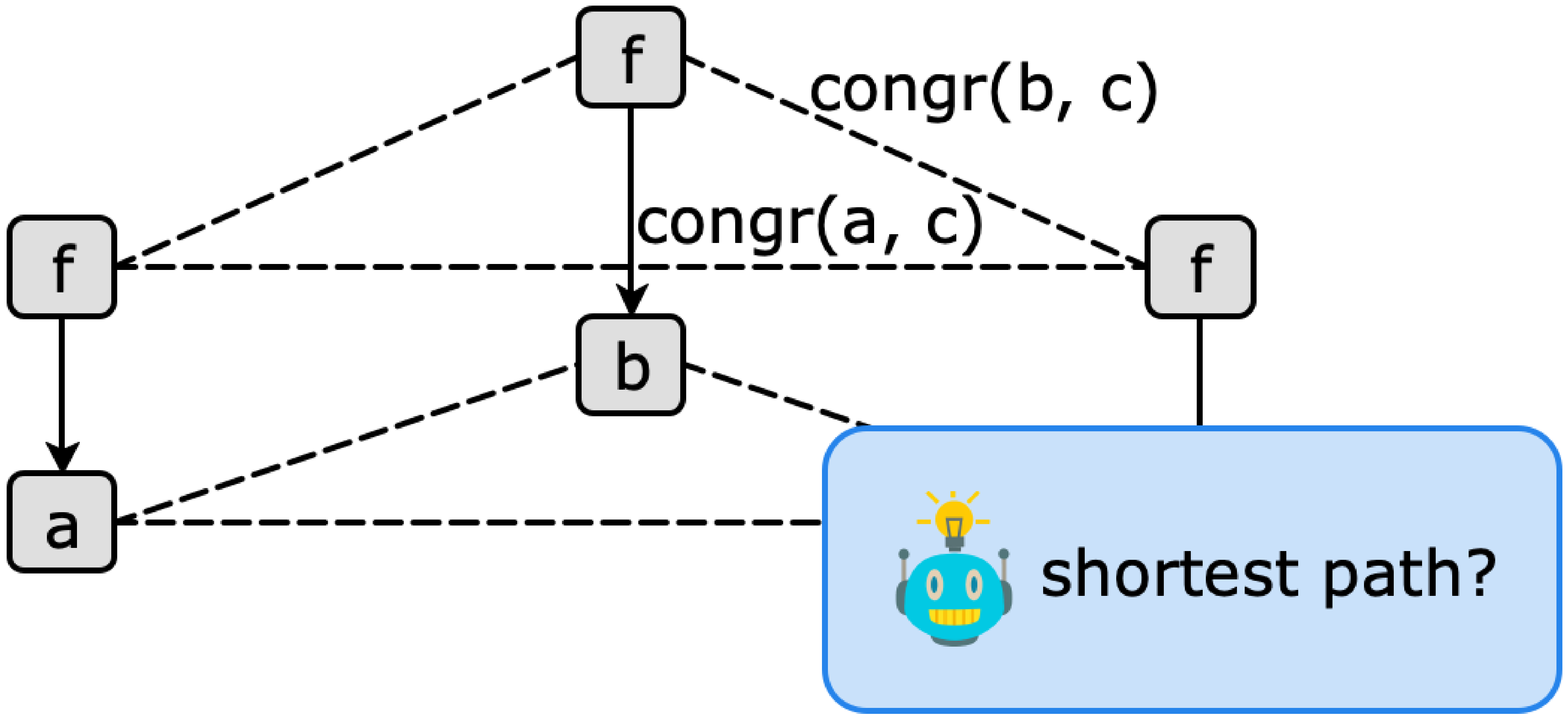
a = c useful



Idea: Shortest Path?

Inputs:

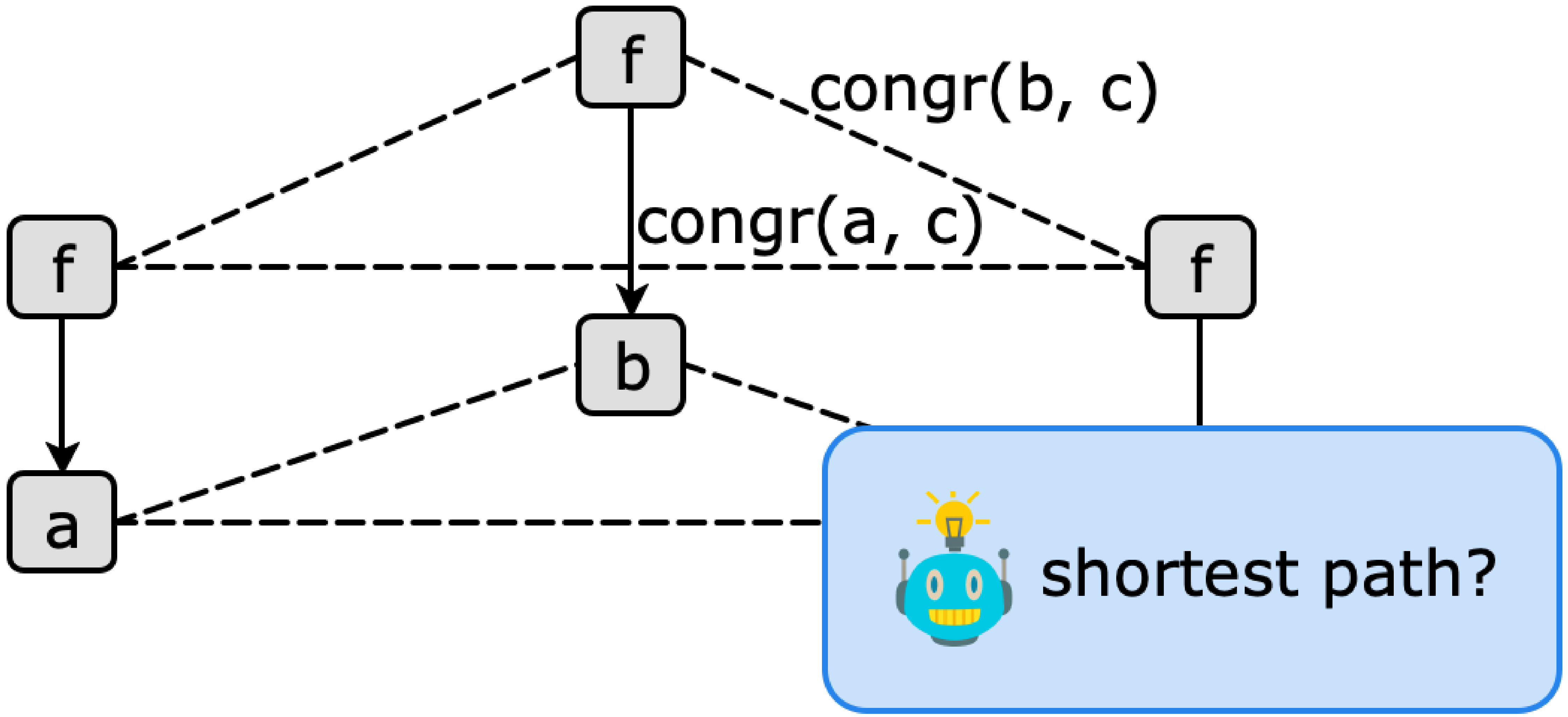
- f(c)
- f(a) = f(b)
- a = b
- b = c
- a = c useful



Idea: Shortest Path?

Inputs:

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



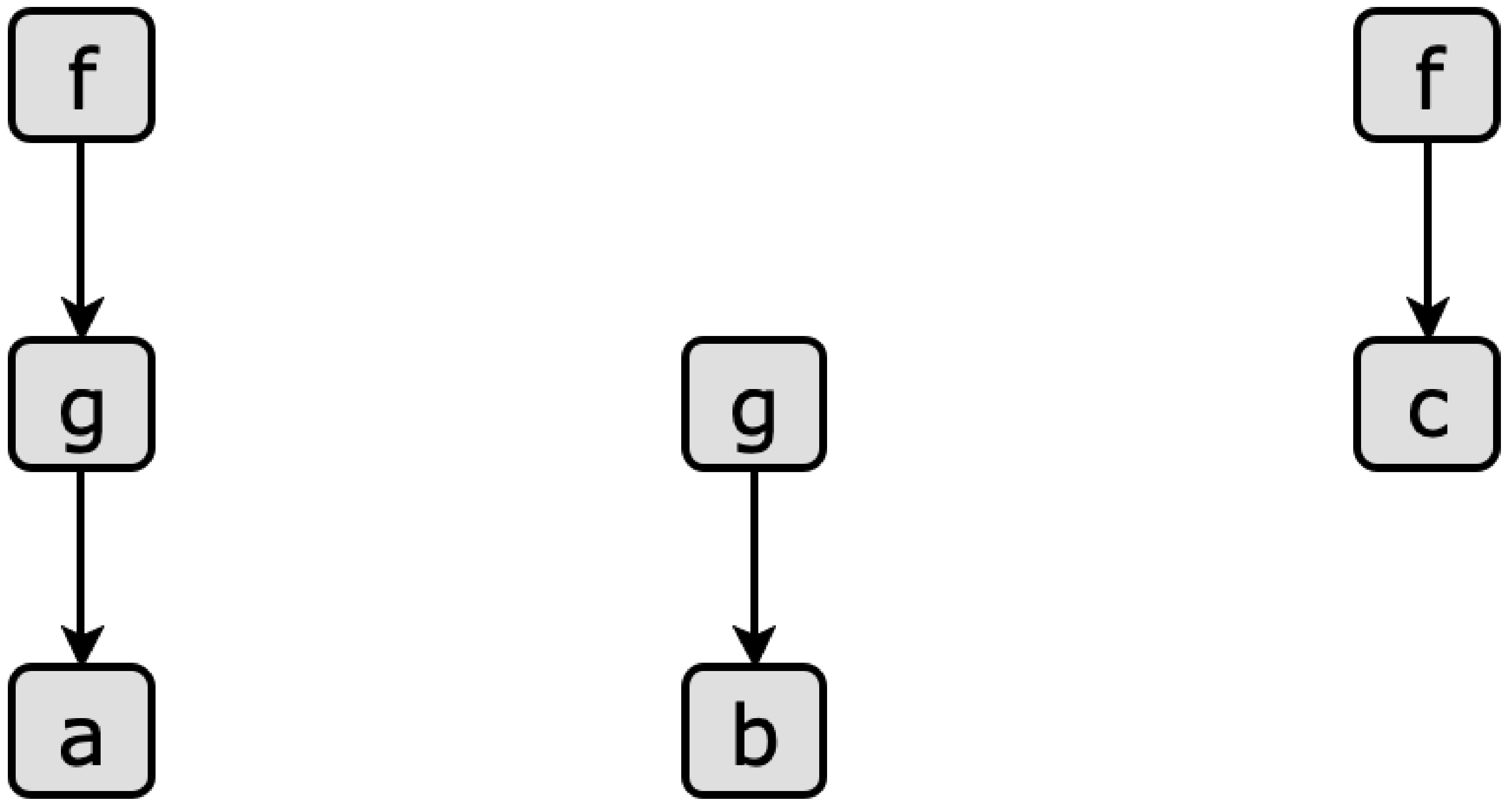
Problem: how big are congruence edges?

Proof Size Estimation

Inputs:

a = b

g(b) = c

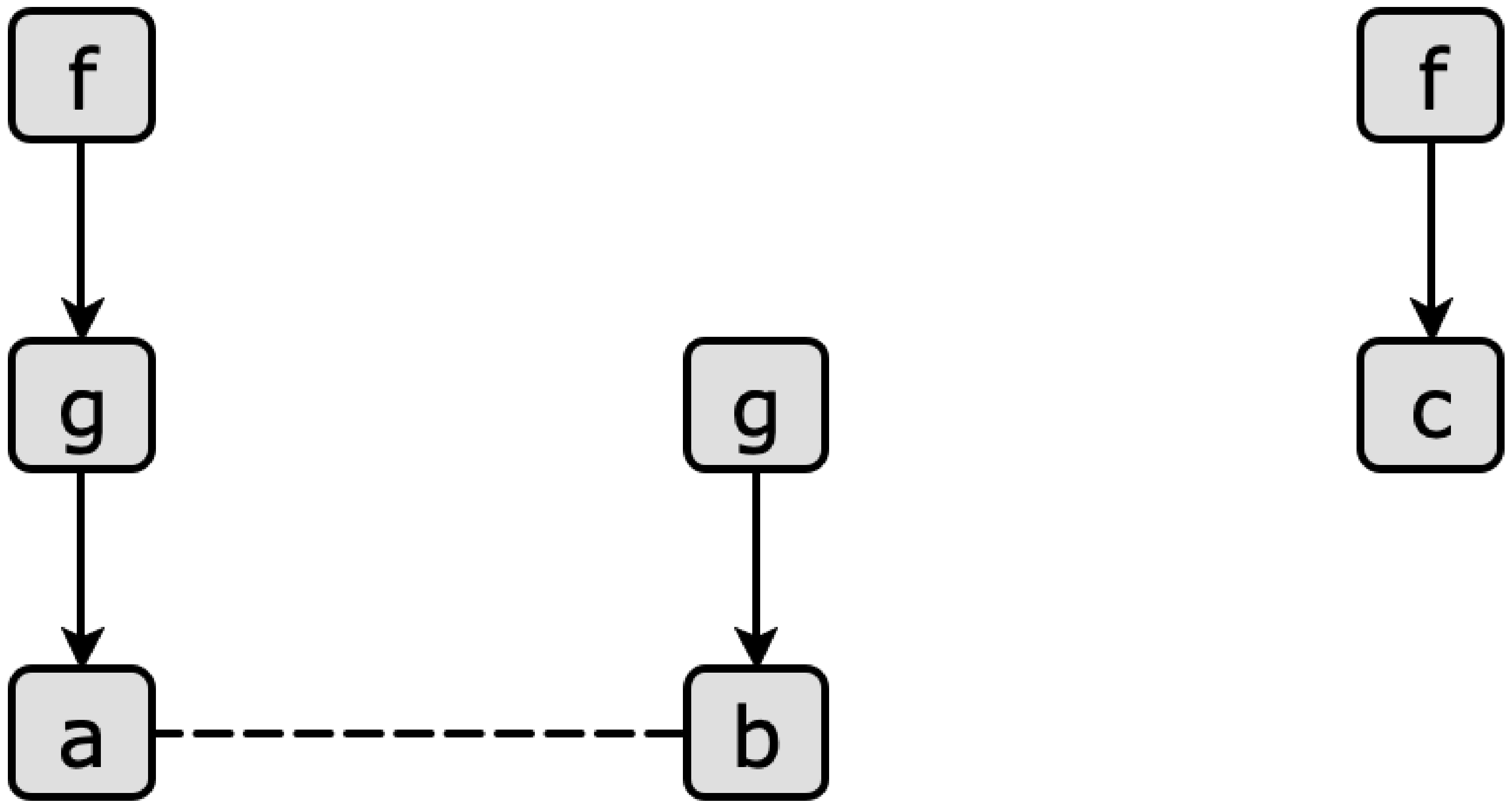


Proof Size Estimation

Inputs:

a = b

g(b) = c

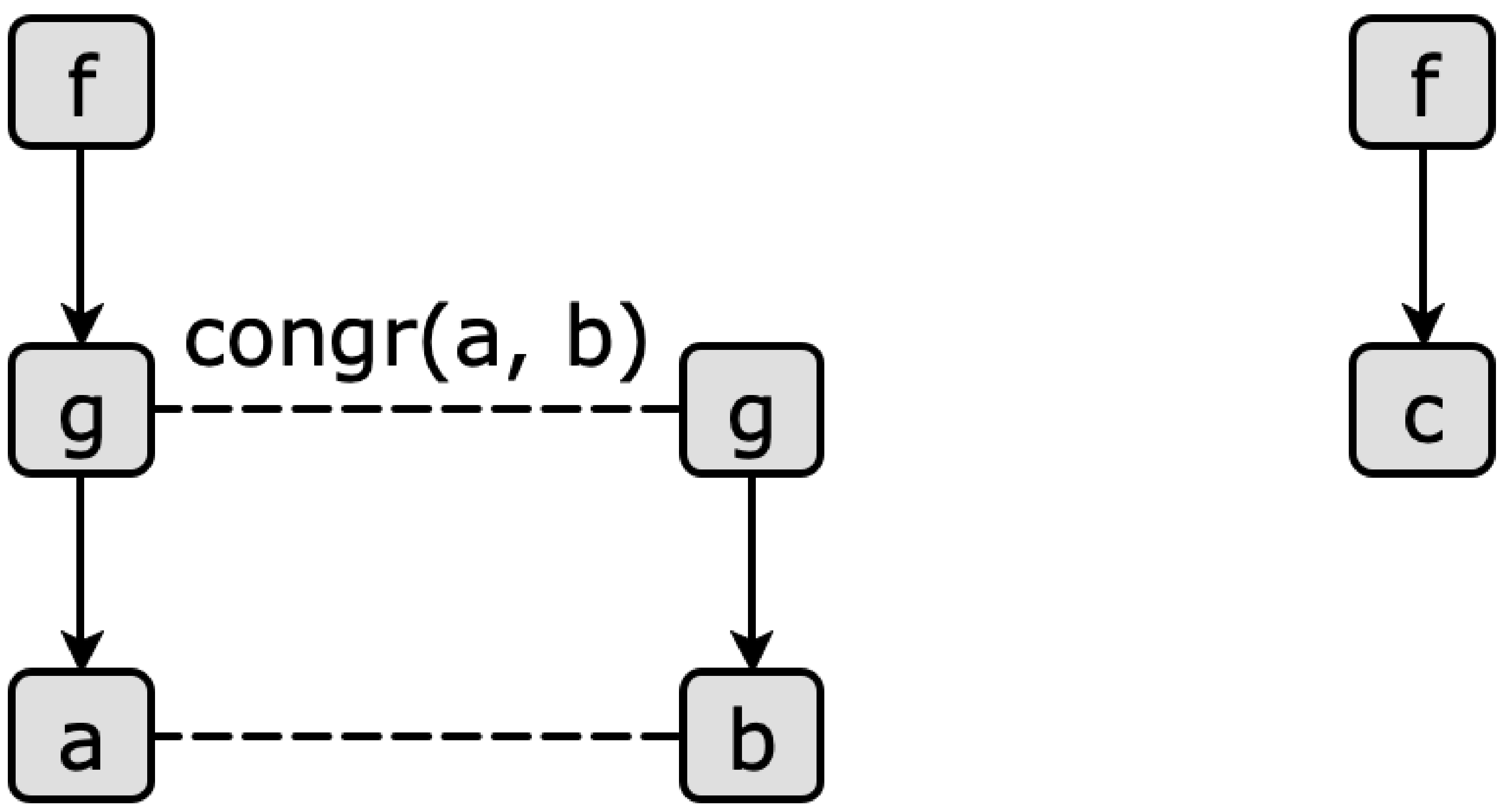


Proof Size Estimation

Inputs:

a = b

g(b) = c

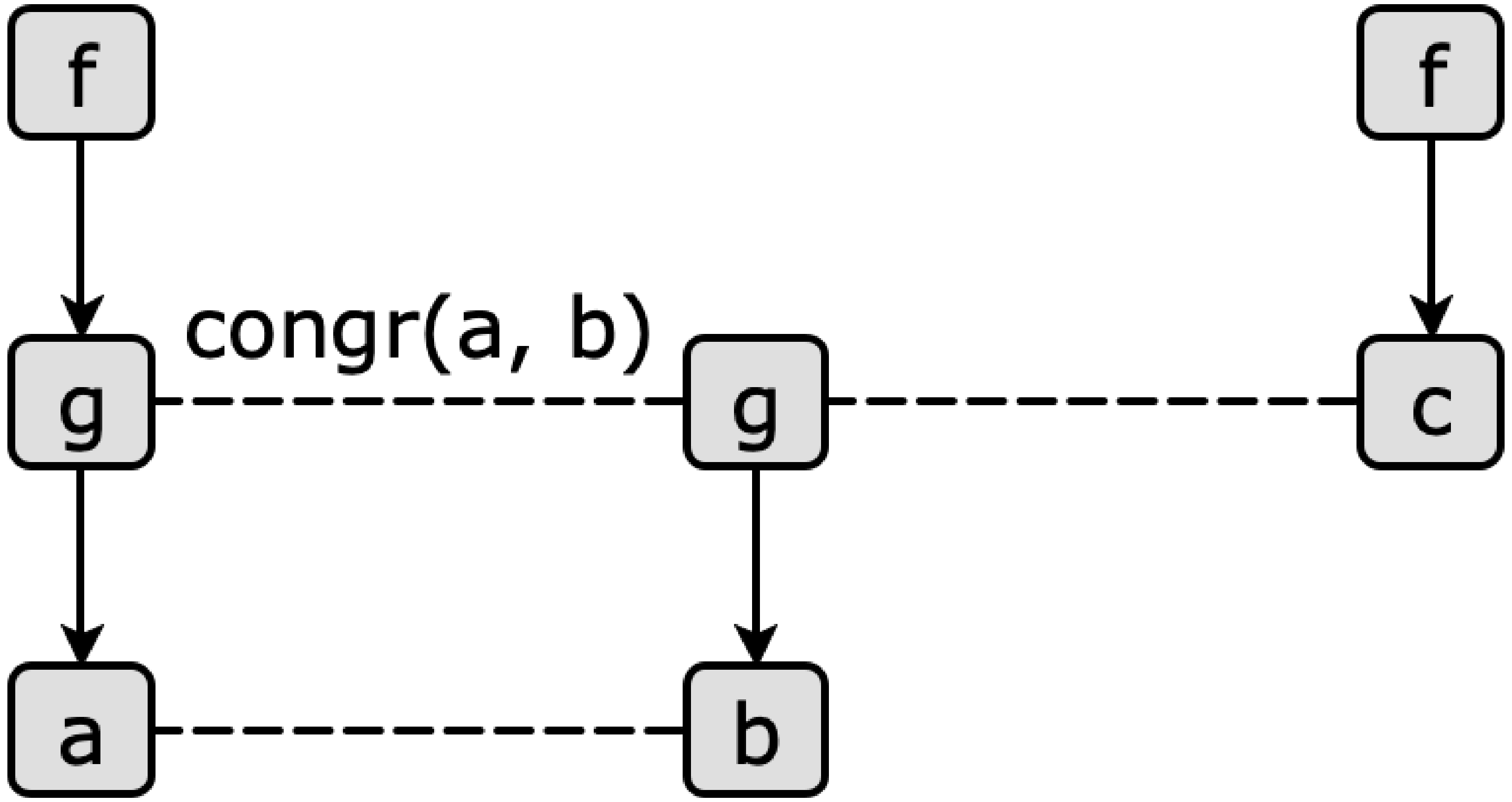


Proof Size Estimation

Inputs:

a = b

g(b) = c

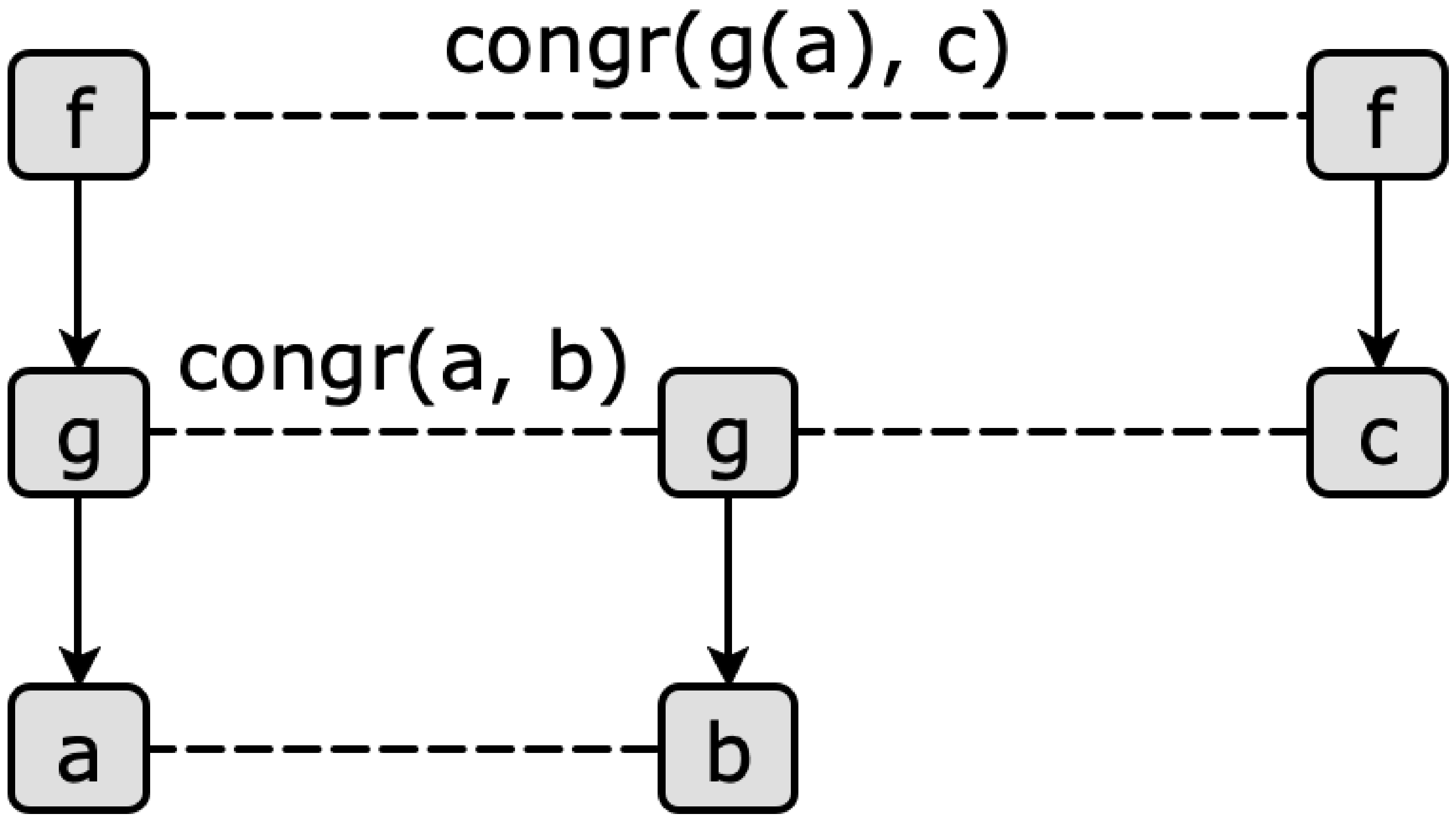


Proof Size Estimation

Inputs:

a = b

g(b) = c

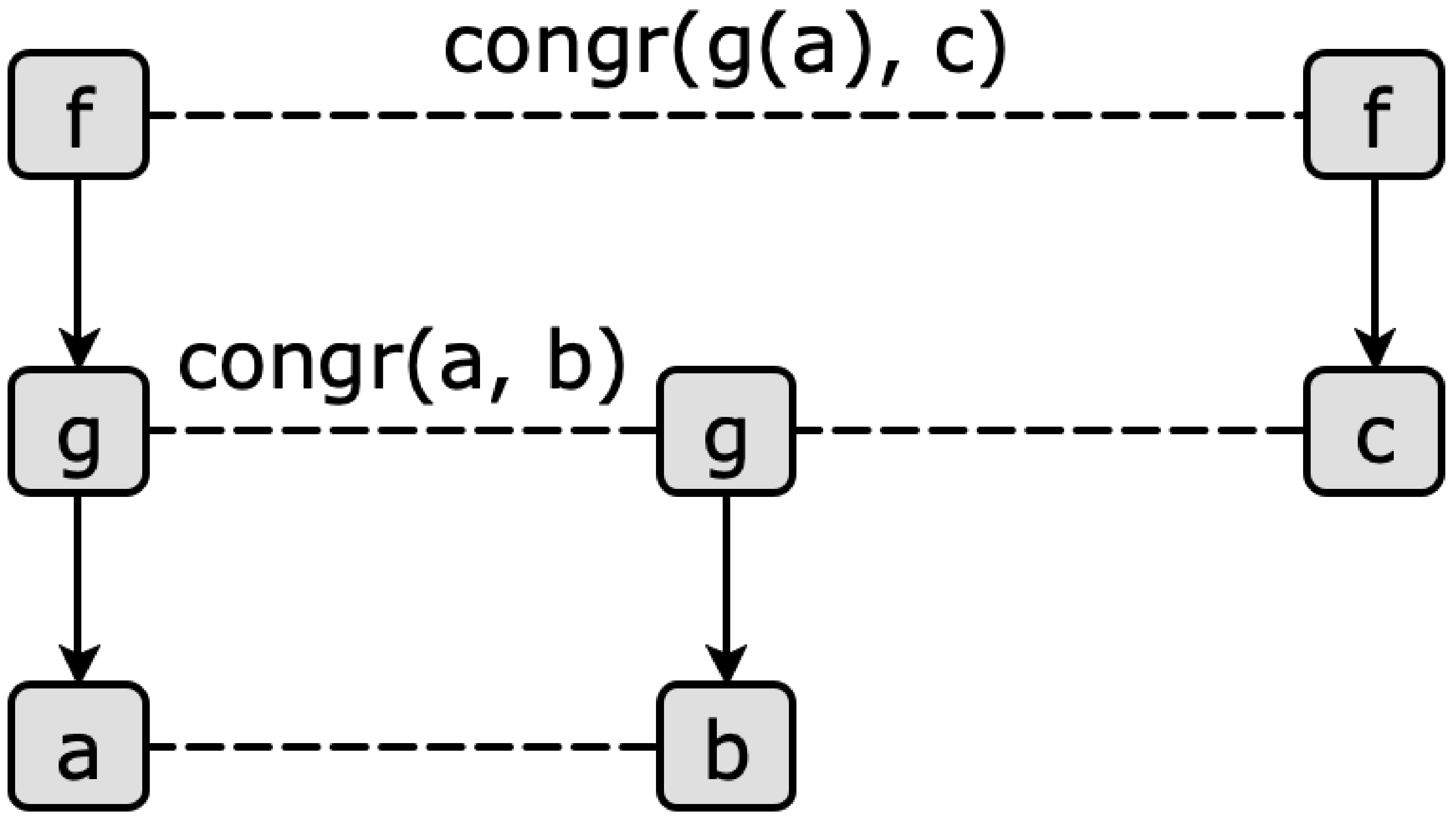


Proof Size Estimation

Inputs:

a = b

g(b) = c



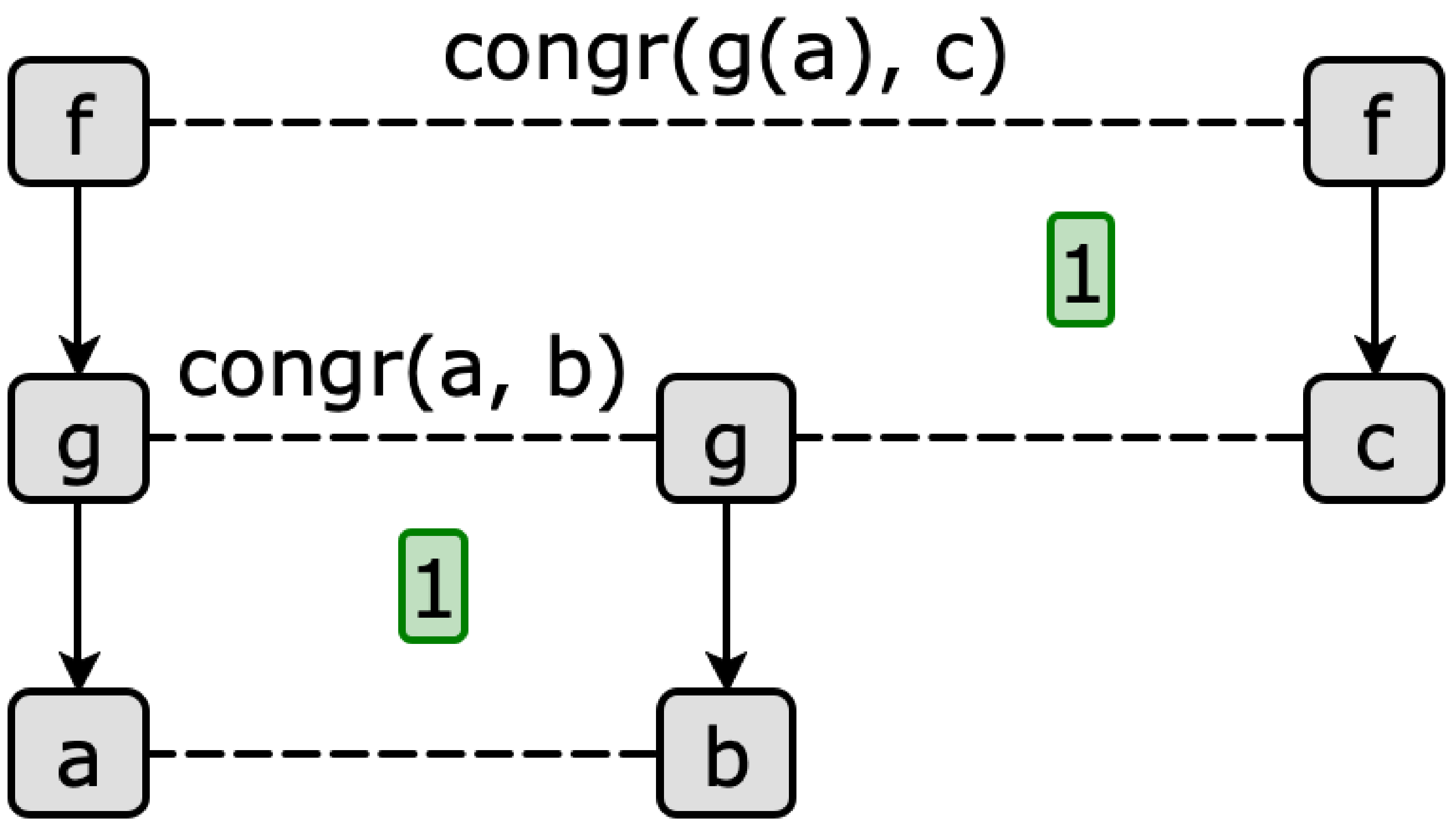
Key idea: compute estimates bottom-up

Proof Size Estimation

Inputs:

a = b

g(b) = c



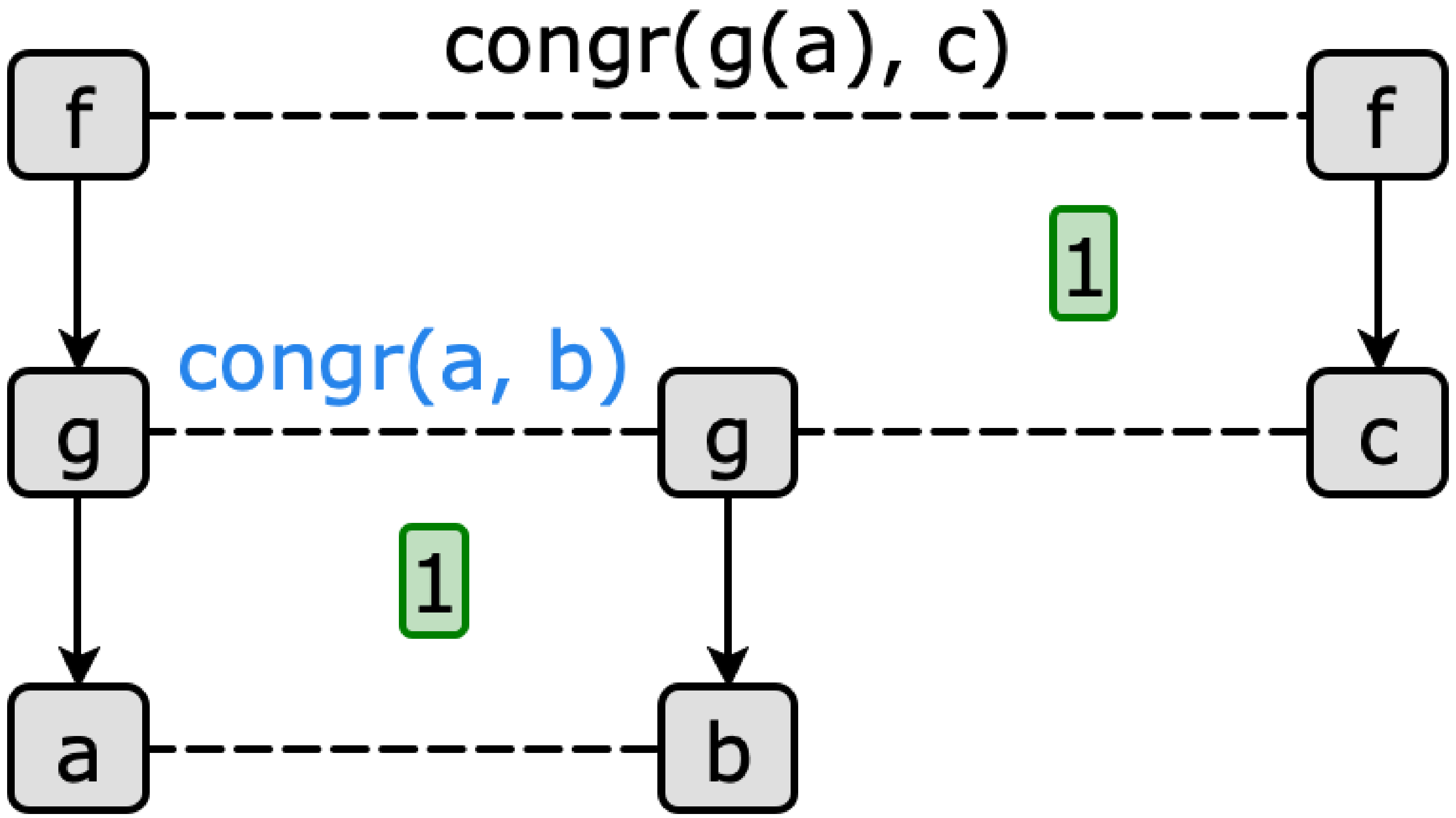
Key idea: compute estimates bottom-up

Proof Size Estimation

Inputs:

a = b

g(b) = c



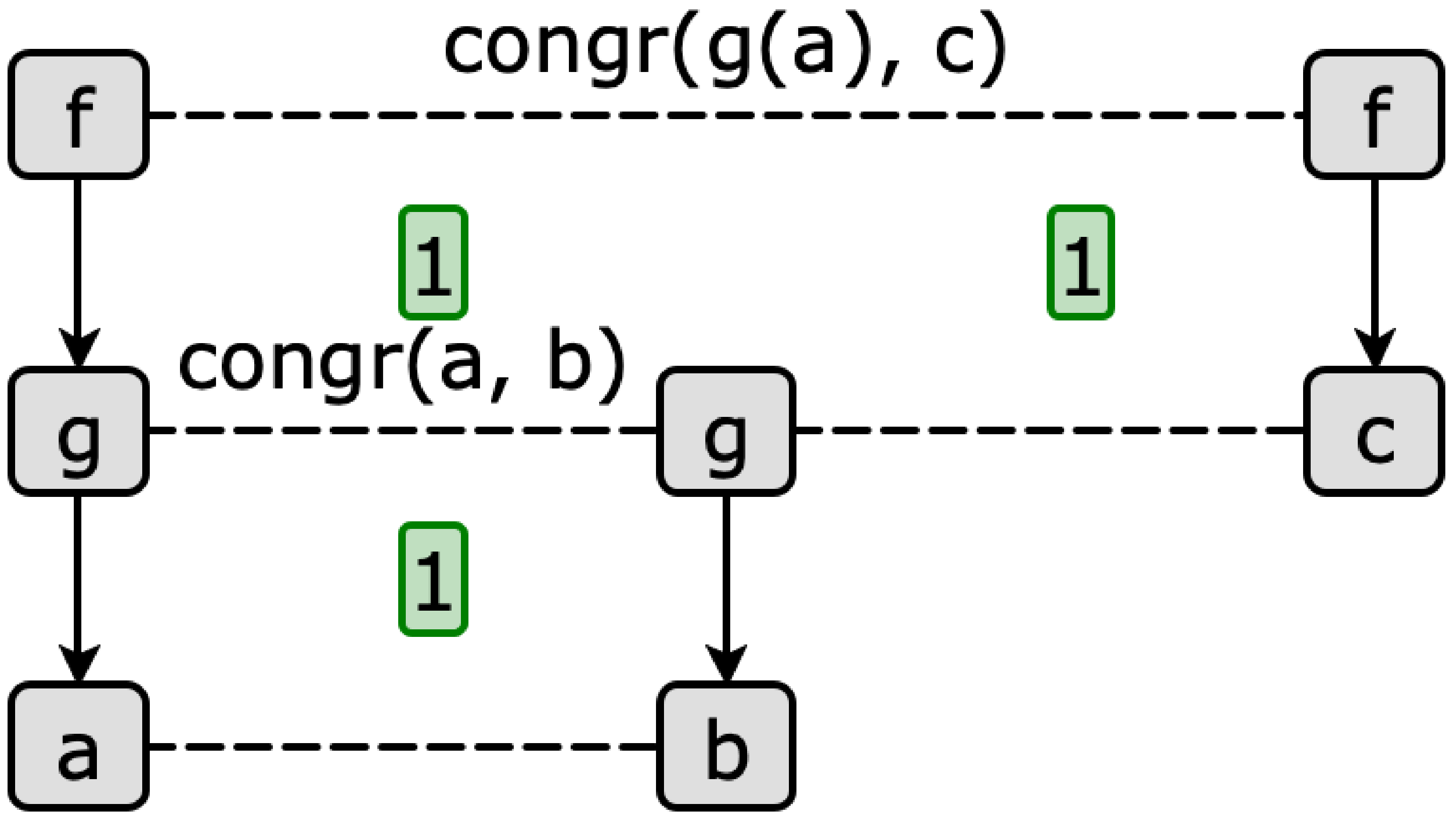
Key idea: compute estimates bottom-up

Proof Size Estimation

Inputs:

a = b

g(b) = c



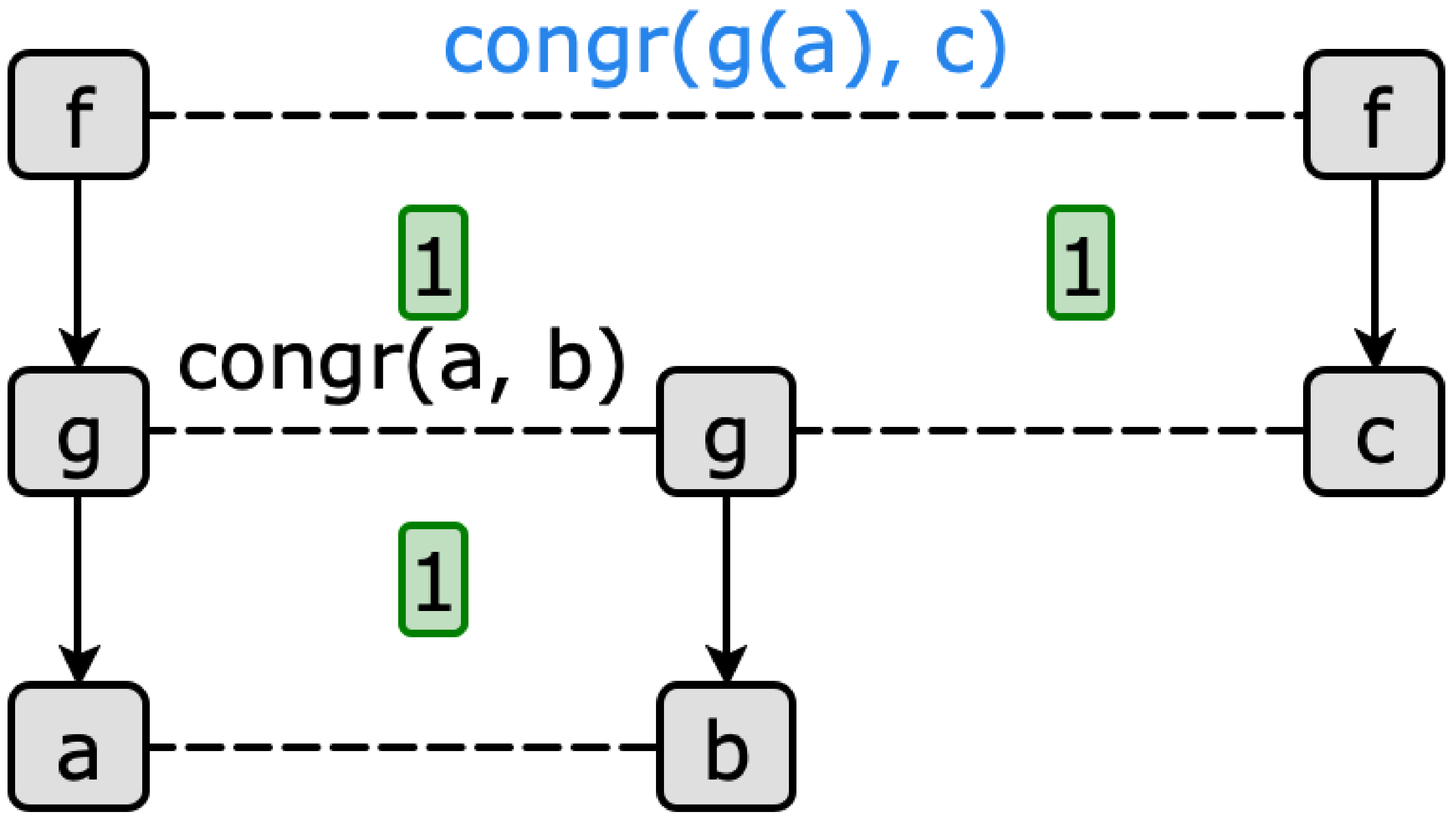
Key idea: compute estimates bottom-up

Proof Size Estimation

Inputs:

a = b

g(b) = c



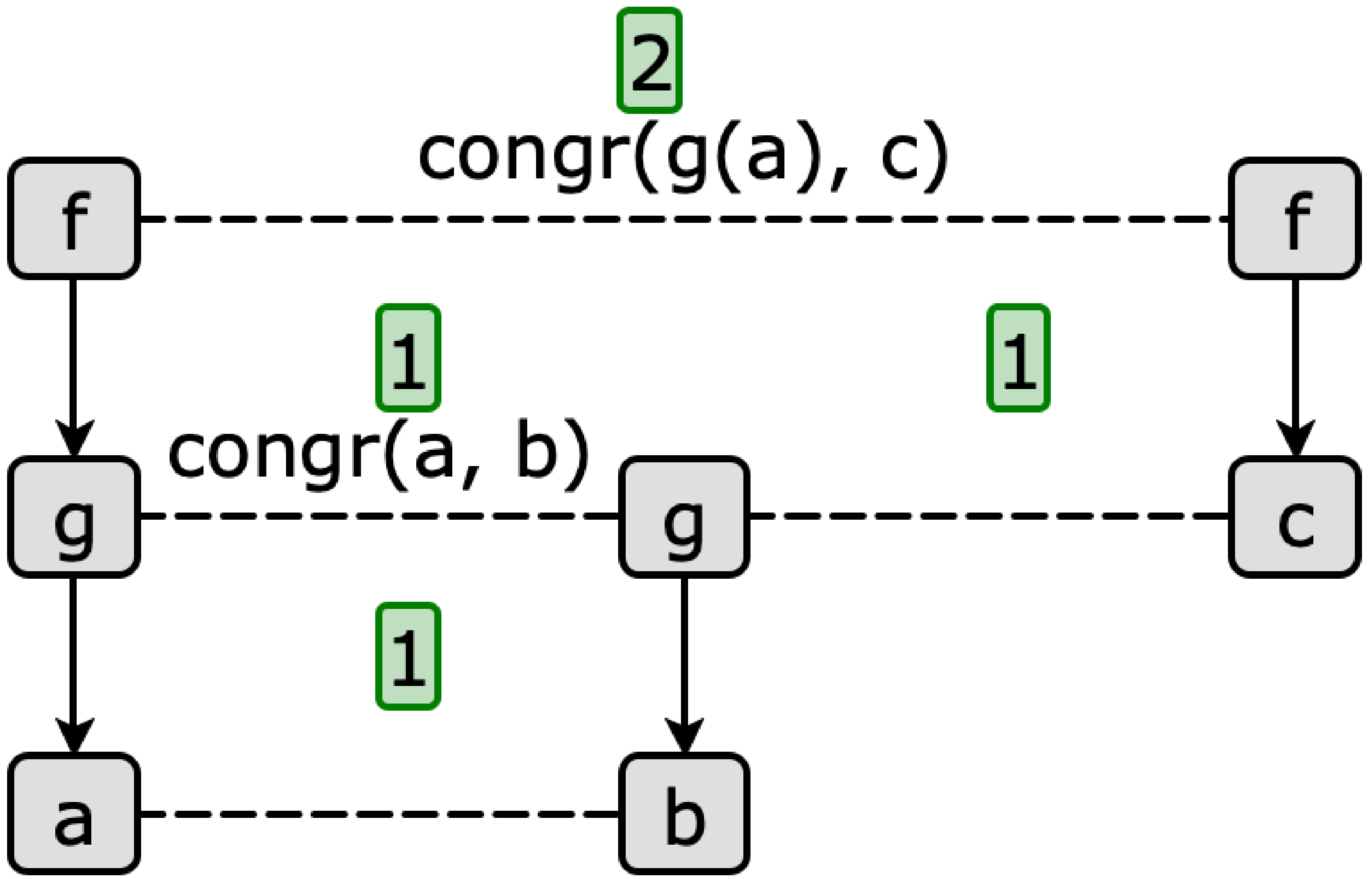
Key idea: compute estimates bottom-up

Proof Size Estimation

Inputs:

a = b

g(b) = c



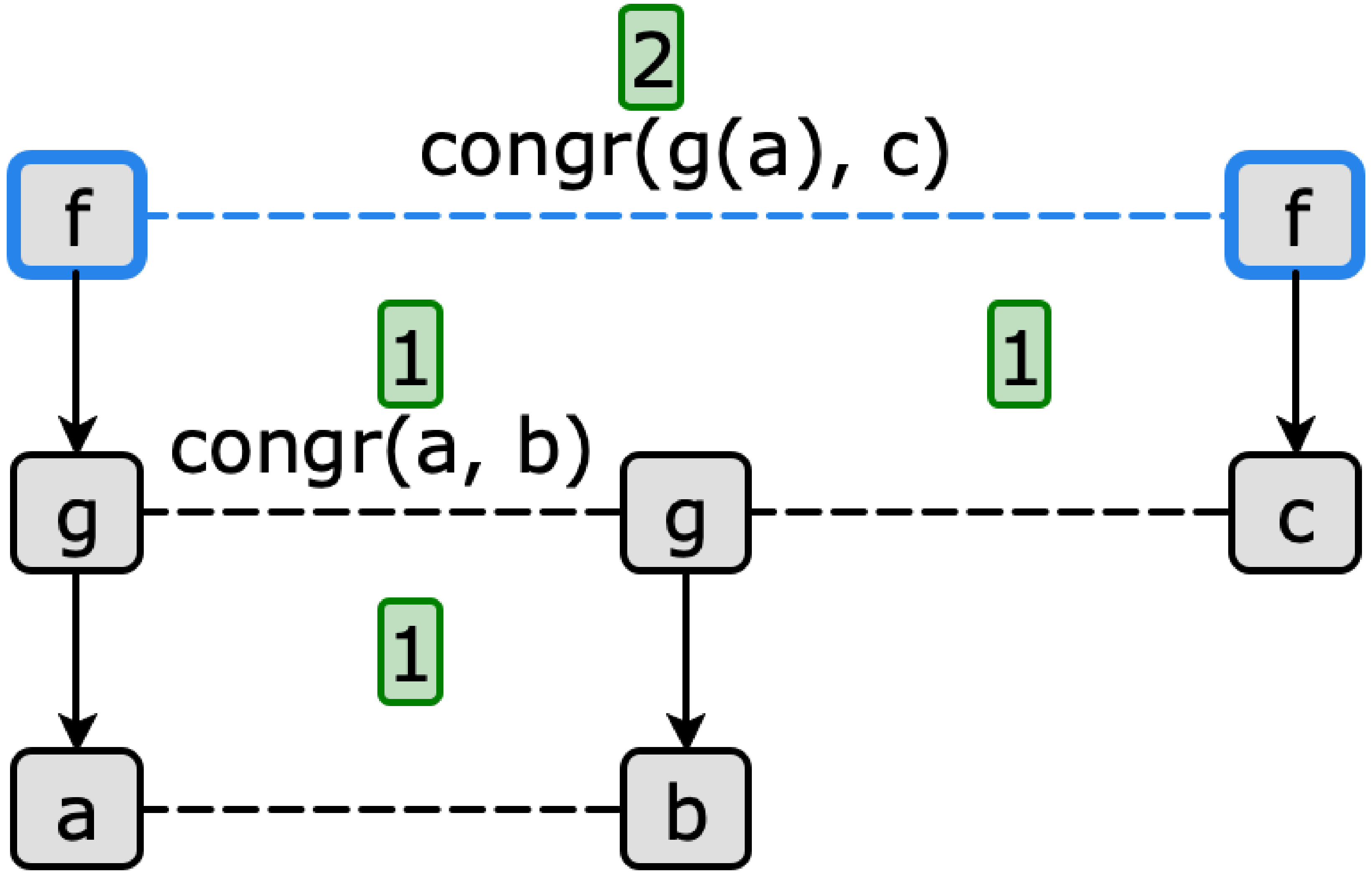
Key idea: compute estimates bottom-up

Proof Size Estimation

Inputs:

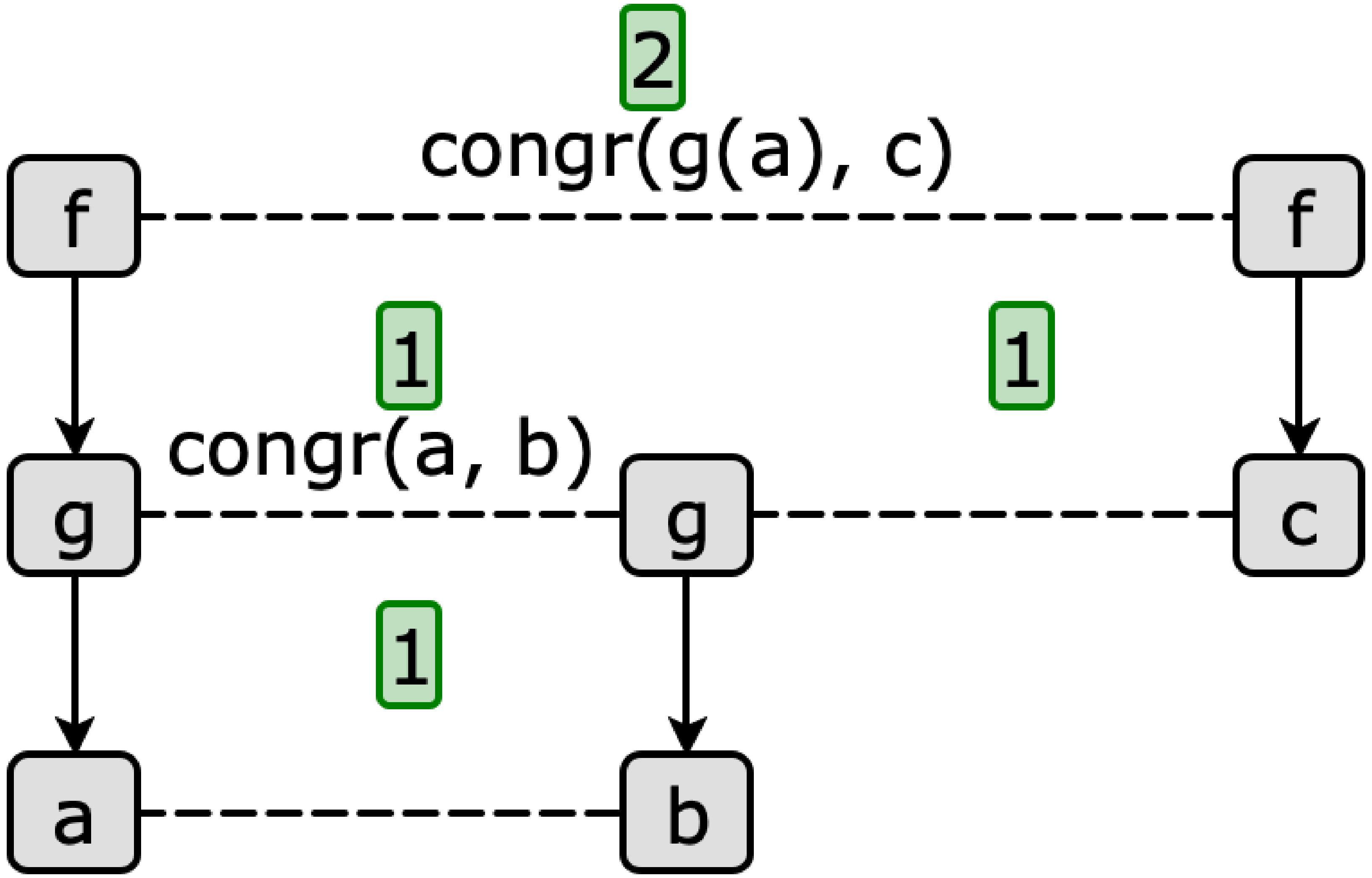
a = b

g(b) = c



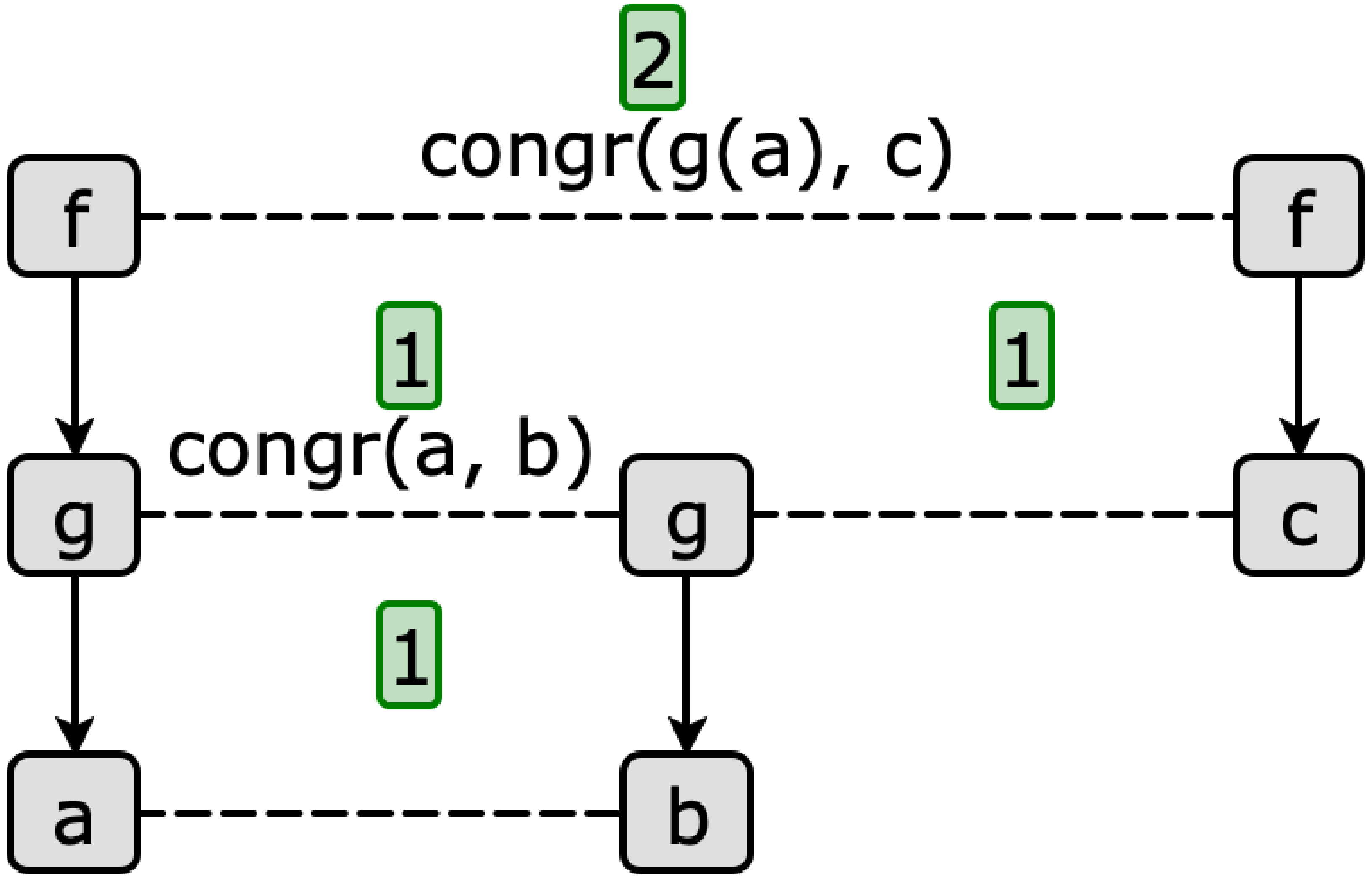
Key idea: compute estimates bottom-up

Putting it All Together



Putting it All Together

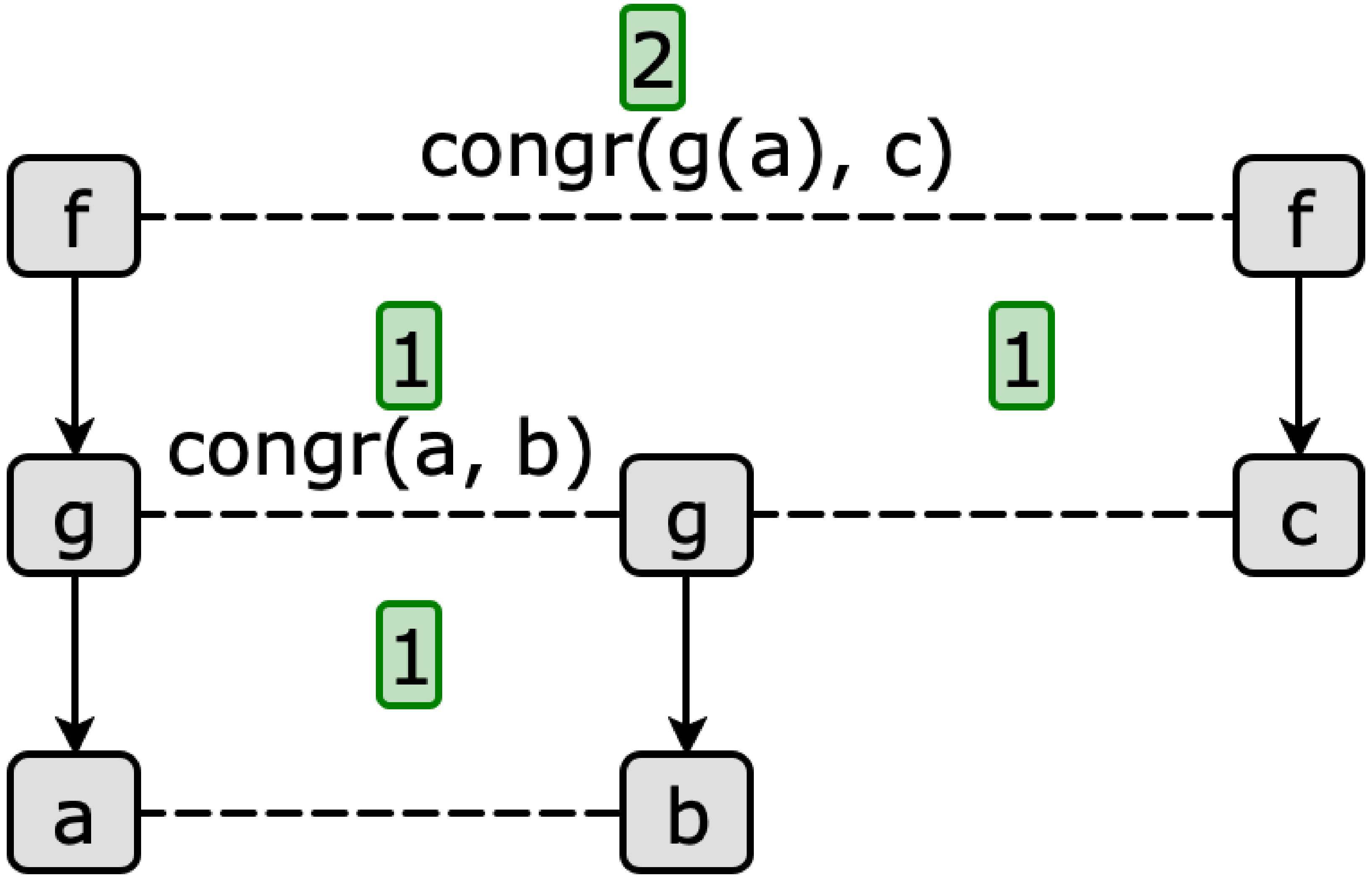
1. Compute size estimates



Putting it All Together

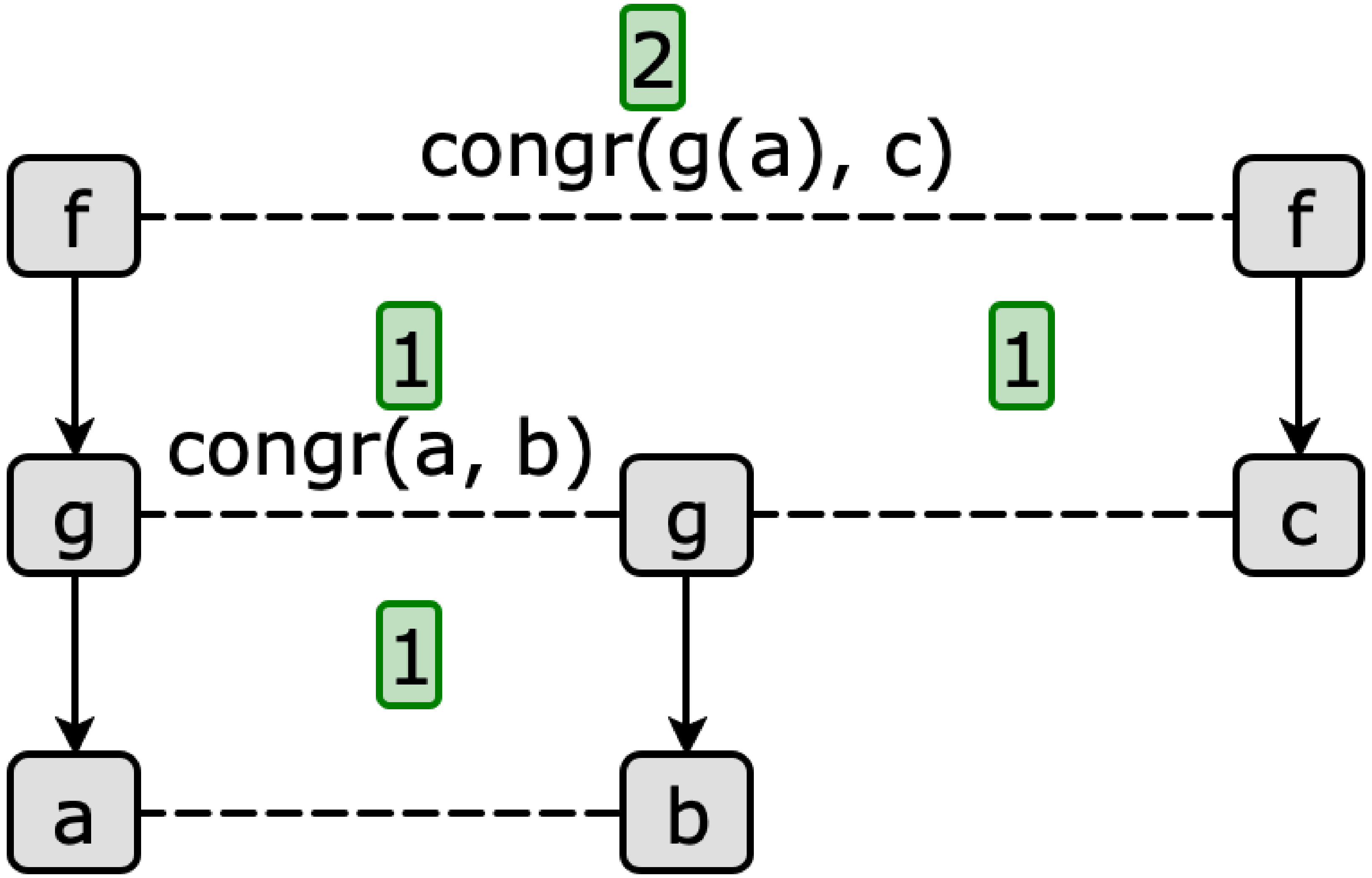
1. Compute size estimates

2. Find shortest path

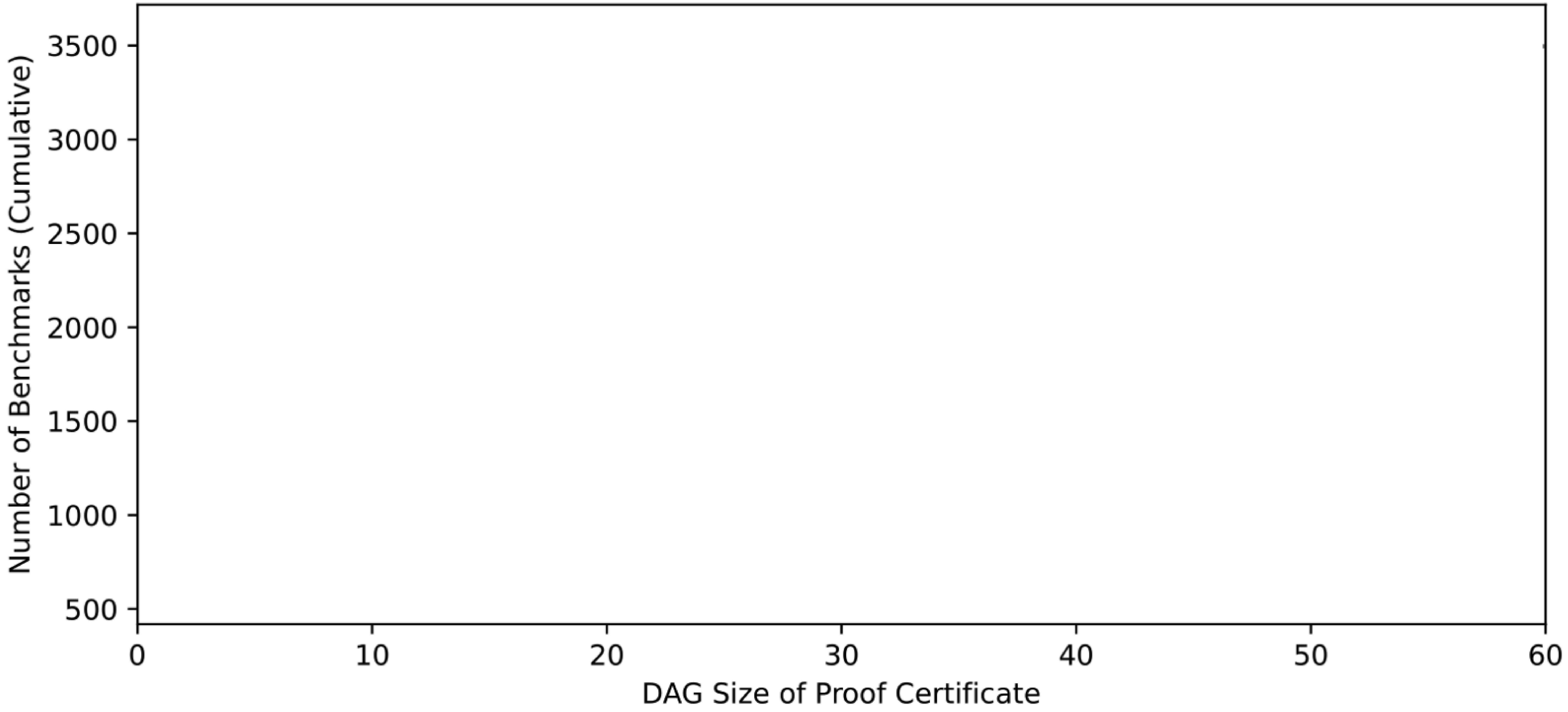


Putting it All Together

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

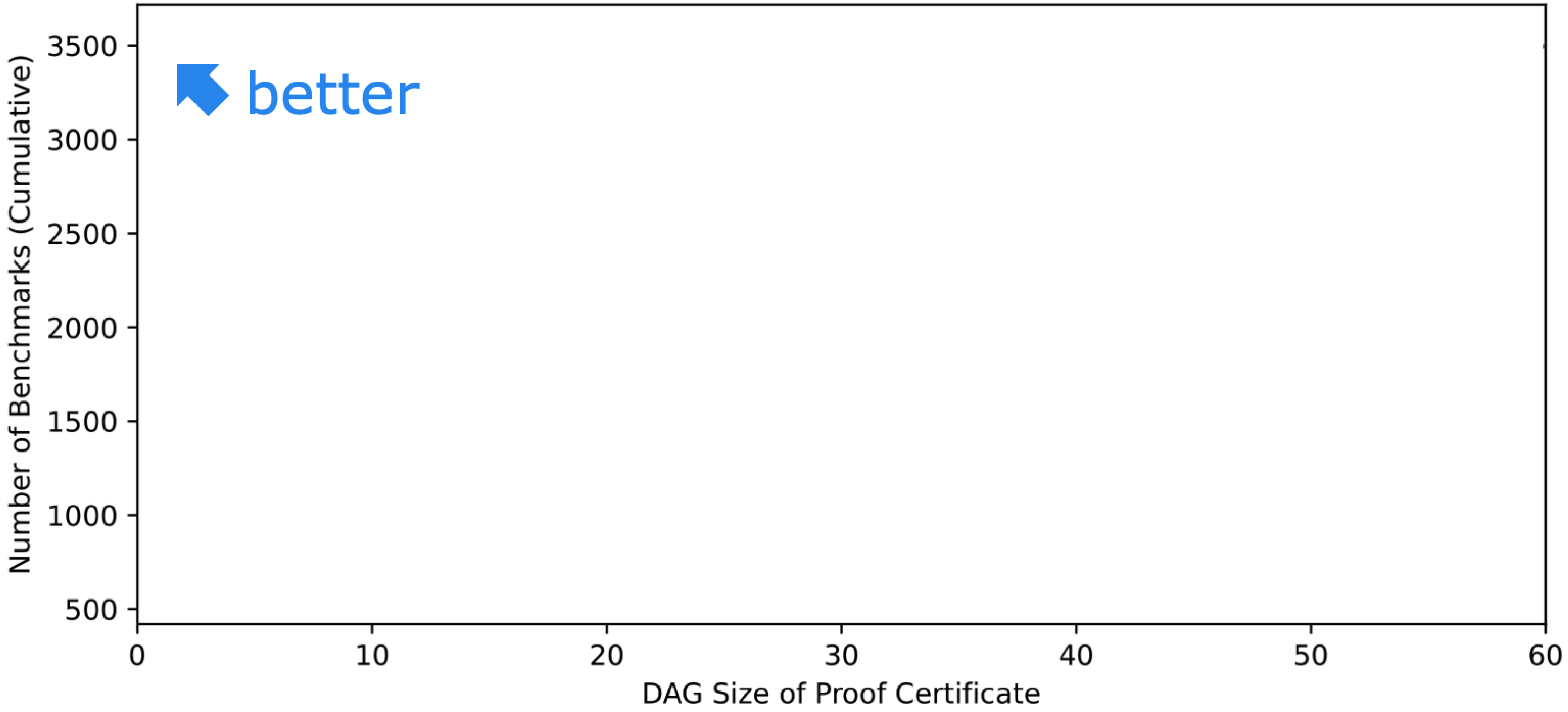


Results



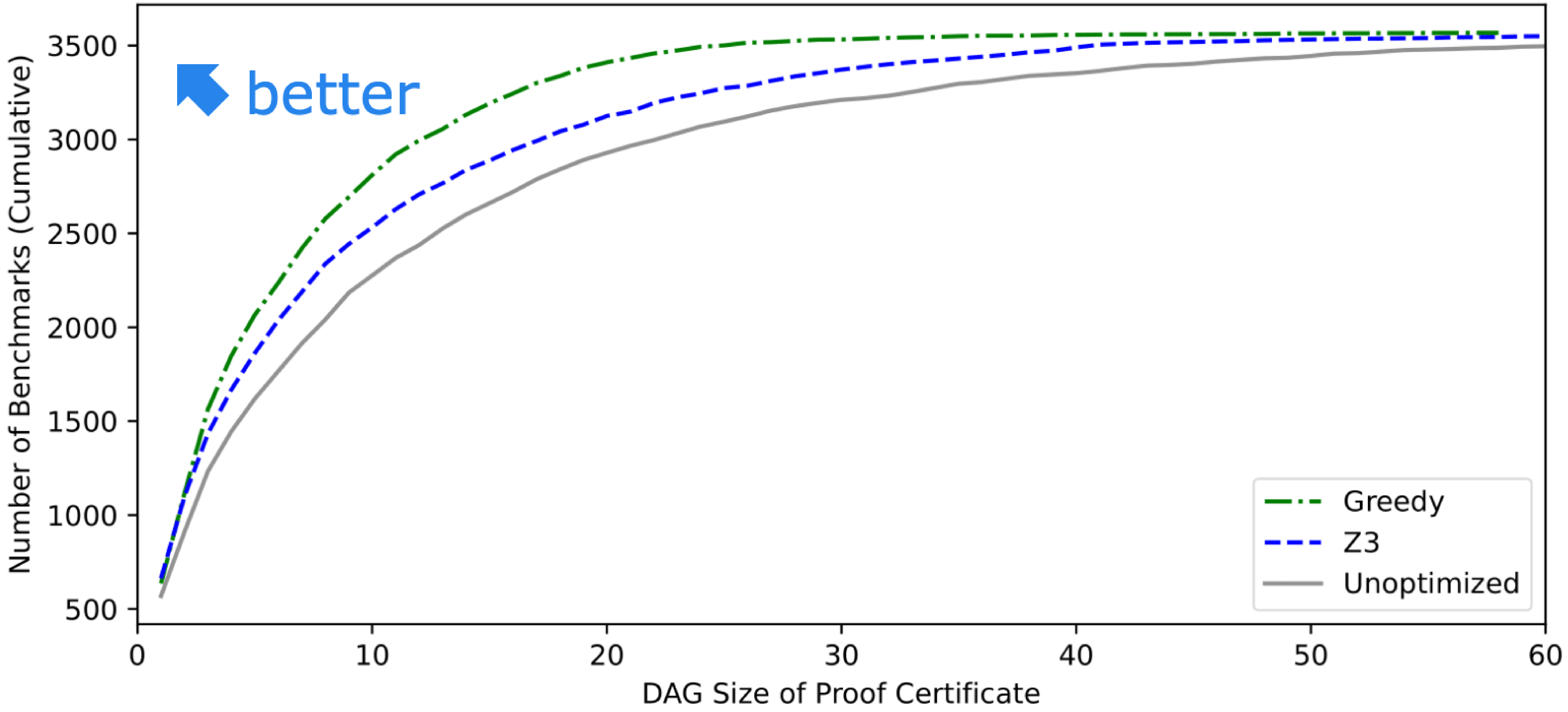
Number of Benchmarks (Cumulative)
3500
3000
2500
2000
1500
1000
500
0
10
20
30
40
50
60
DAG Size of Proof Certificate

Results



better
Number of Benchmarks (Cumulative)
3500
3000
2500
2000
1500
1000
500
0
10
20
30
40
50
60
DAG Size of Proof Certificate

Results



better
Number of Benchmarks (Cumulative)
3500
3000
2500
2000
1500
1000
500
0
10
20
30
40
50
60
DAG Size of Proof Certificate
Greedy
Z3
Unoptimized

Intel Case Study

Multi-operation circuit optimization and translation validation with egg

Intel Case Study

Multi-operation circuit optimization and translation validation with egg

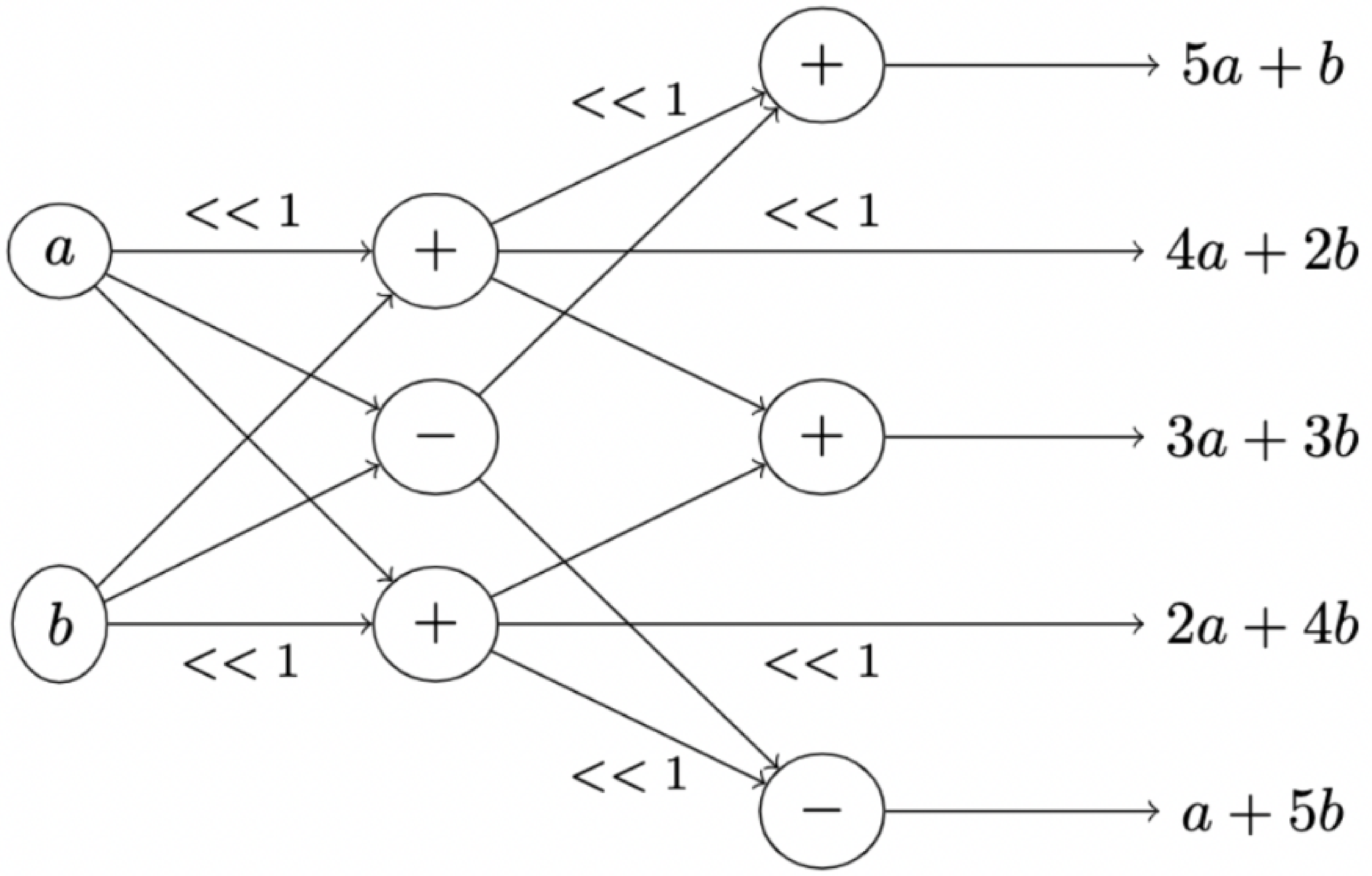
4.7 hours -> 2.3 hours

Intel Case Study

Multi-operation circuit optimization and translation validation with egg

4.7 hours -> 2.3 hours

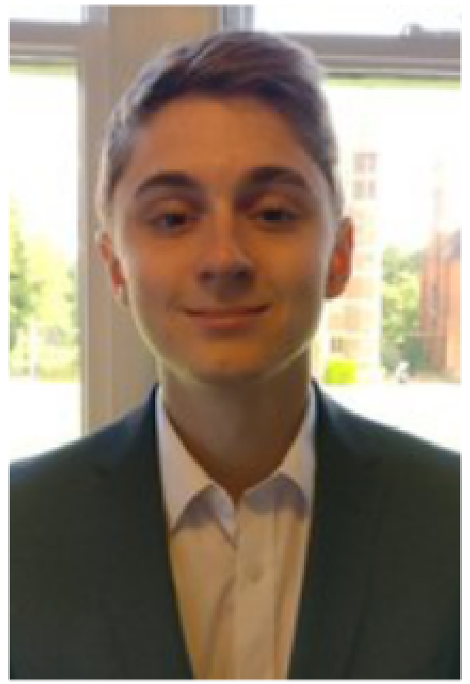
2 inputs



5 outputs

Team and Acknowledgments


Oliver Flatt


Samuel Coward


Max Willsey


Zachary Tatlock


Pavel Panchekha

Special thanks to:

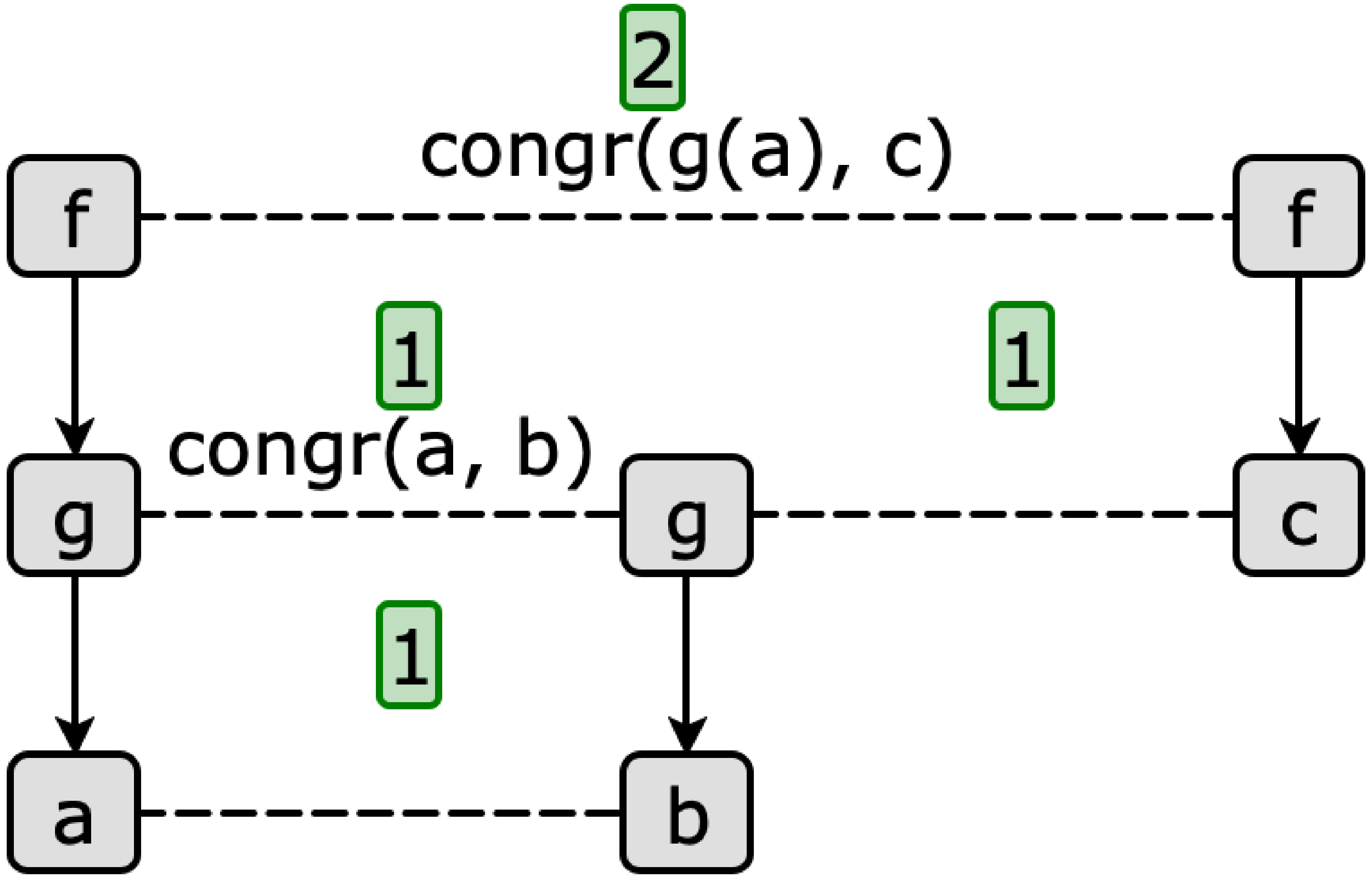
Theo Drane (Intel)

George A. Constantinides (Imperial College)

Leonardo de Moura (Microsoft)

Questions?

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



oflatt@cs.washington.edu