Concurrent Cube-and-Conquer

Peter van der Tak[1]
Marijn Heule[1,2]
Armin Biere[3]

[1]Delft University of Technology, The Netherlands

[2]University of Texas at Austin, United States

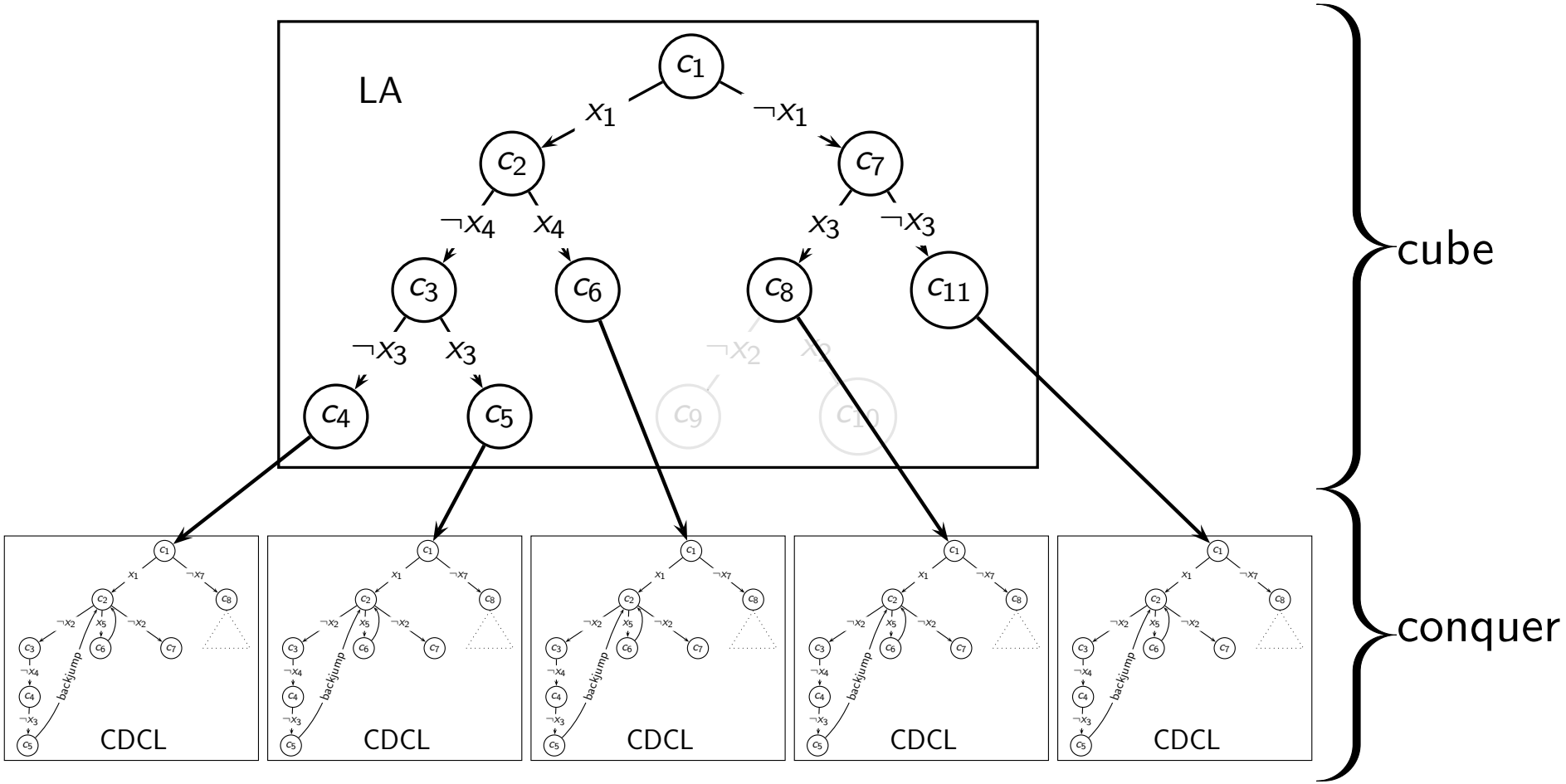
[3]Johannes Kepler University Linz, Austria

June 16, 2012

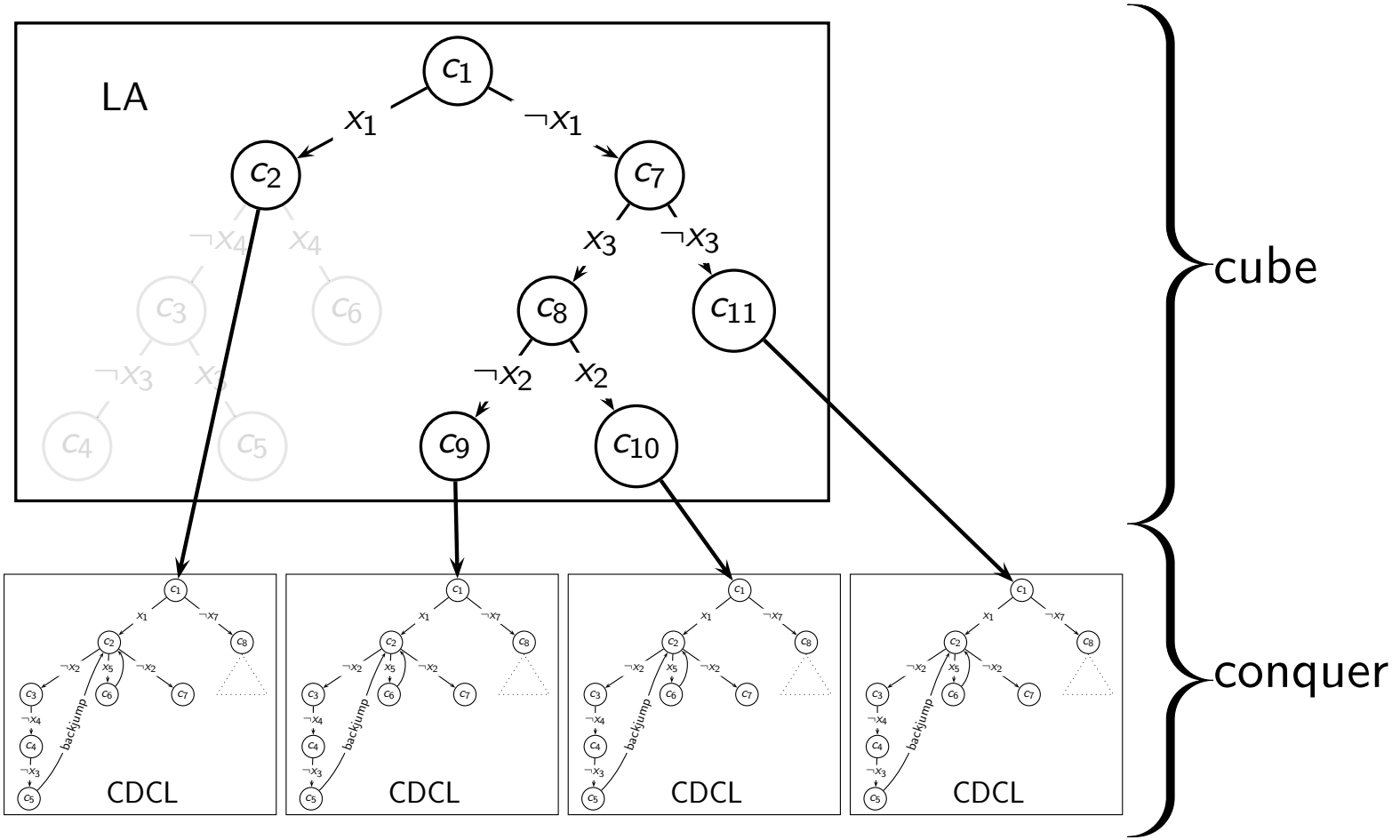
Introduction

- lookahead (LA)
  - recursively split instance: binary search tree
  - good for small, hard problems
- conflict-driven clause learning (CDCL)
  - learn implied clauses: less systematic search
  - best for large, "easy" industrial instances
- cube-and-conquer (CC)
  - partition using LA into thousands or millions of subproblems
  - solve subproblems in parallel using CDCL

Cube-and-conquer



Cube-and-conquer



Old CC cutoff heuristic

$d(c_{\mathrm{id}}) := |\varphi_{\mathrm{dec}}|^2 \cdot (|\varphi_{\mathrm{dec}}| + |\varphi_{\mathrm{imp}}|)$

$d(c_{\mathrm{id}}) >$ threshold $\rightarrow$ cut off

Dynamic threshold:

- LA refutes cube $\rightarrow$ decrease
- increase gradually

Motivation

Limitations of cube-and-conquer (CC):

- partitioning not ideal
- lookahead not always effective

Proposed solutions:

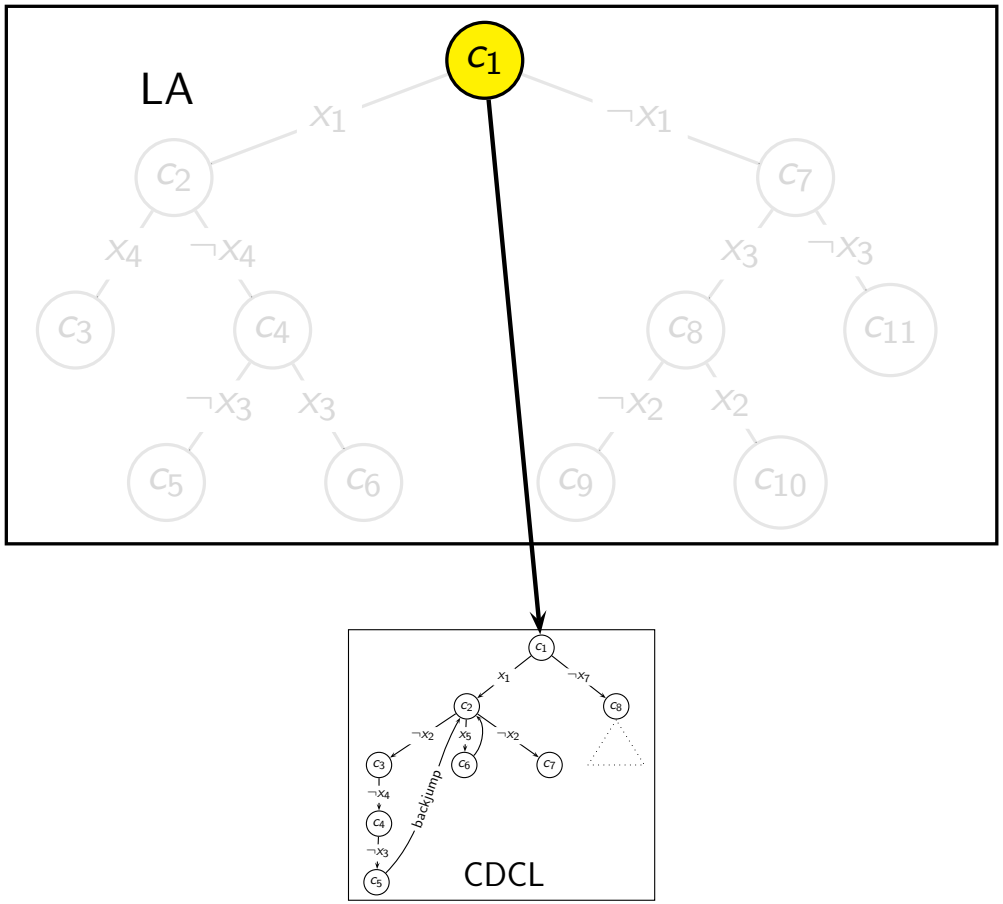
- run CDCL and LA concurrently in partitioning phase
- predict unsuitable instances

Concurrent cube-and-conquer

Solve by adding CDCL to cube phase:

- run LA and CDCL concurrently
- add decisions by LA as assumptions to CDCL
- use existing solvers (`March_rw` and `MiniSAT 2.2`)

Concurrent cube-and-conquer



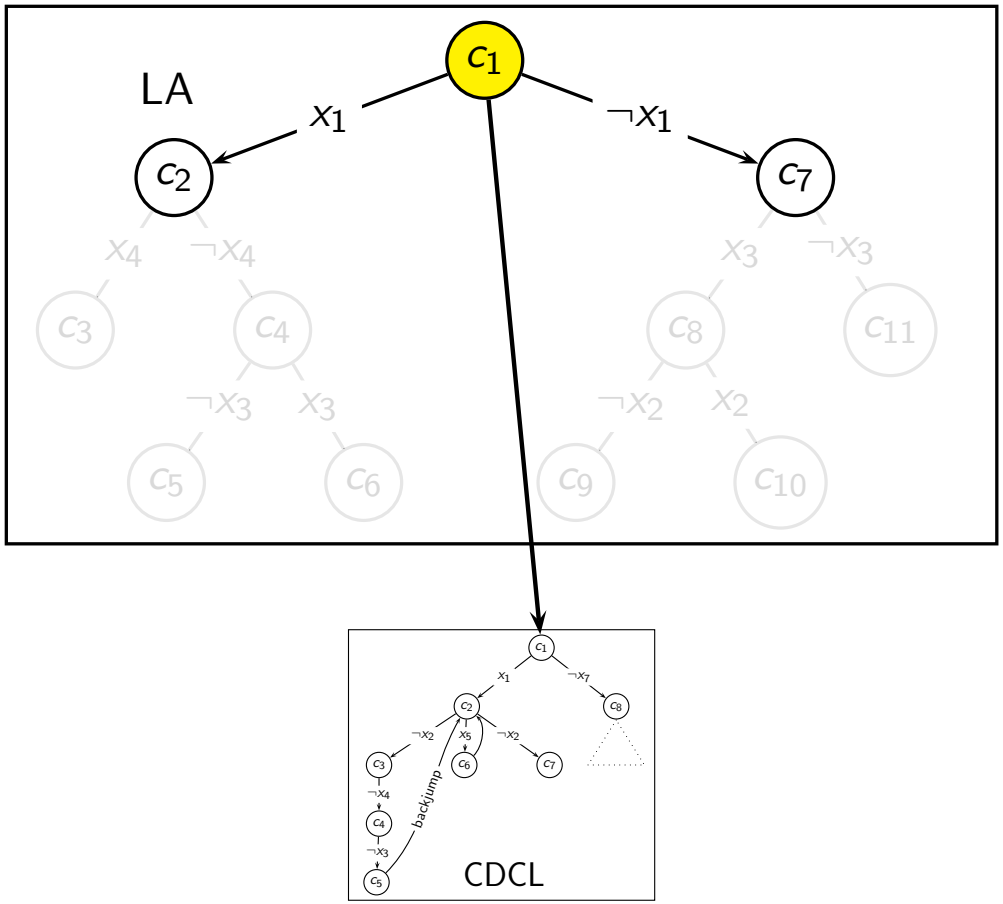
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



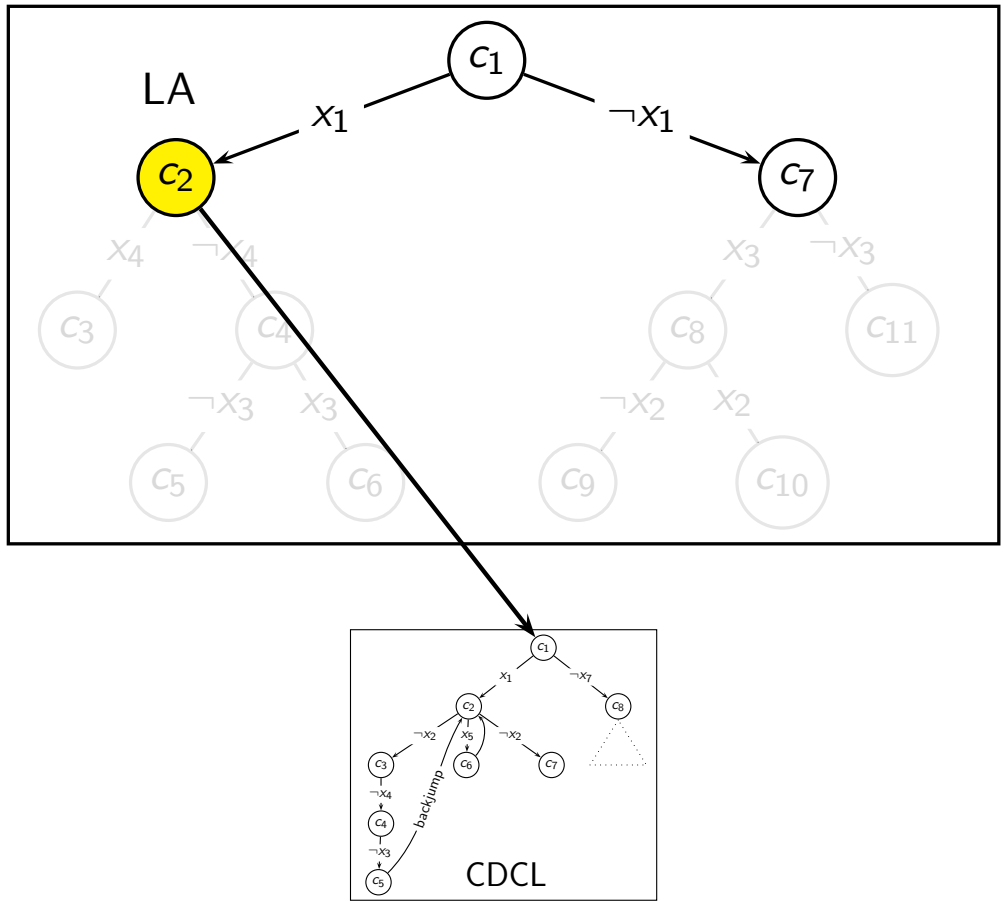
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_1, c_2, 0 \rangle$

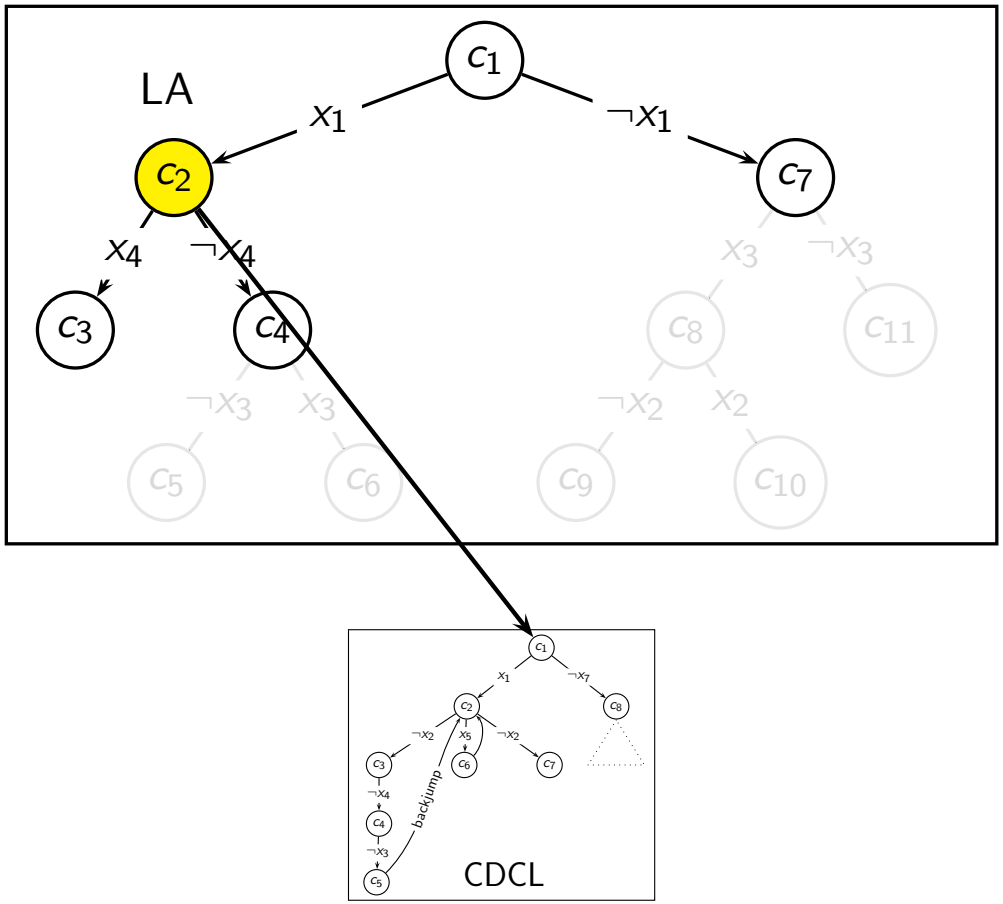
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



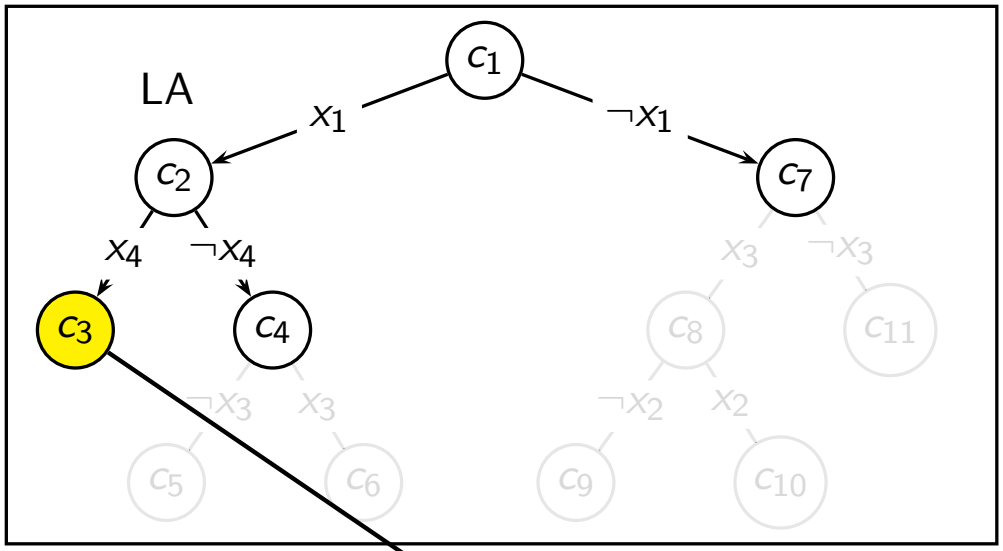
c_x cube

c_x current

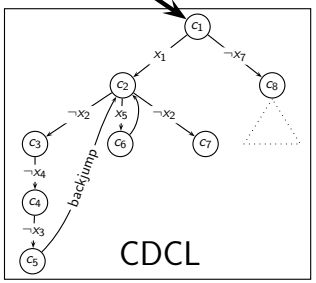
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_4, c_3, 1 \rangle$



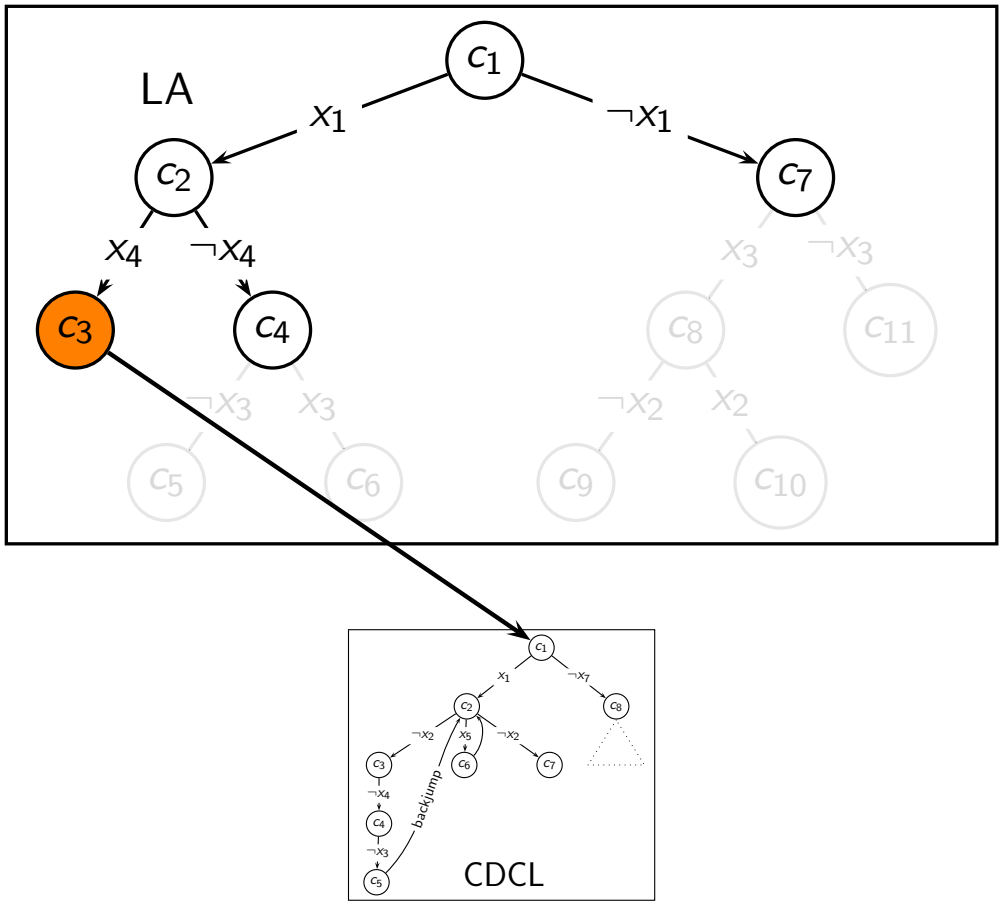
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



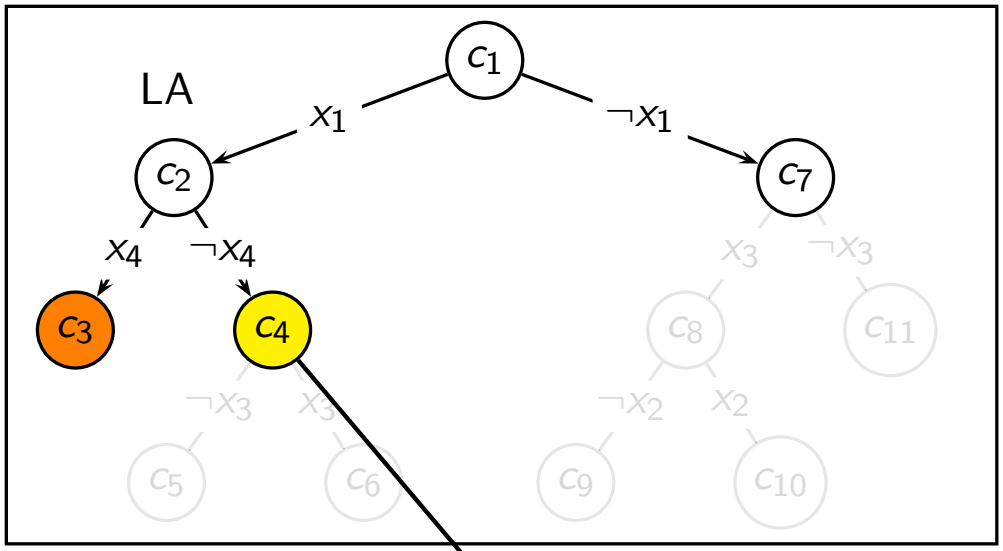
c_x cube

c_x current

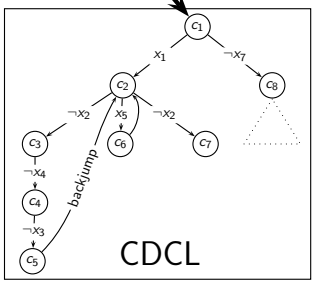
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_4, c_4, 1 \rangle$



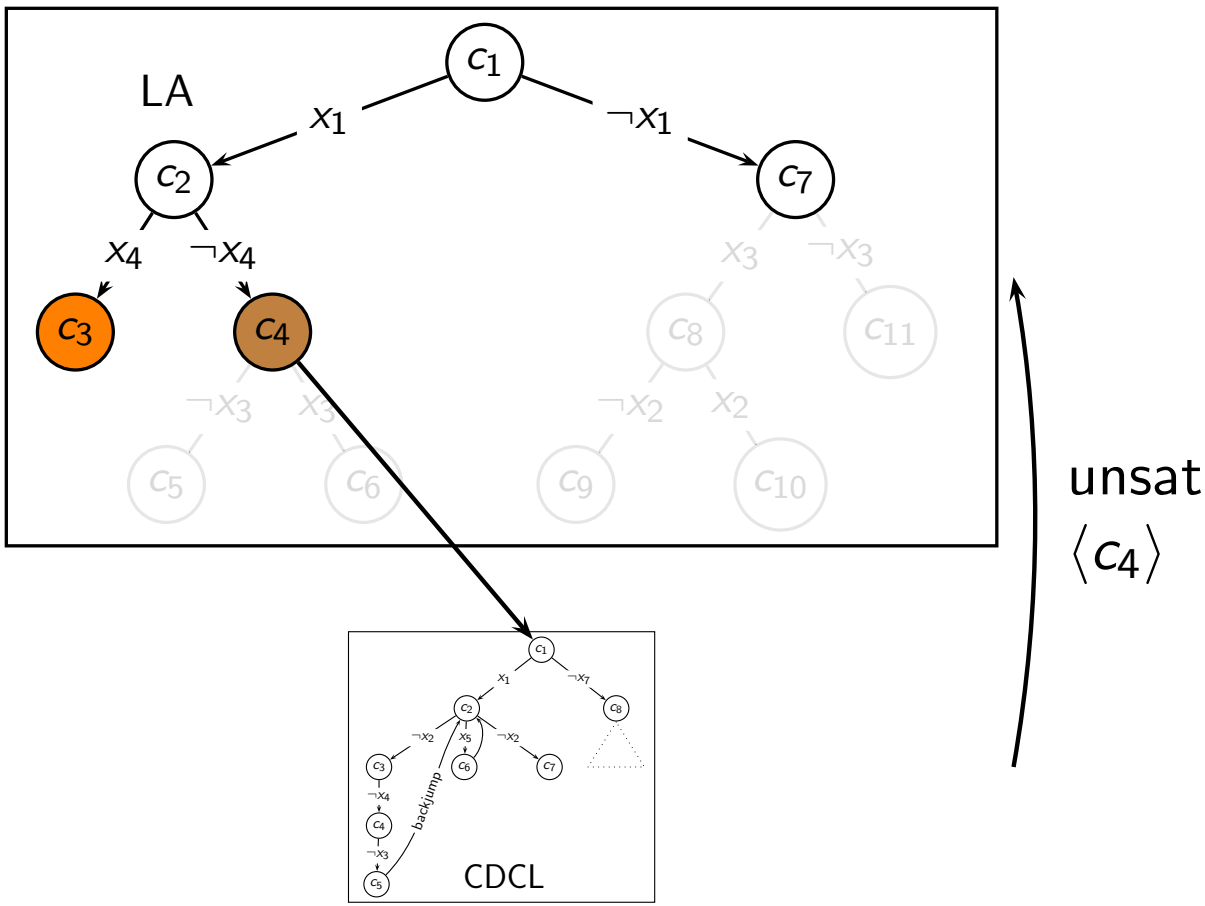
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



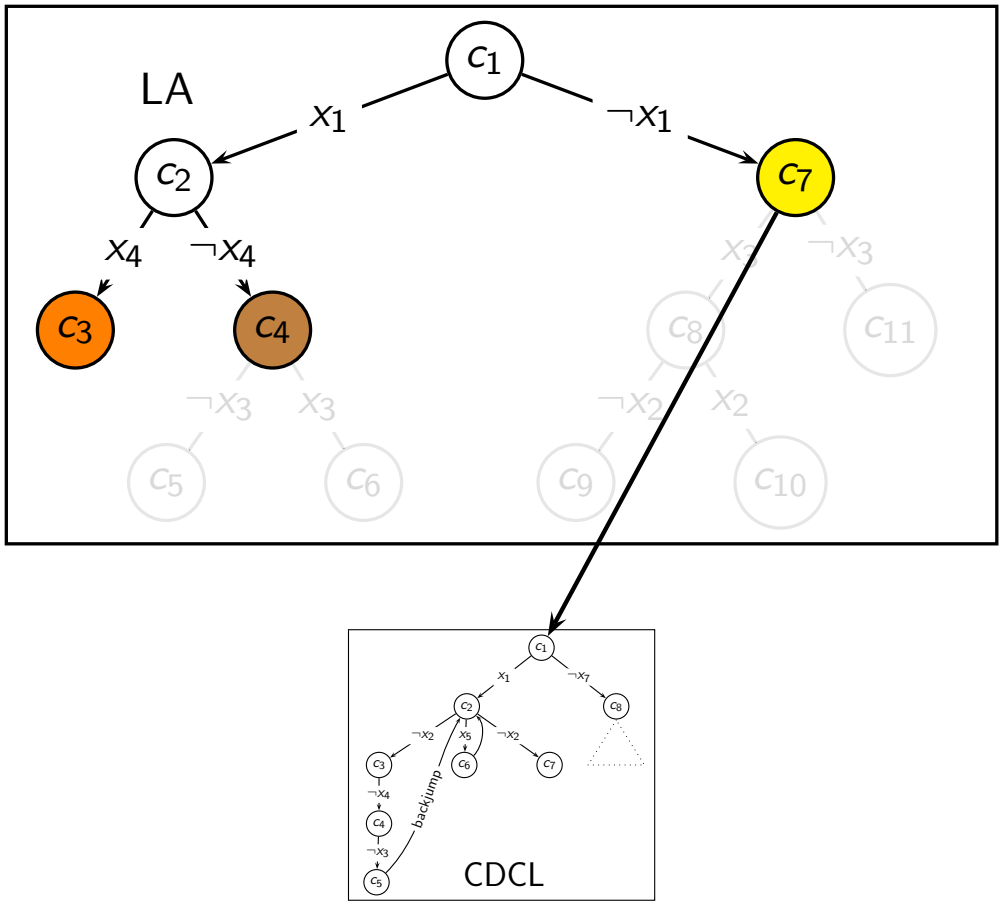
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

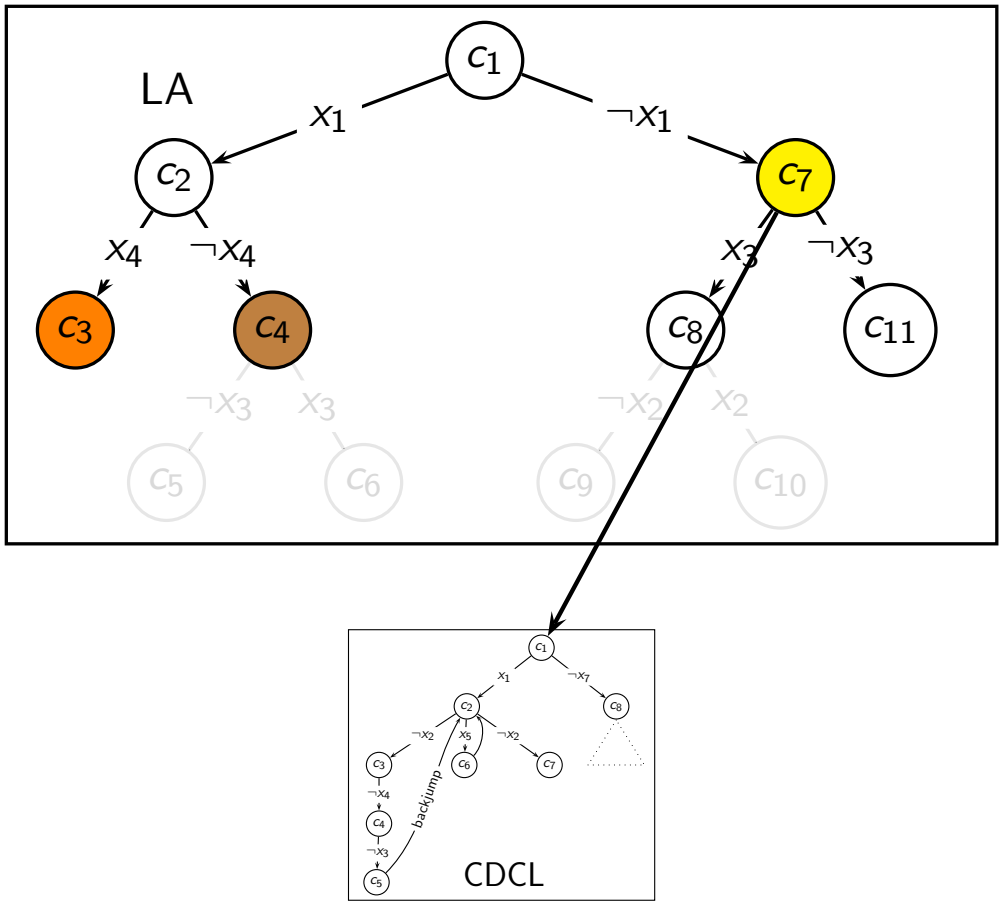
Concurrent cube-and-conquer



decision
$\langle \neg x_1, c_7, 0 \rangle$

- c_x cube
- c_x current
- c_x refuted by LA
- c_x refuted by CDCL

Concurrent cube-and-conquer



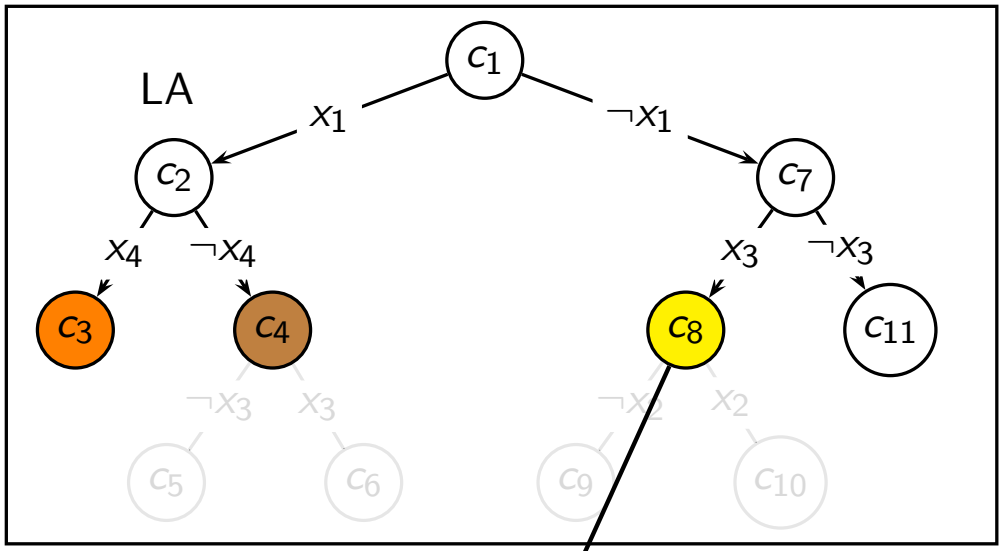
c_x cube

c_x current

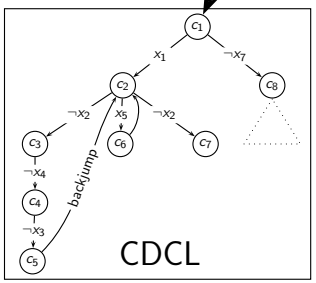
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_3, c_8, 1 \rangle$



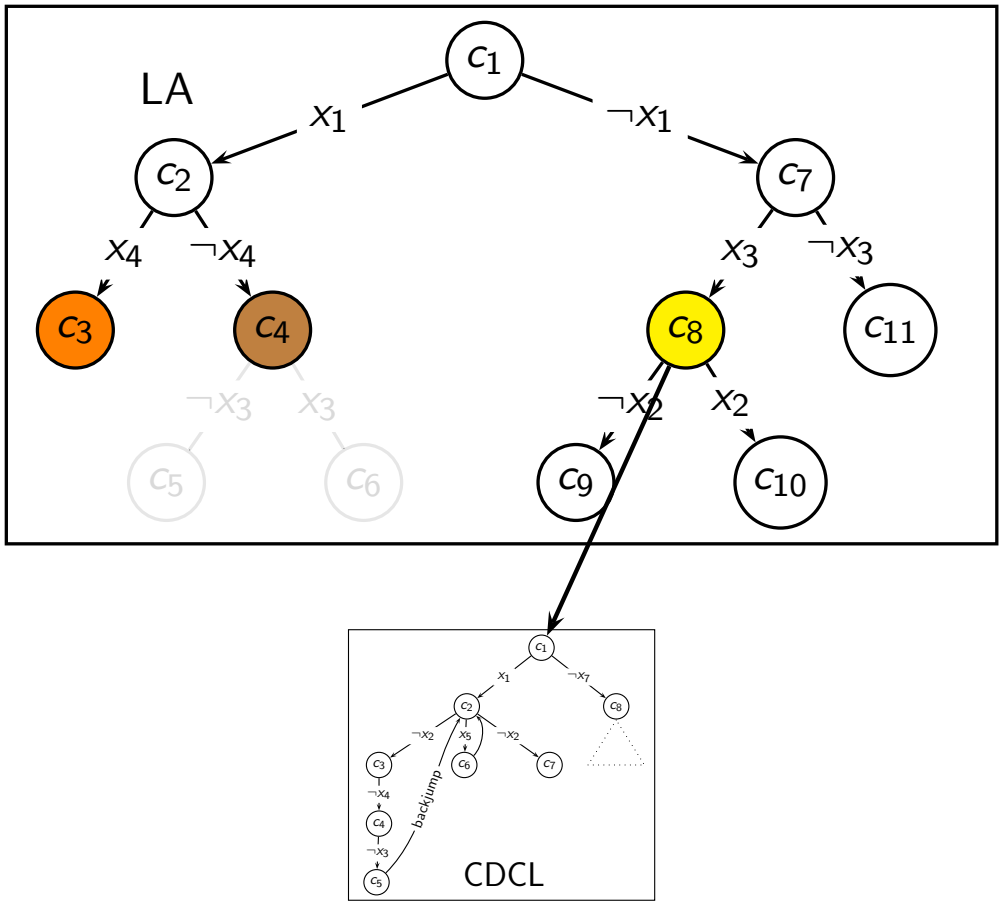
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



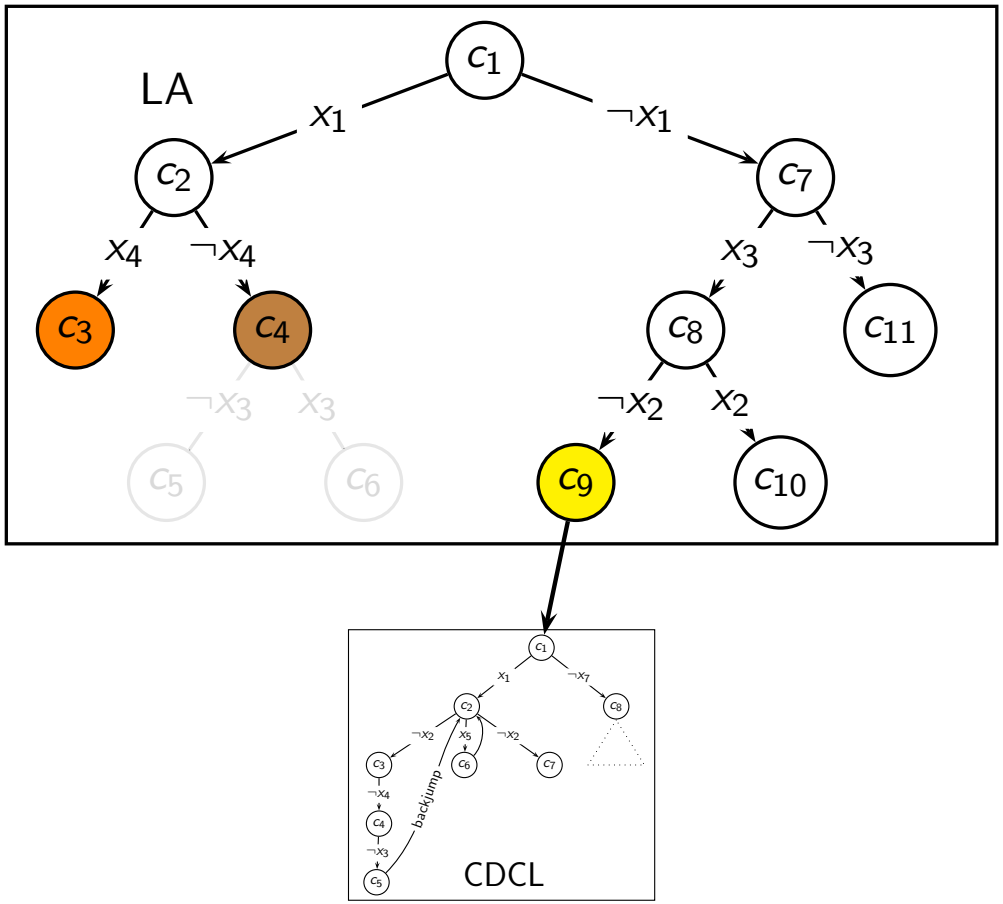
c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision

$\langle \neg x_2, c_9, 2 \rangle$

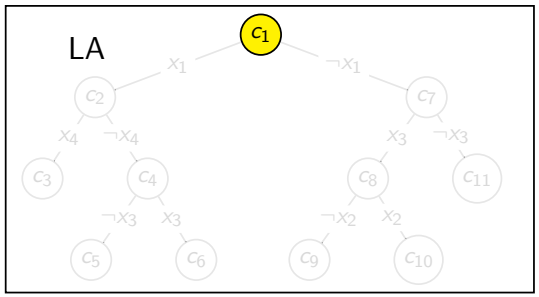
c_x cube

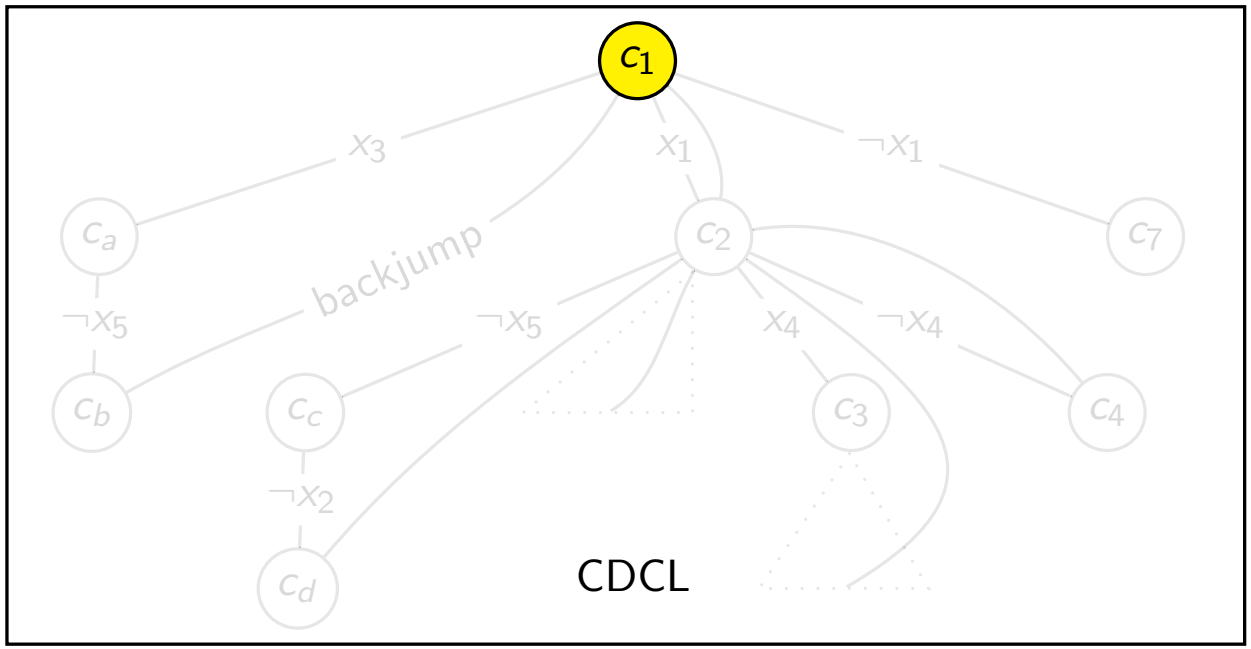
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





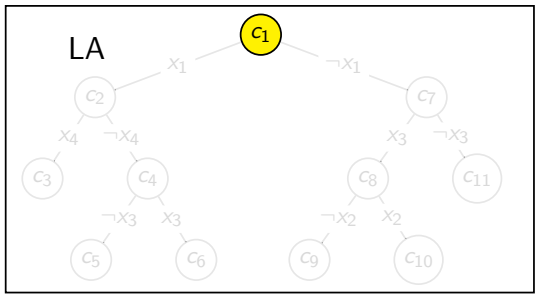
c_x cube

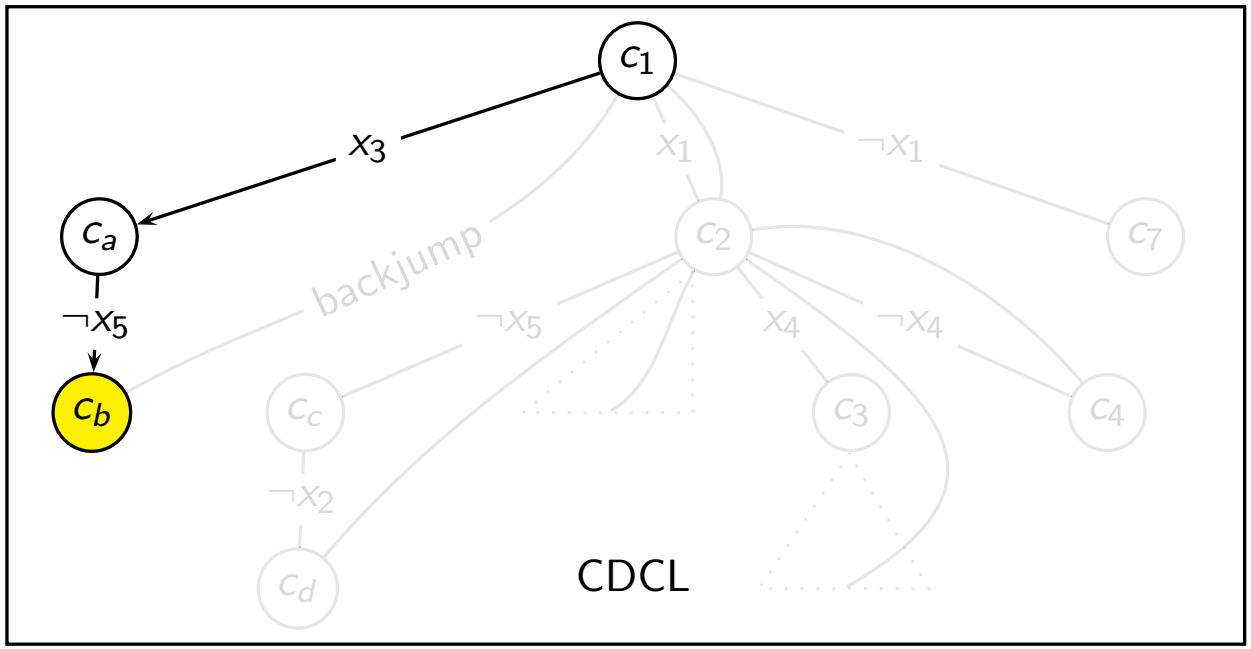
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





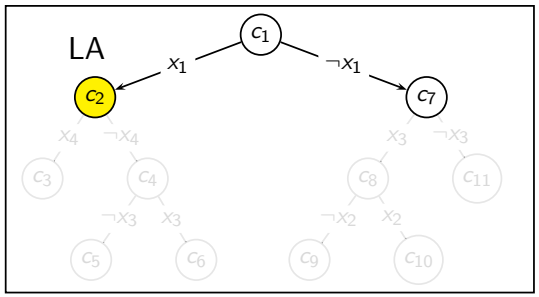
c_x cube

c_x current

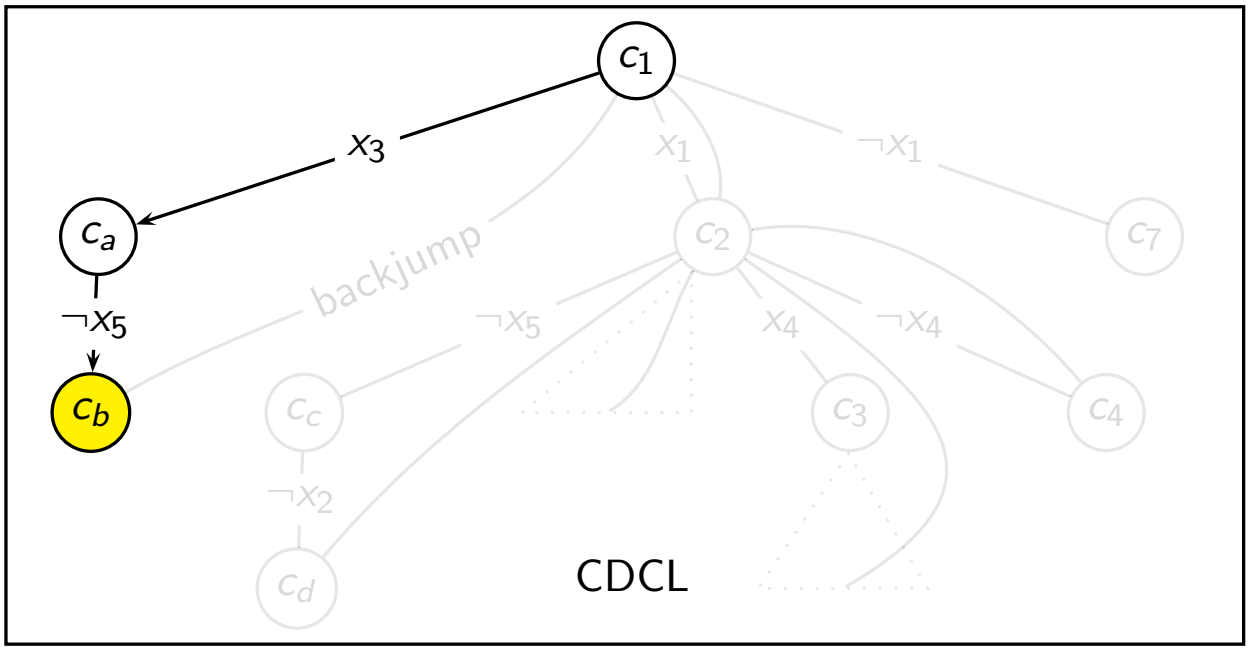
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_1, c_2, 0 \rangle$



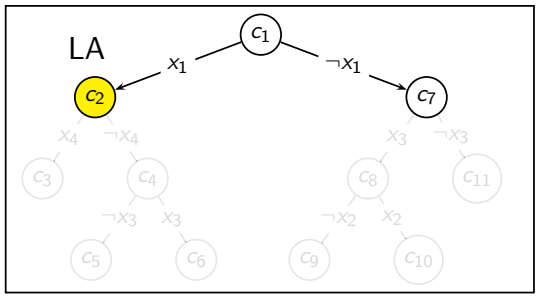
c_x cube

c_x current

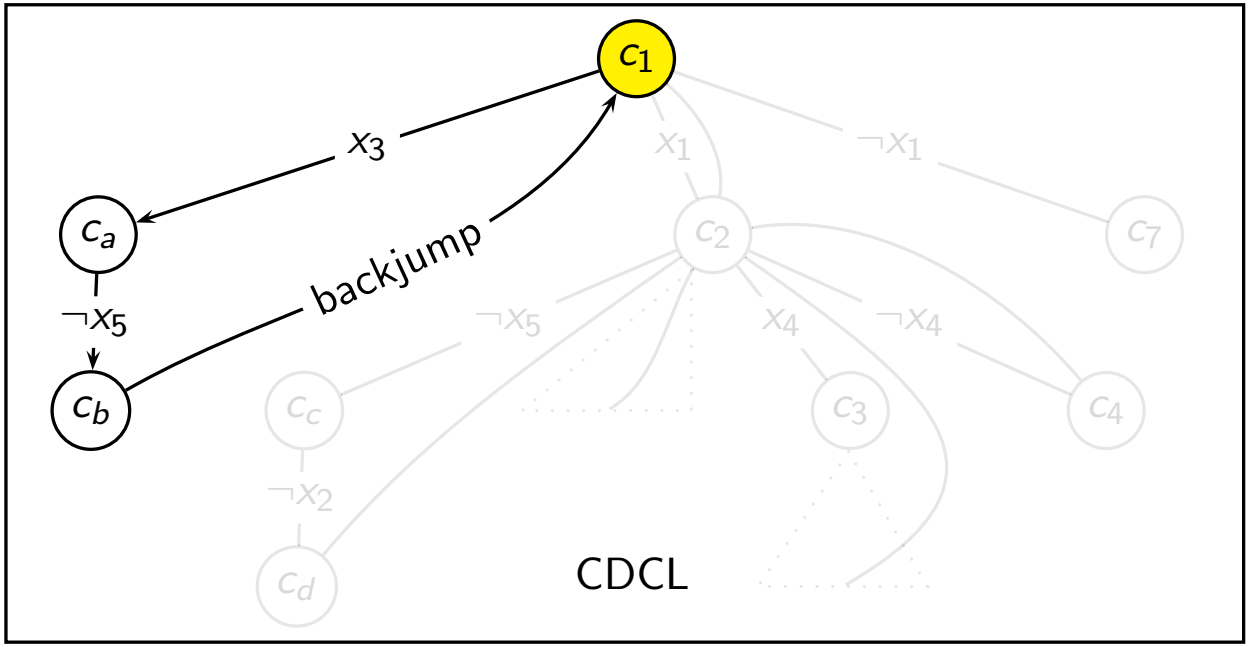
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_1, c_2, 0 \rangle$



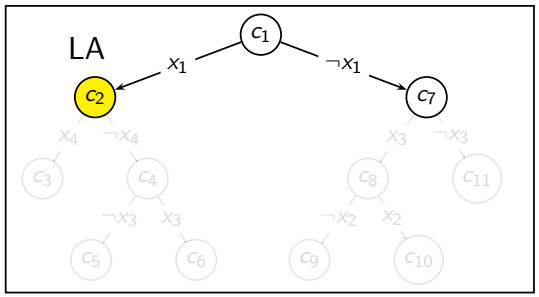
c_x cube

c_x current

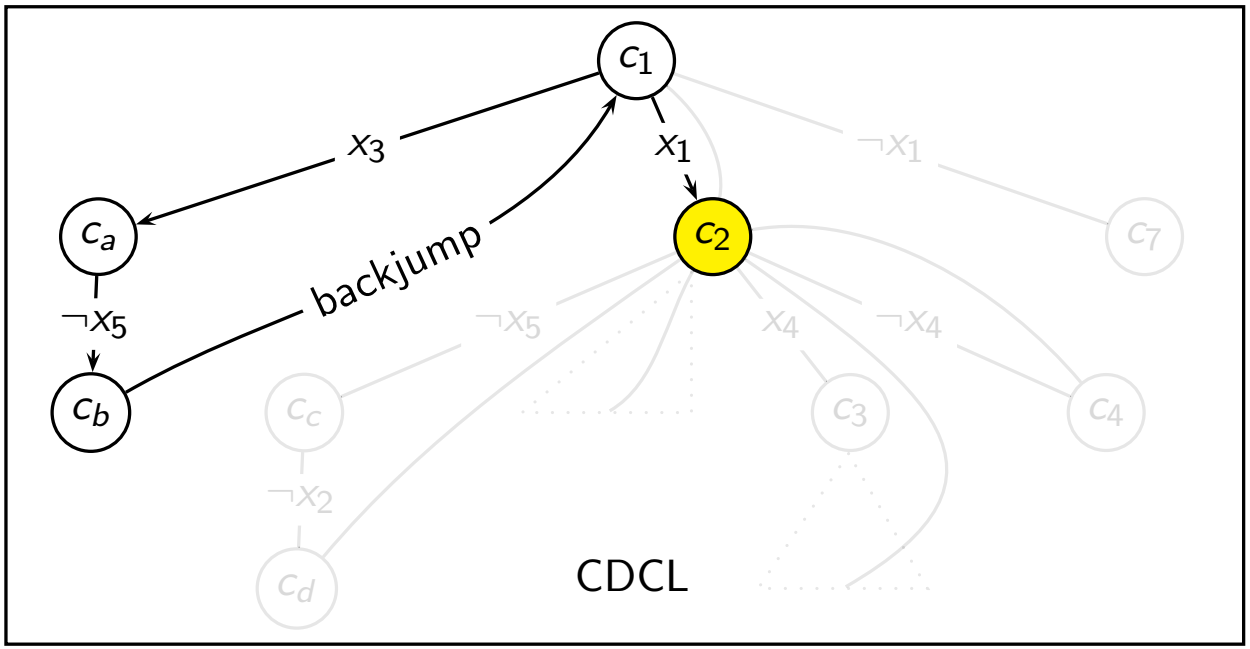
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_1, c_2, 0 \rangle$



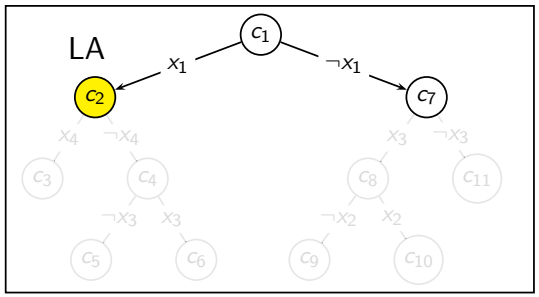
c_x cube

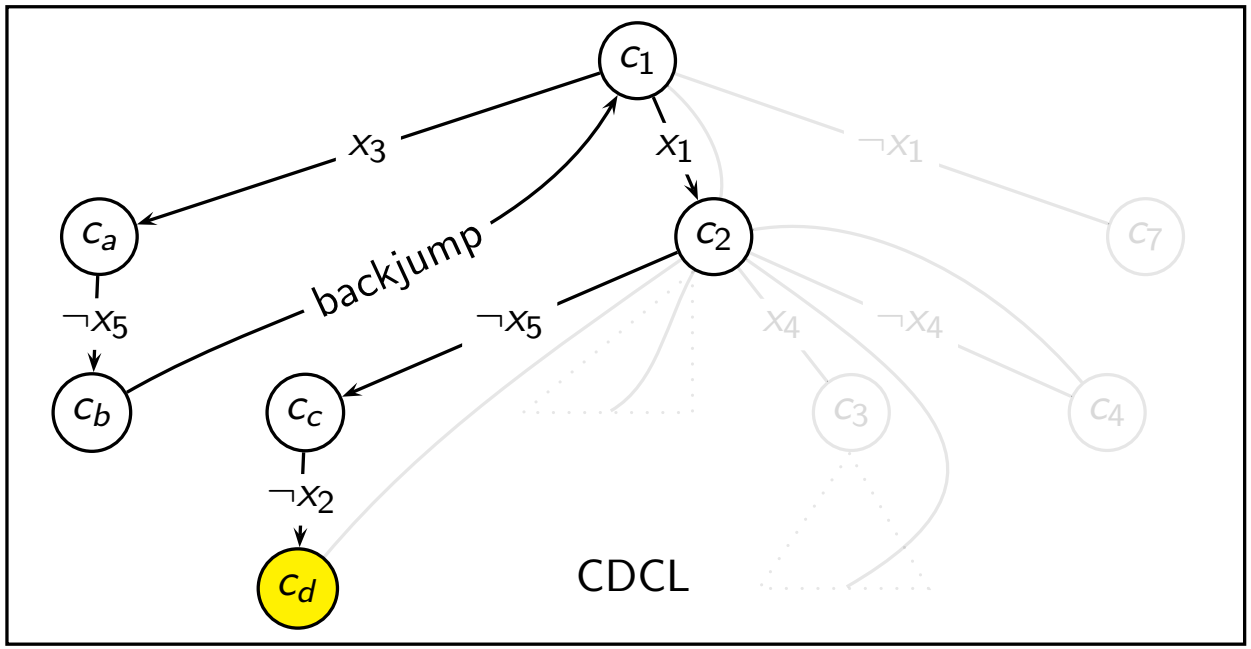
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





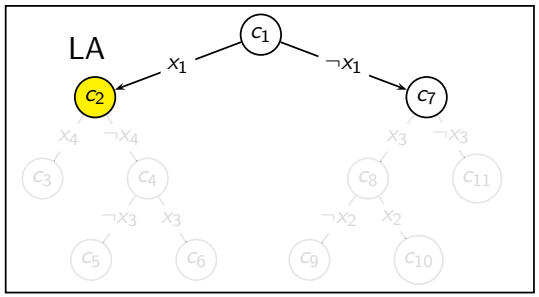
c_x cube

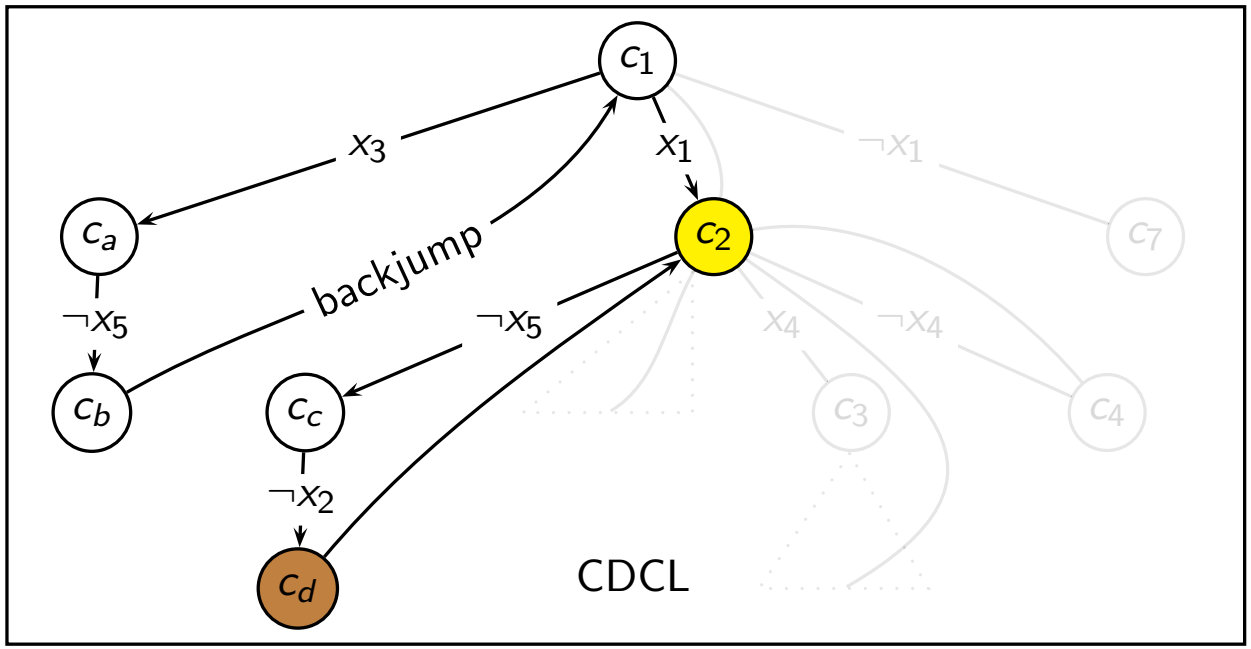
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





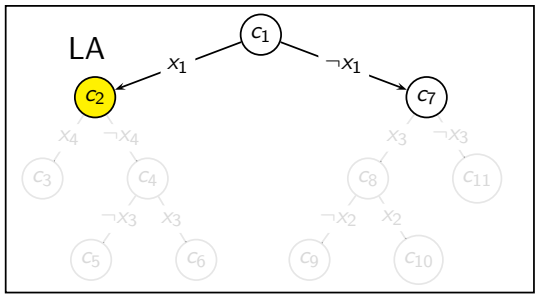
c_x cube

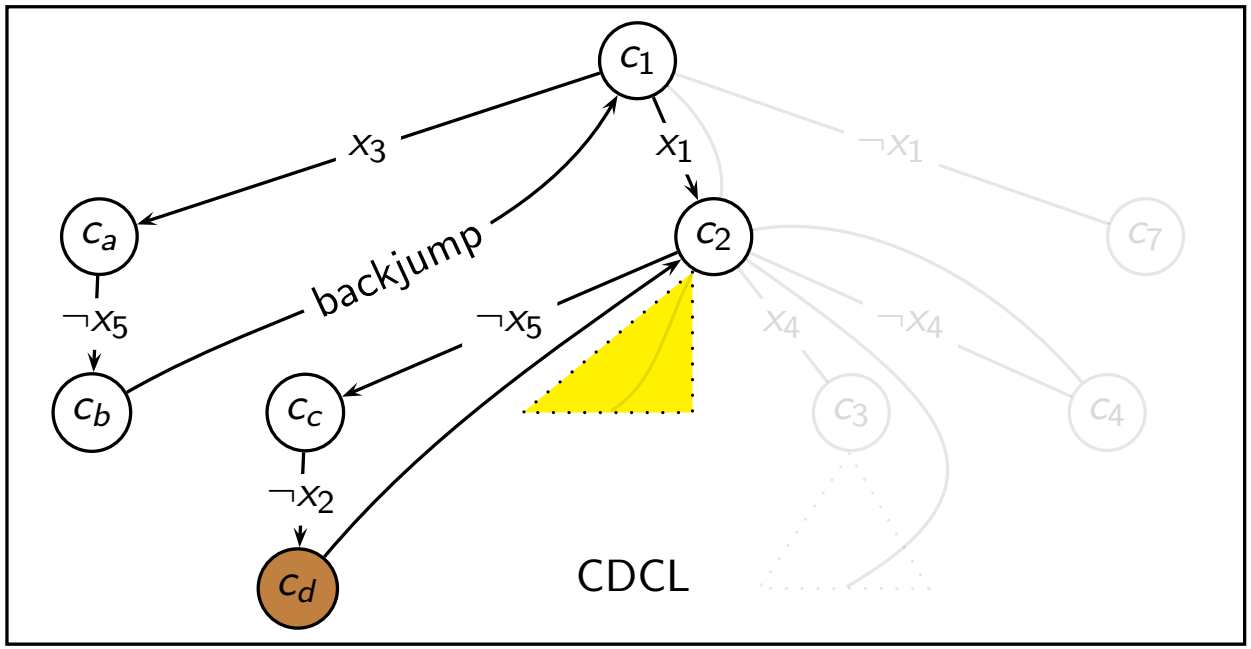
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





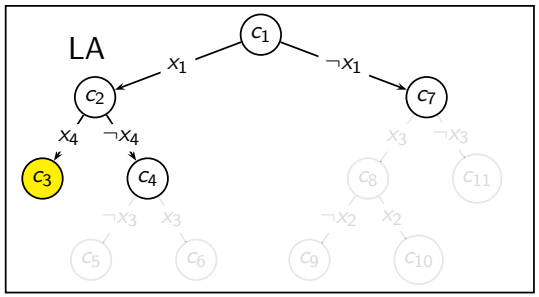
c_x cube

c_x current

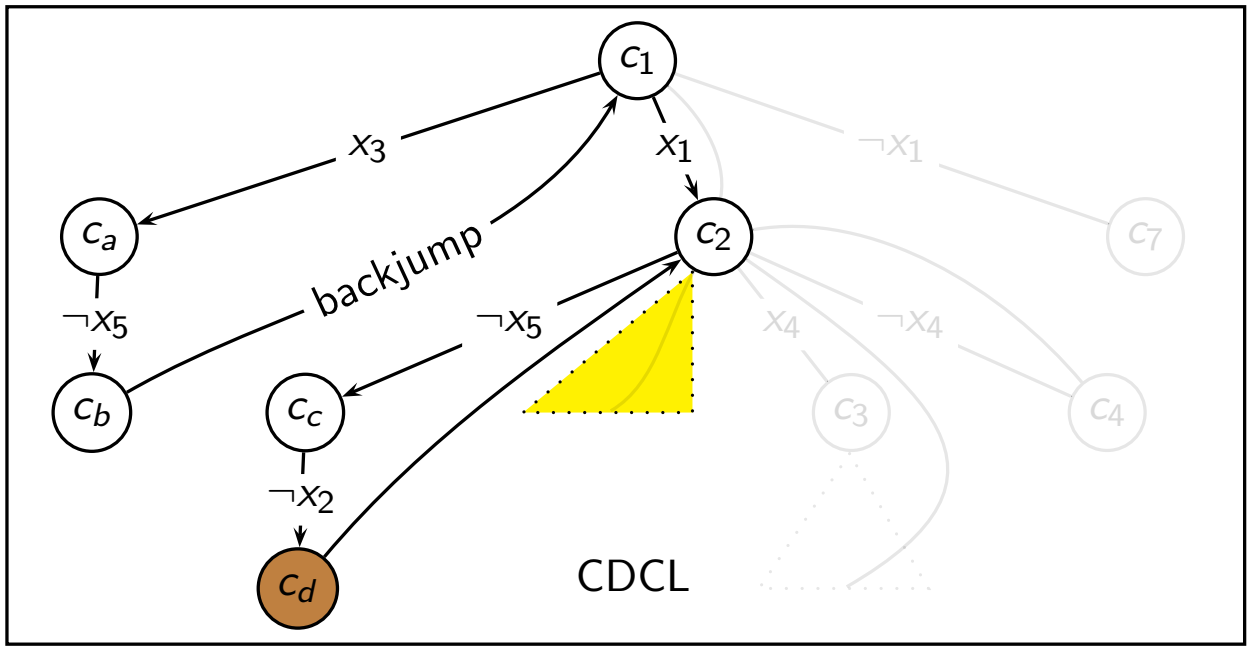
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_4, c_3, 1 \rangle$



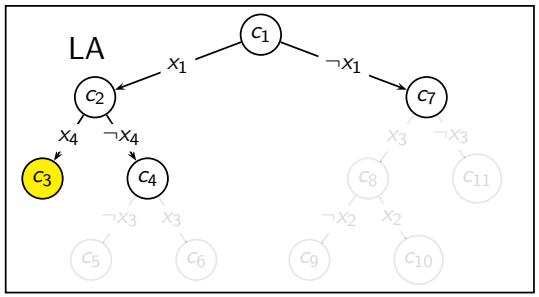
c_x cube

c_x current

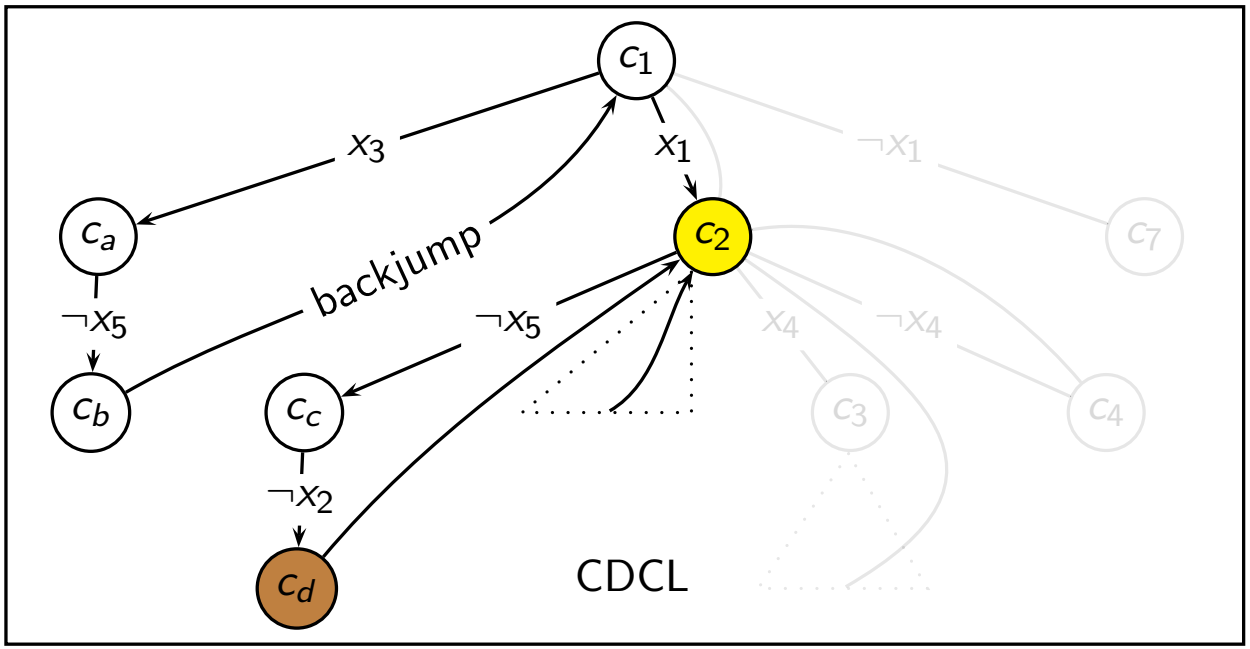
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_4, c_3, 1 \rangle$



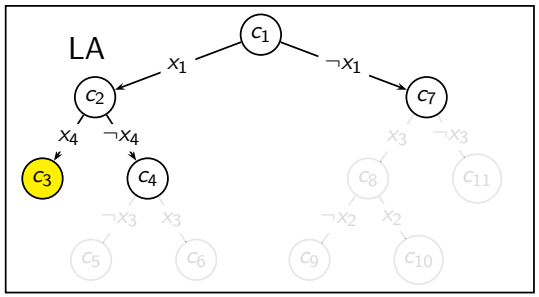
c_x cube

c_x current

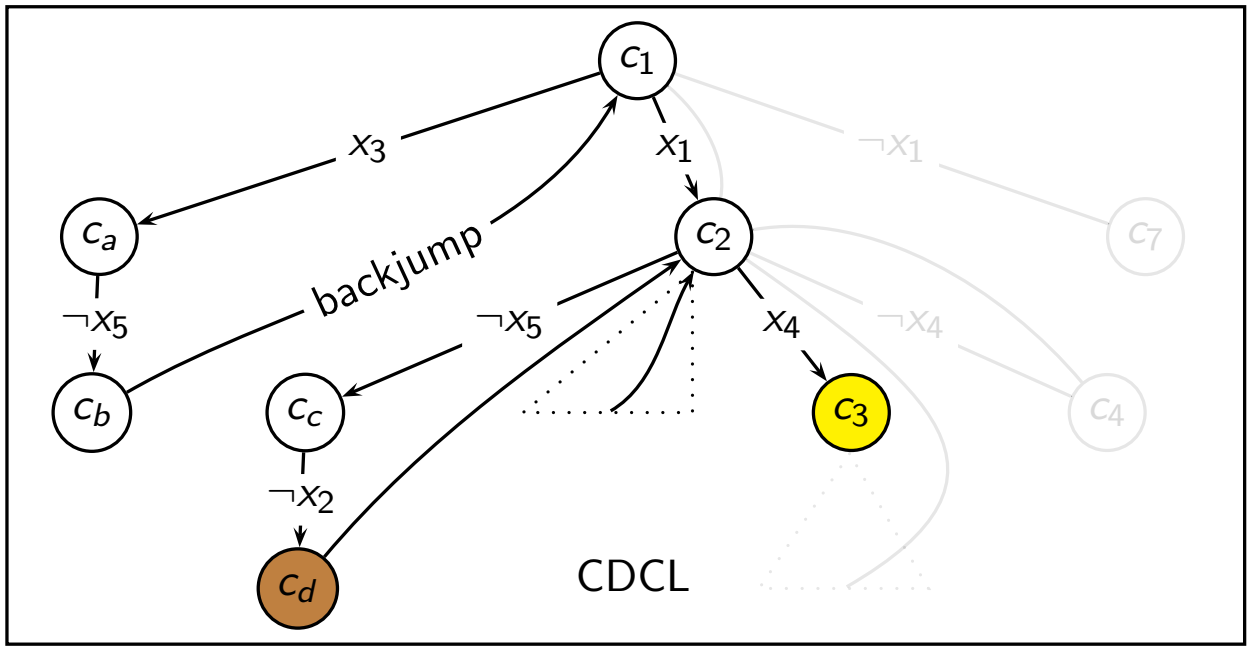
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle x_4, c_3, 1 \rangle$



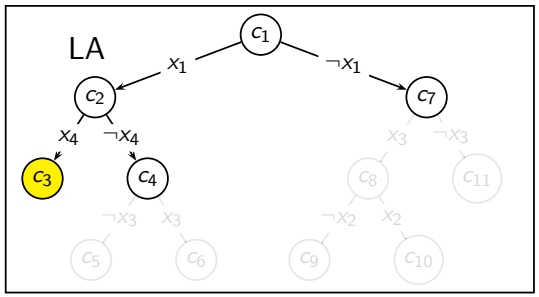
c_x cube

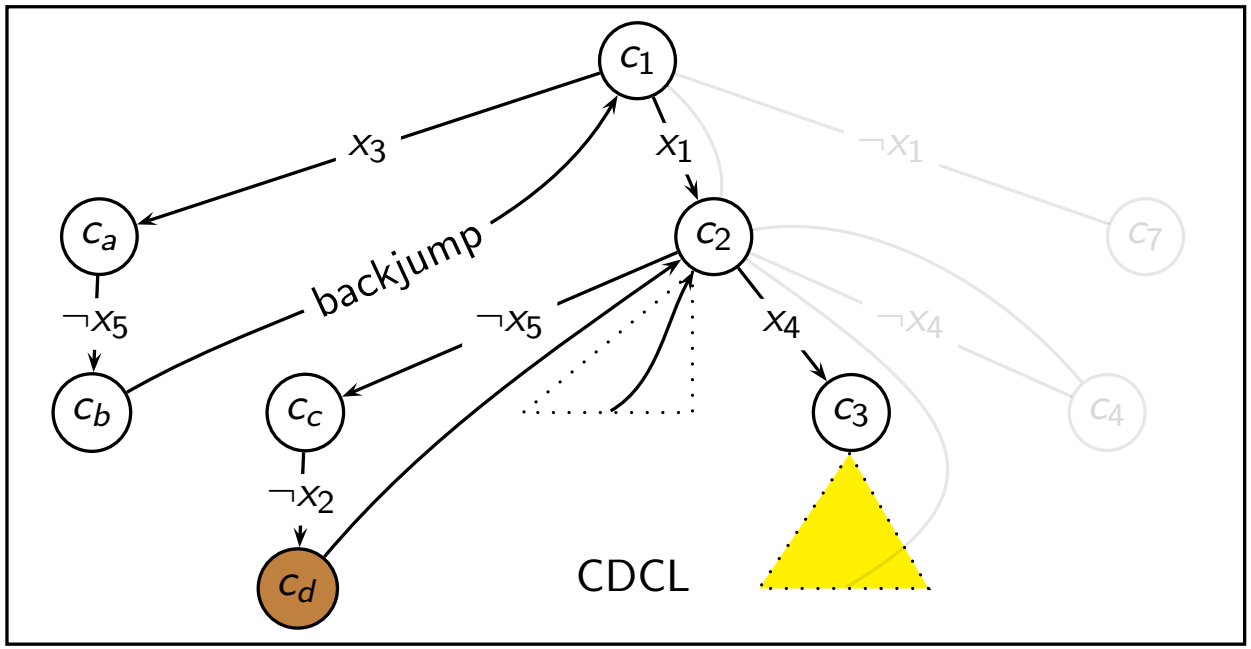
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





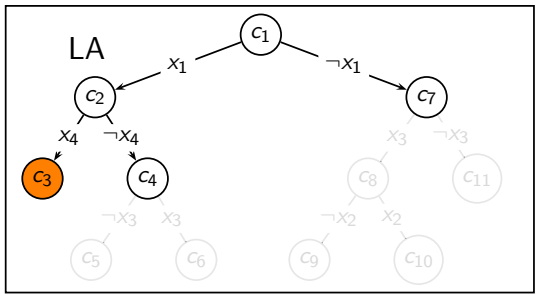
c_x cube

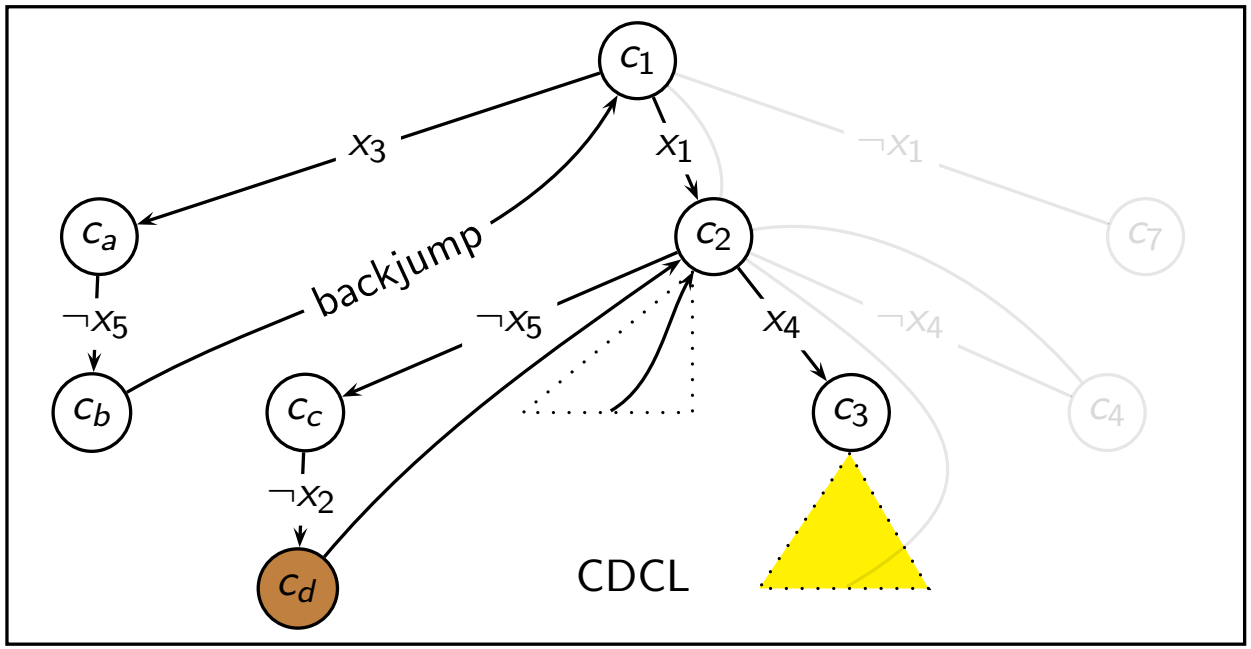
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





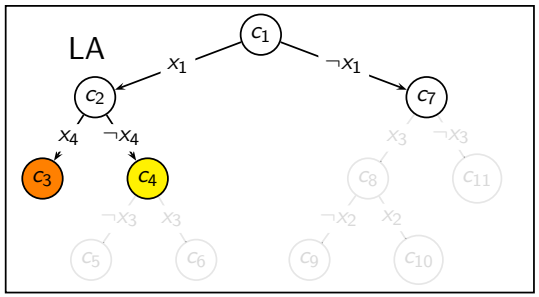
c_x cube

c_x current

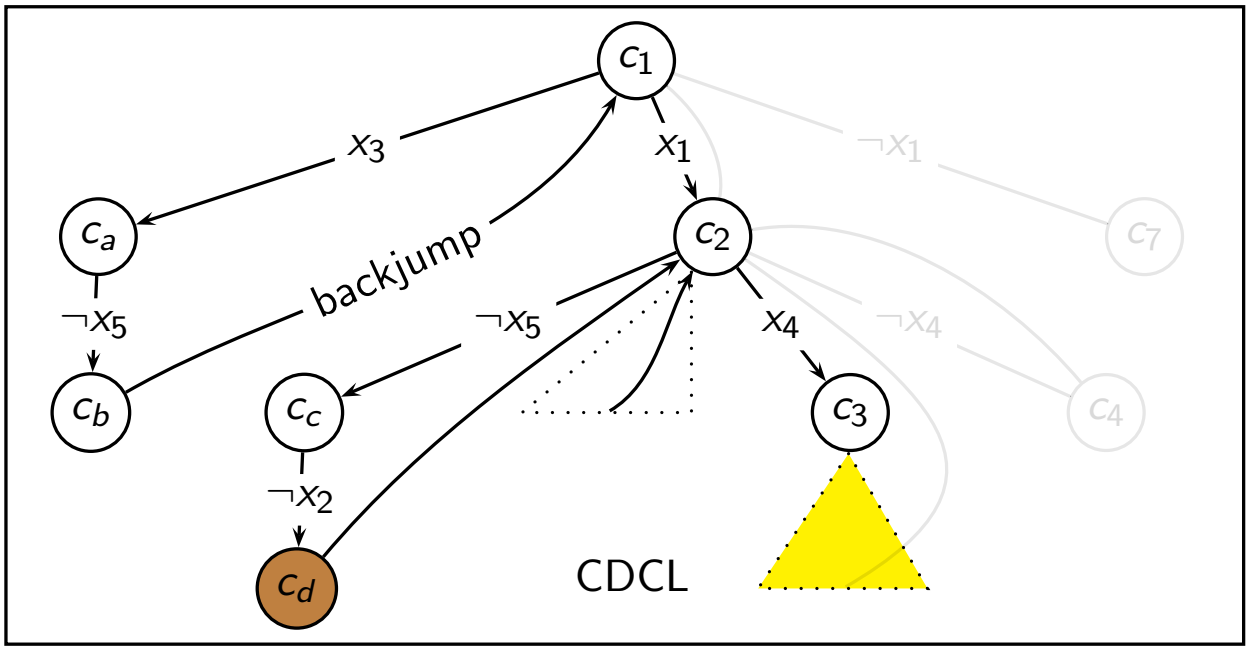
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_4, c_4, 1 \rangle$



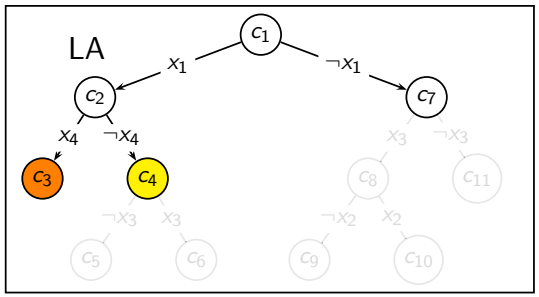
c_x cube

c_x current

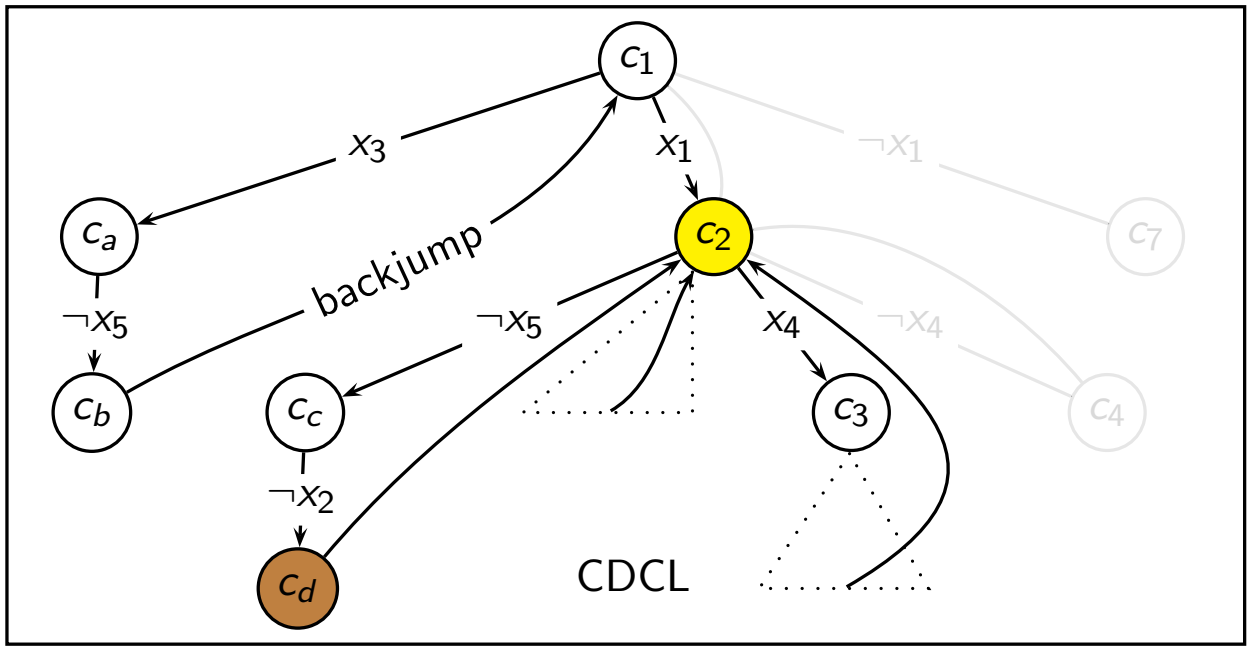
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_4, c_4, 1 \rangle$



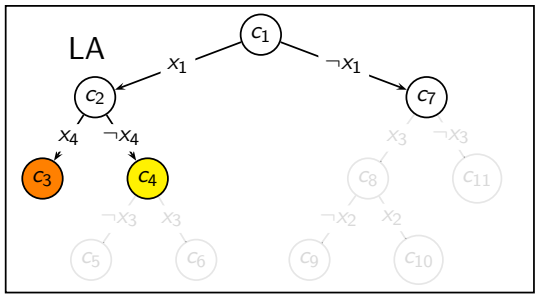
c_x cube

c_x current

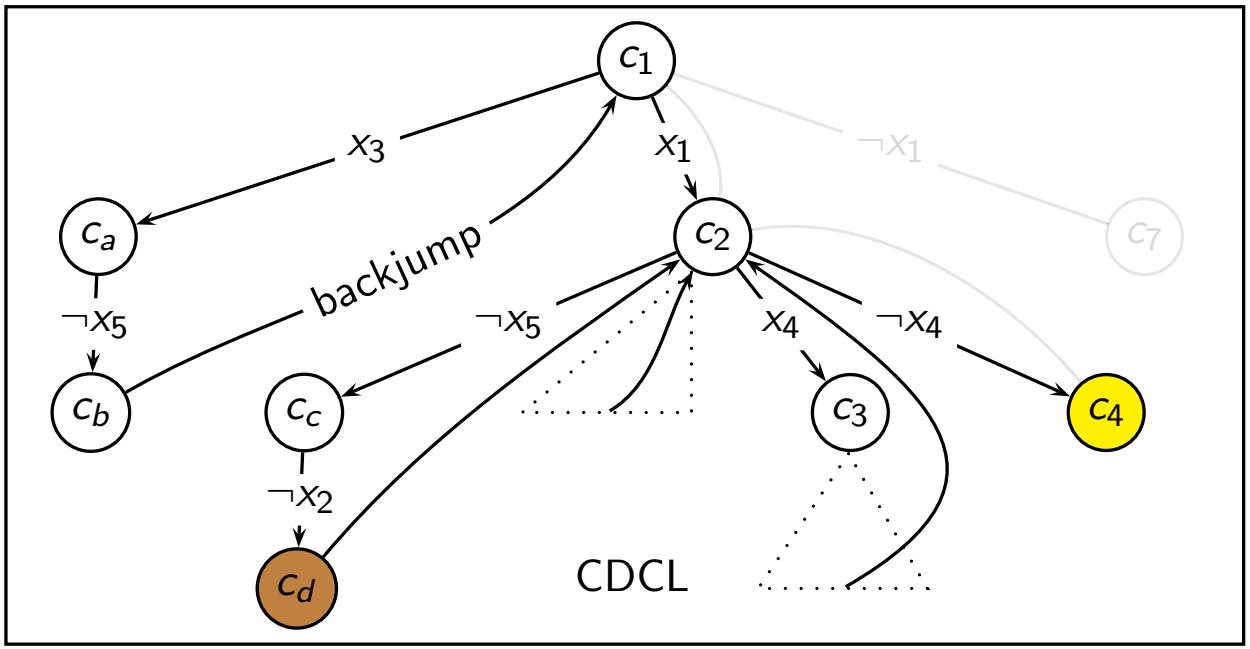
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_4, c_4, 1 \rangle$



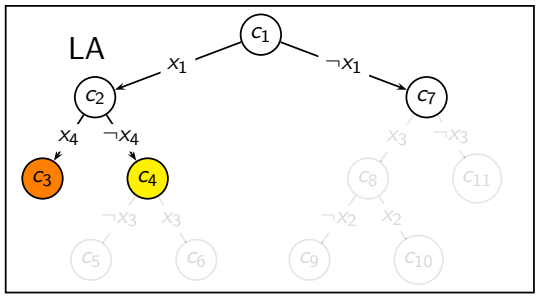
c_x cube

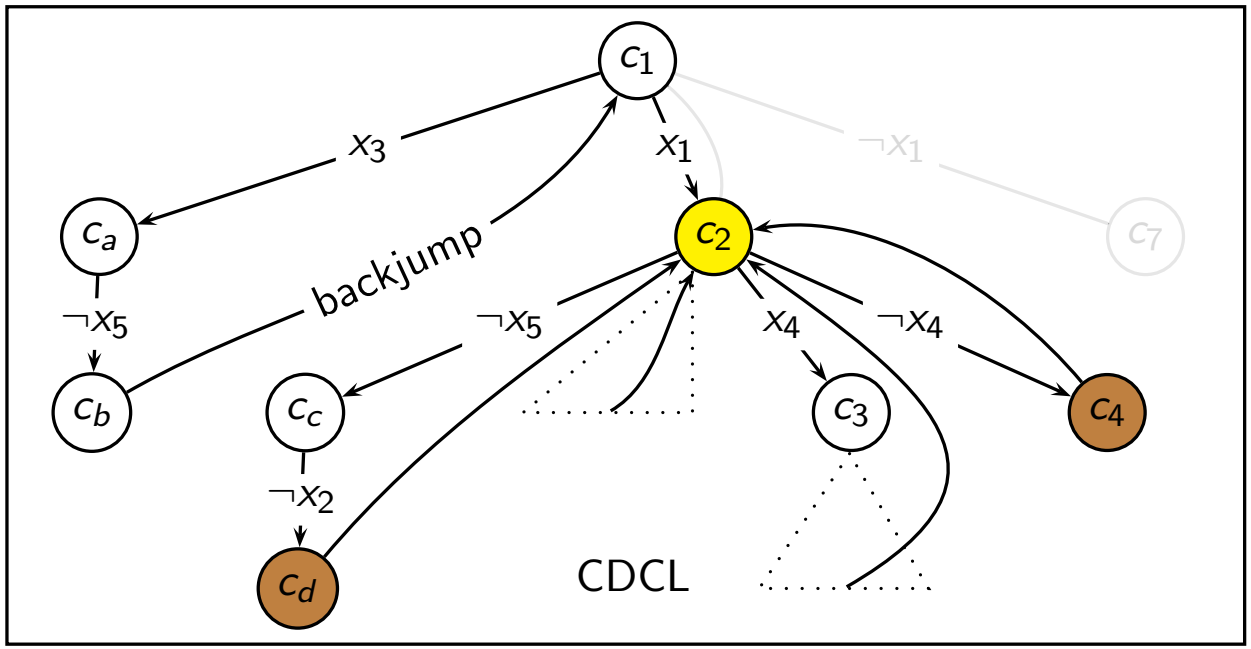
c_x current

c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer





unsat
$\langle c_4 \rangle$

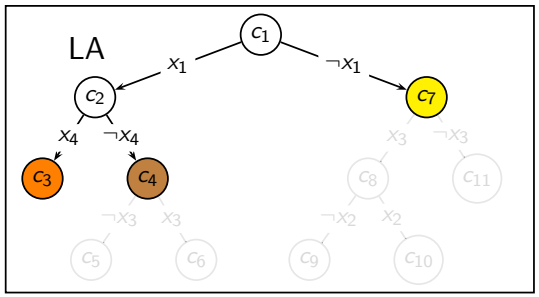
c_x cube

c_x current

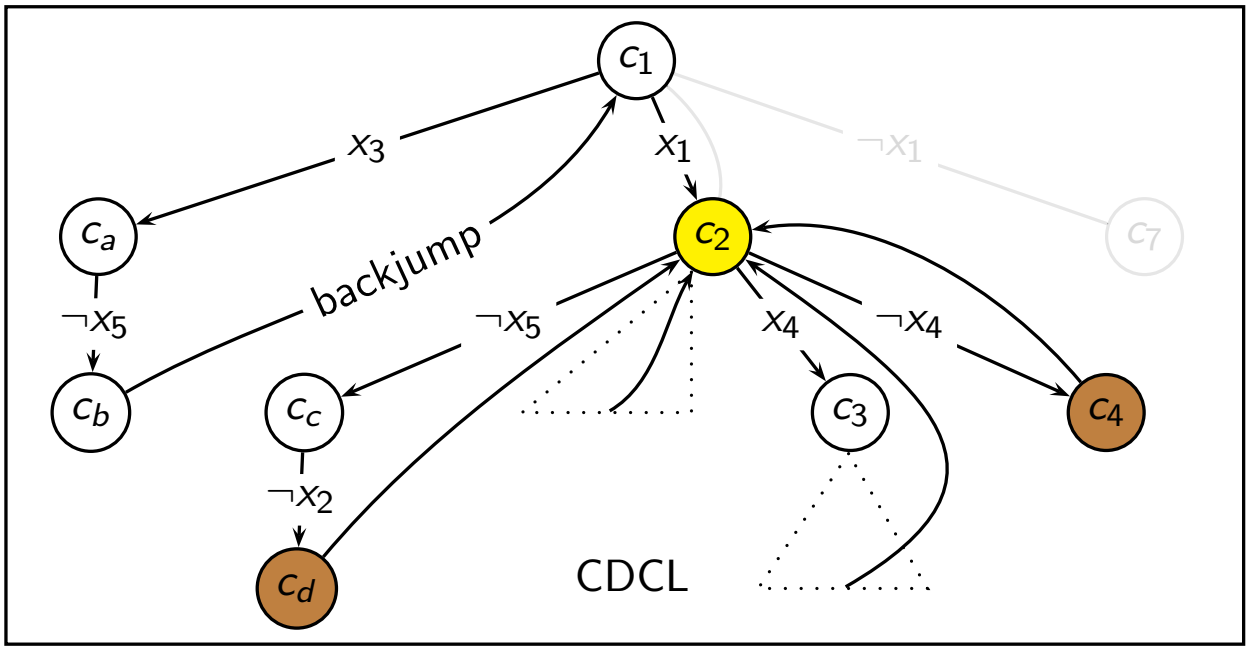
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_1, c_7, 0 \rangle$



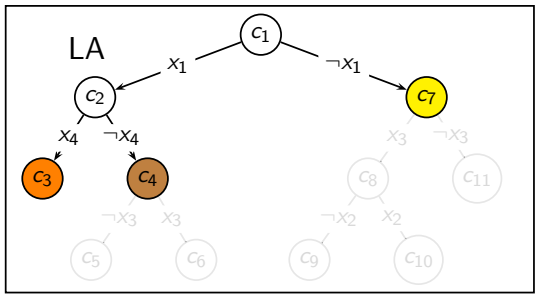
c_x cube

c_x current

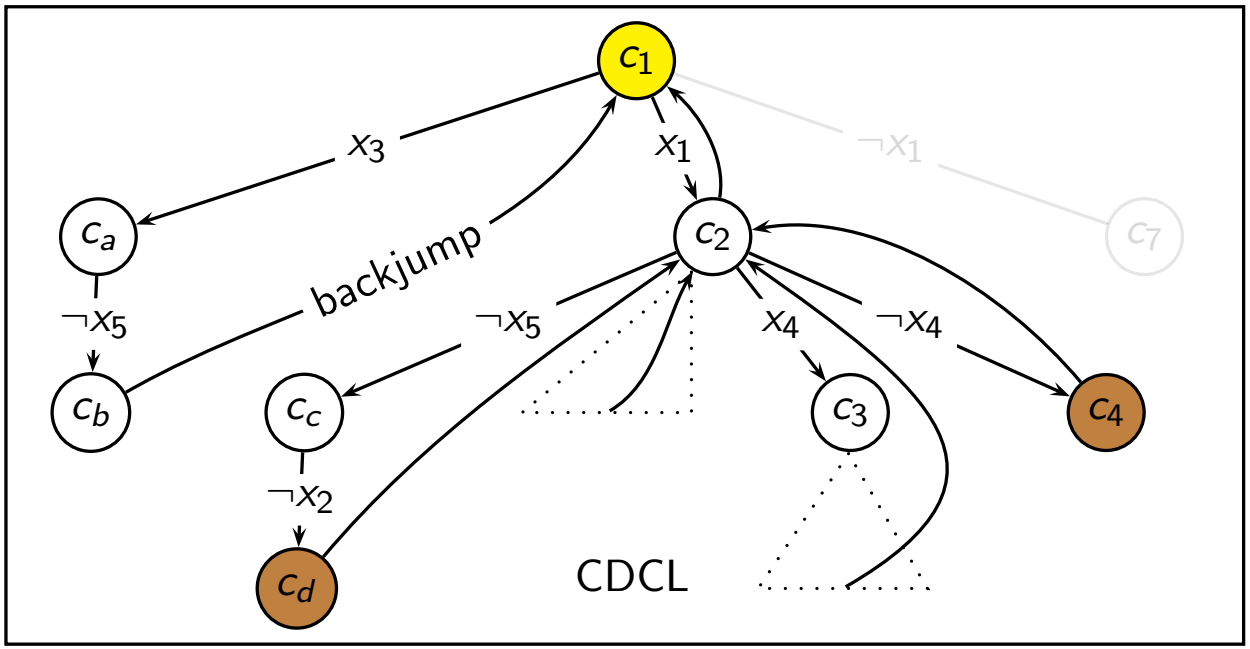
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_1, c_7, 0 \rangle$



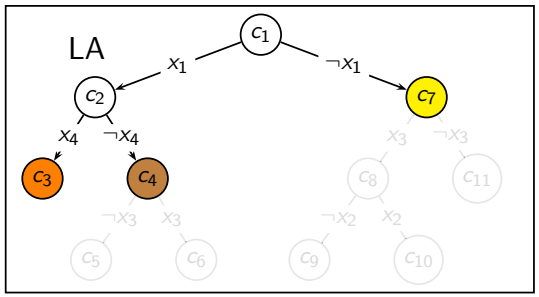
c_x cube

c_x current

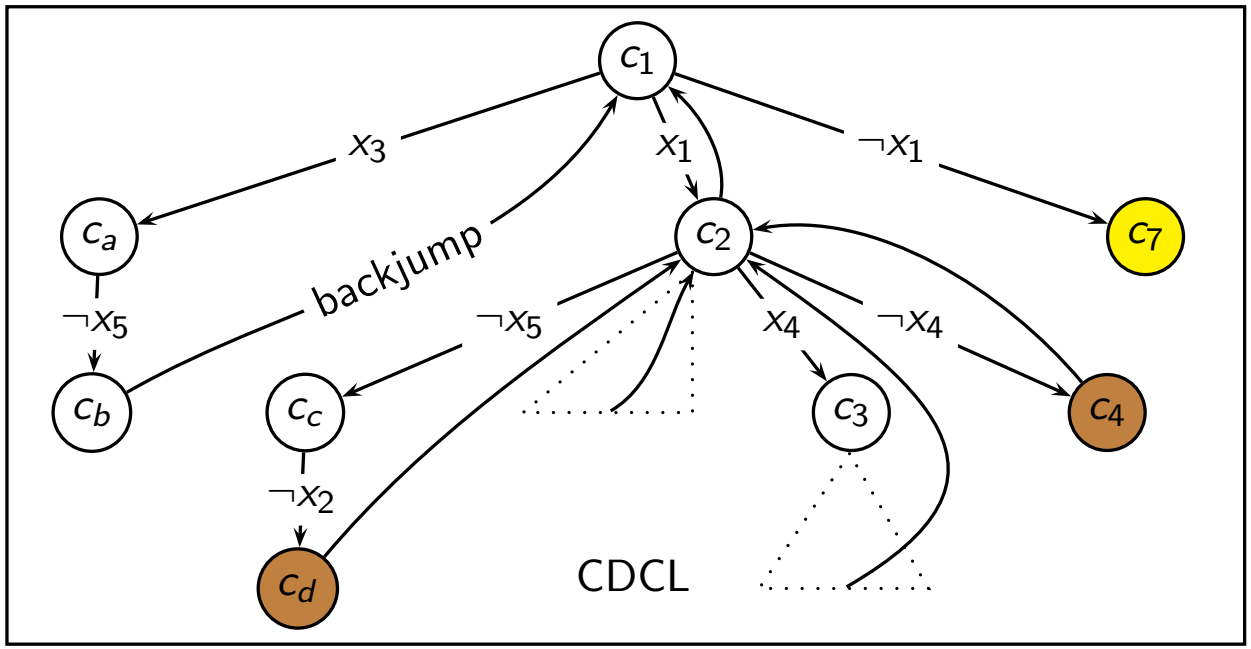
c_x refuted by LA

c_x refuted by CDCL

Concurrent cube-and-conquer



decision
$\langle \neg x_1, c_7, 0 \rangle$



c_x cube

c_x current

c_x refuted by LA

c_x refuted by CDCL

New CCC cutoff heuristic

Use information from the CDCL solver in CCC's cutoff heuristic

Like in CC, $d(c_{\mathrm{id}}) >$ threshold $\rightarrow$ cut off

But now:

- CDCL refutes cube $\rightarrow$ decrease threshold
- LA refutes cube $\rightarrow$ increase threshold (vs. decrease in CC)

Motivation

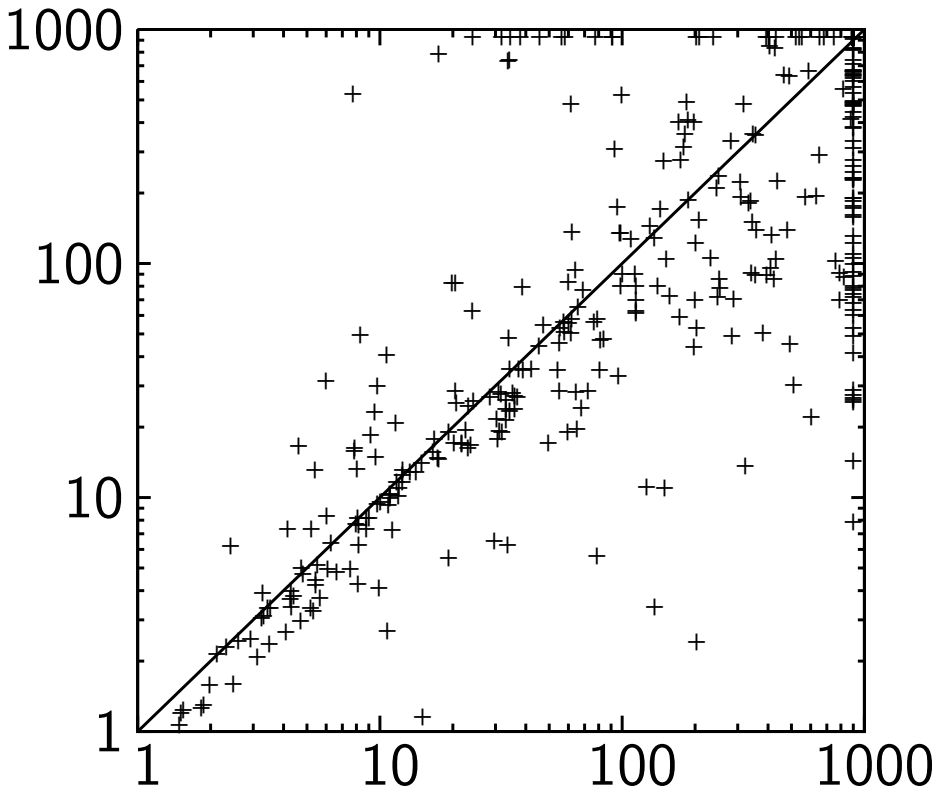
Limitations of cube-and-conquer (CC):

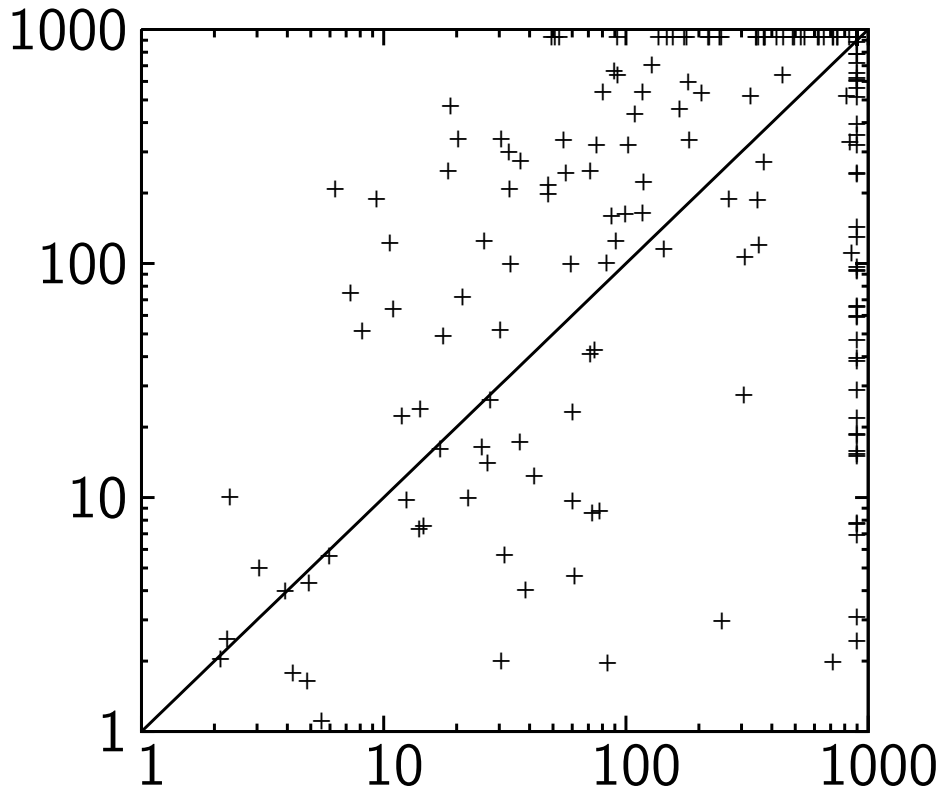
- partitioning not ideal
- lookahead not always effective

Proposed solutions:

- run CDCL and LA concurrently in partitioning phase
- predict unsuitable instances

CCC without prediction





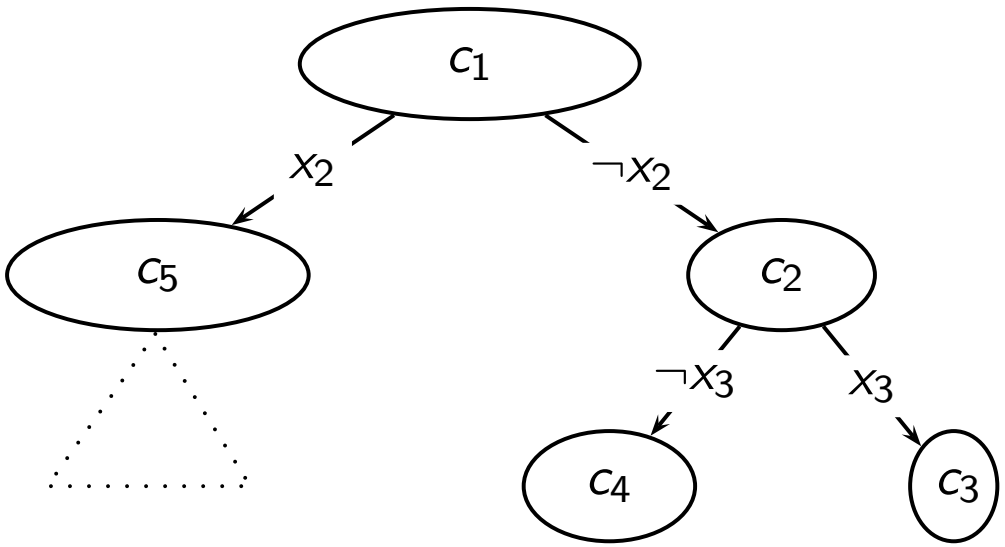
Run time of MiniSAT (vertical) and $\mathrm{CCC}_{\mathrm{mini}}$ (horizontal) on SAT 2009 and 2011 instances.
Application (left) and crafted (right) instances.

Predicting effectiveness of (C)CC

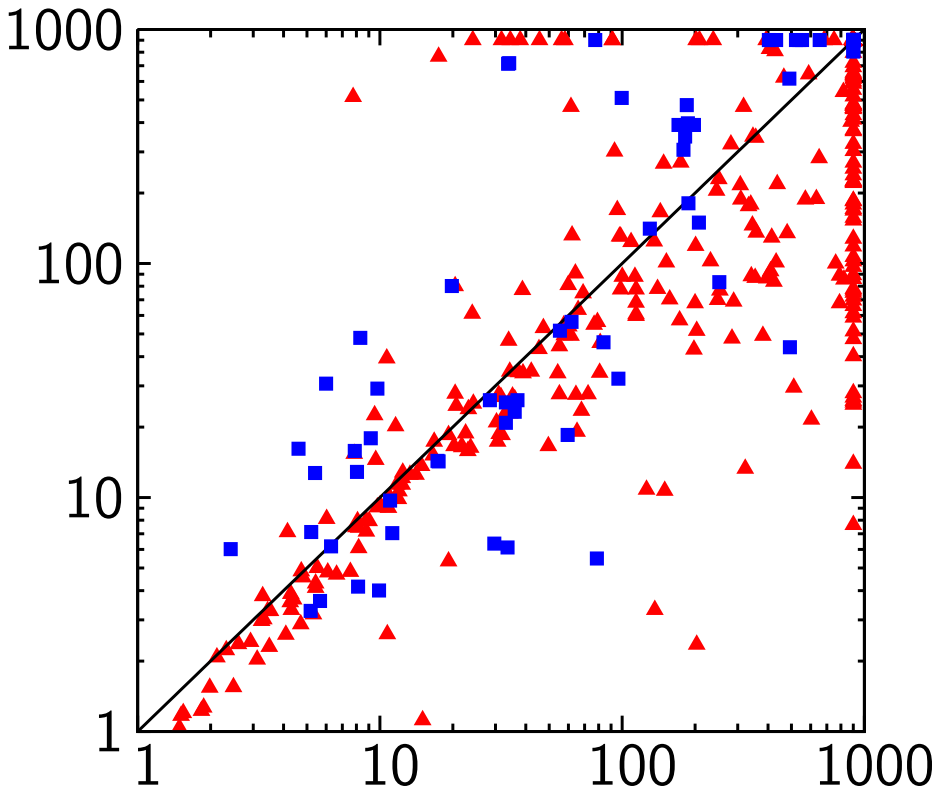
Lookahead effective if

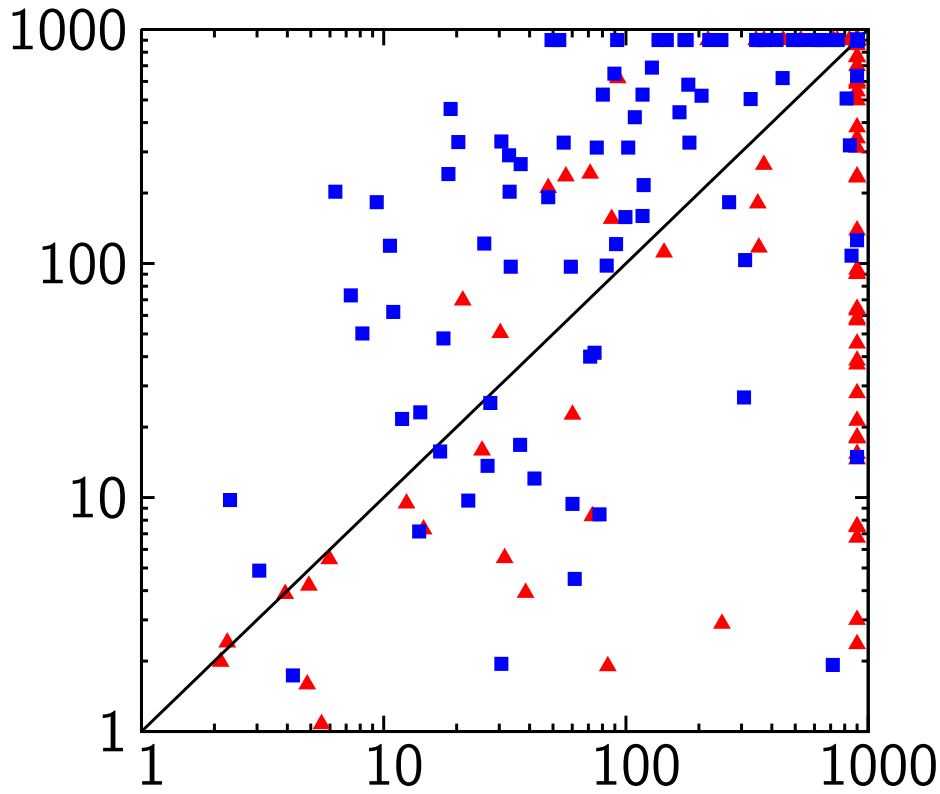
- lookahead refutes cubes before CDCL
- limited number of *right branches*

Use pure CDCL if CCC seems ineffective after 5 seconds



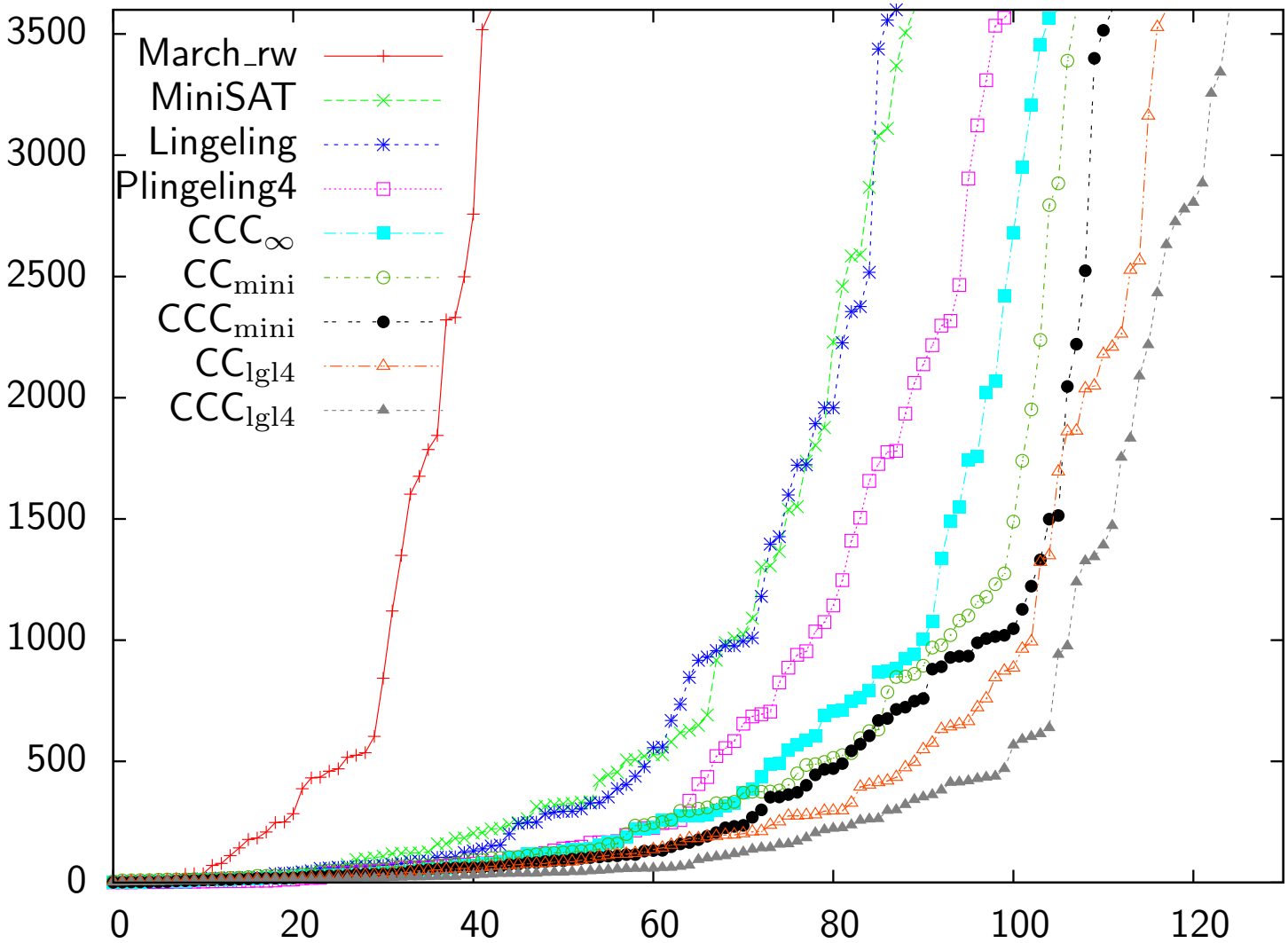
Predicting effectiveness of (C)CC



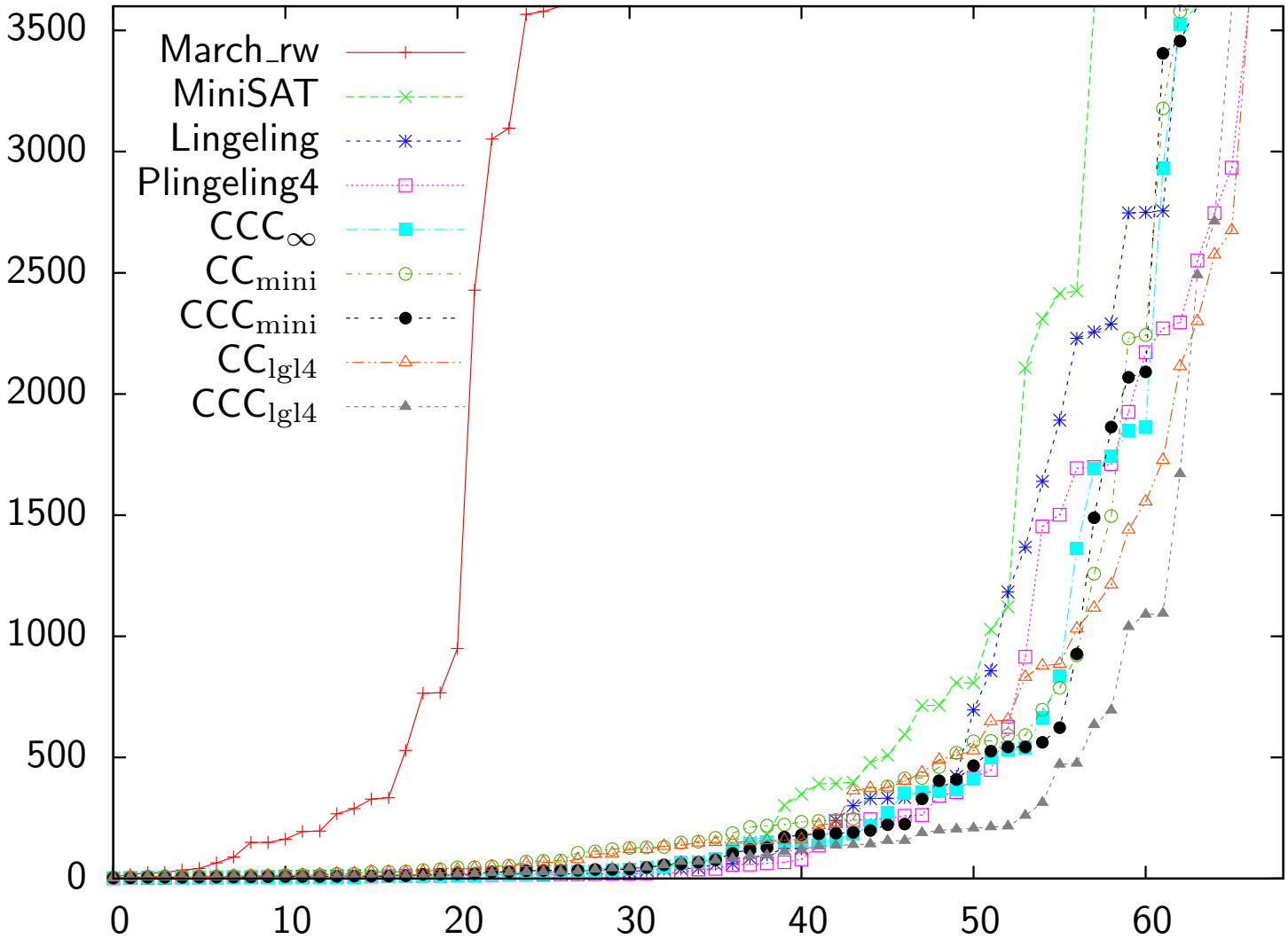


Run time of MiniSAT (vertical) and CCC_{mini} (horizontal) on instances selected (■) and not selected (▲) by the predictor. Application (left) and crafted (right) instances.

Results on filtered crafted instances



Results on filtered application instances



Conclusion

CCC solves CC's limitations

- CCC uses CDCL to find a better cutoff point
- CCC switches to pure CDCL if partitioning performs poorly

And is

- often faster than CDCL, LA, and CC
- natural to parallelize
- and ready to compete in the SAT Challenge 2012

Concurrent Cube-and-Conquer

Peter van der Tak[1]
Marijn Heule[1,2]
Armin Biere[3]

[1]Delft University of Technology, The Netherlands

[2]University of Texas at Austin, United States

[3]Johannes Kepler University Linz, Austria

June 16, 2012