Explanation-Based Propagators for the Table Constraint

Comparing Eager vs. Lazy Explanations in Lazy Clause Generation Solvers

Author: Markas Aisparas

Responsible Professor: Emir Demirović
Supervisor: Maarten Flippo

TU Delft

1. Table Constraint

Encodes valid combinations of variable assignments.

x	y	z
1	1	2
1	2	3
2	3	1

Given:
$x \in \{1,2\}$, $y \in \{3\}$, $z \in \{1\}$

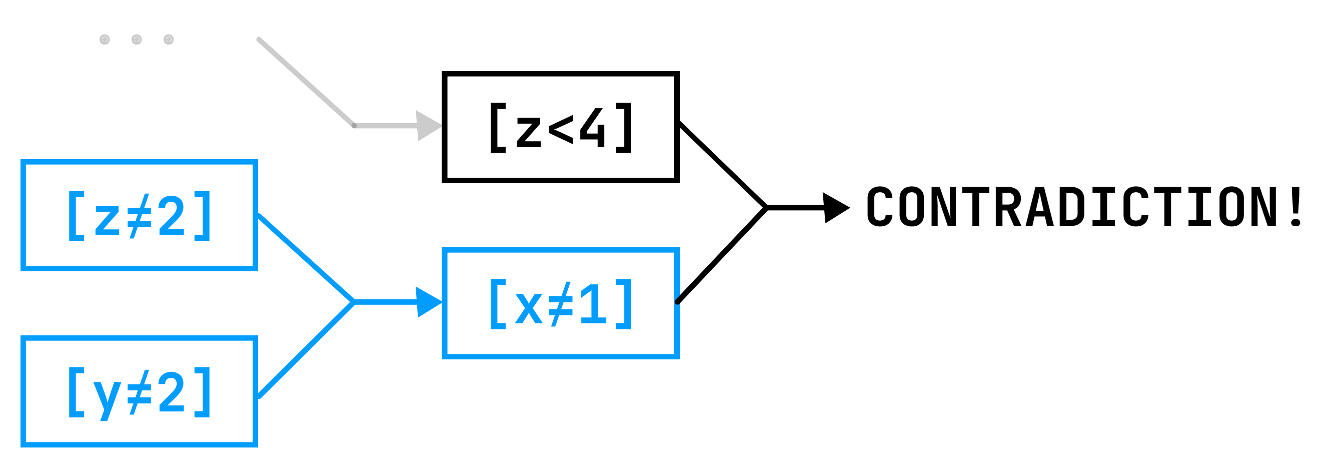
We know $x \neq 1$!

Many possible explanations:

(1). $[y \neq 1] \wedge [y \neq 2]$
(2). $[z \neq 2] \wedge [z \neq 3]$
(3). $[y \neq 1] \wedge [z \neq 3]$
(4). $[z \neq 2] \wedge [y \neq 2]$

2. Implication Graph

LCG solvers chain explanations to show why values become invalid.



(chose explanation 4)

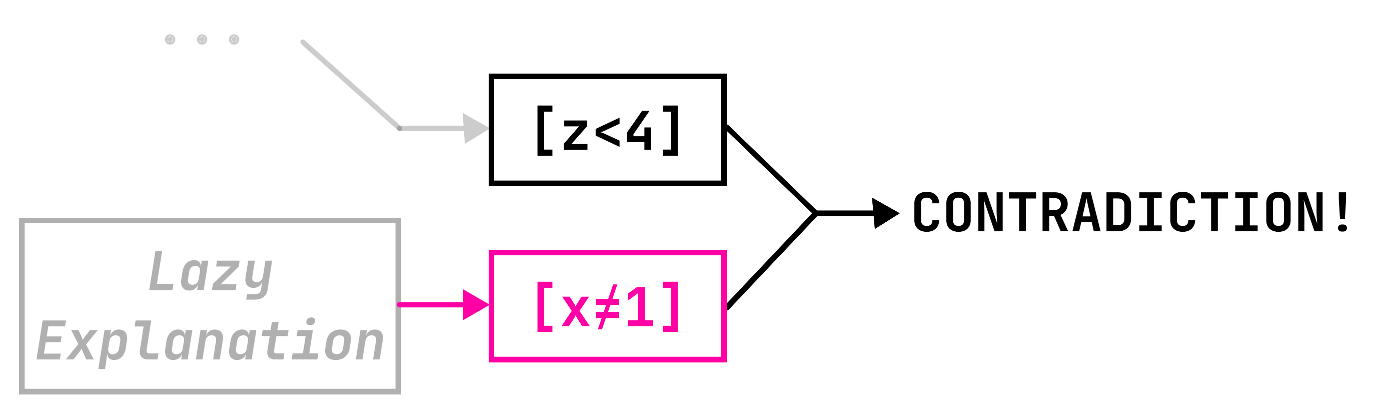
3. Nogood Generation

To avoid the same contradiction in the future, add a new “constraint” called a **nogood**.

$$\begin{tikzcd}[x \neq 1] \wedge [z < 4] \arrow[d, "\text{Expand } [x \neq 1] \text{ explanation}"] \\ [z \neq 2] \wedge [y < 2] \wedge [z < 4]\end{tikzcd}$$

4. Lazy Explanations

Idea: Delay explanation until conflict.

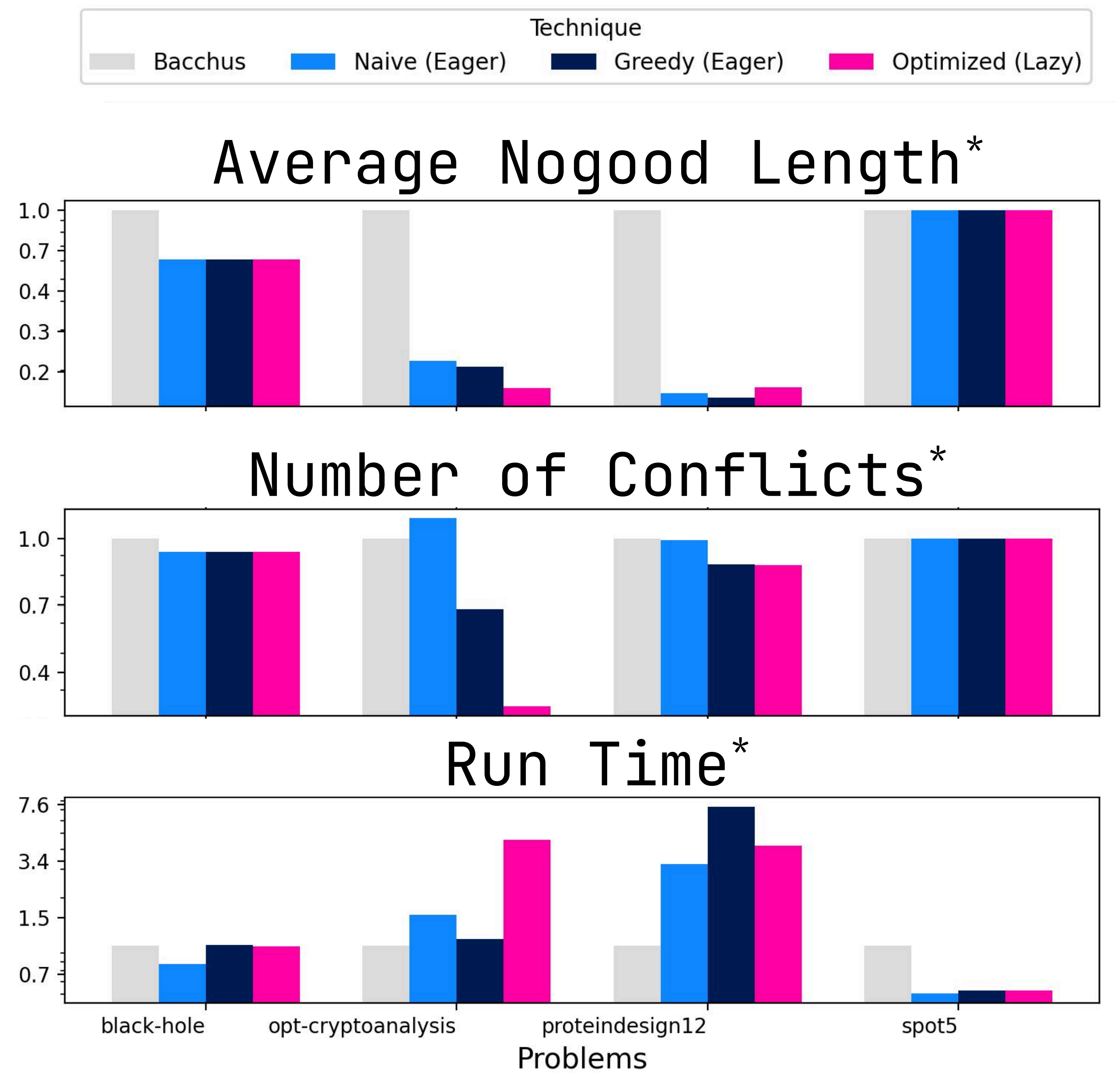


NEW: Choose explanation lazily that minimizes the nogood.

$$\begin{tikzcd}[x \neq 1] \wedge [z < 4] \arrow[d, "\text{Choose explanation 2}"] \\ [z \neq 2] \wedge [z \neq 3] \wedge [z < 4] \arrow[d, "\text{Simplify (subsumption)}"] \\ [z < 2]\end{tikzcd}$$

5. Results & Conclusions

* relative to Bacchus



- Lazy explanations reduce conflicts.
- Explanations are slow to generate.

6. Future Work

- Faster value removal using FD propagators.
- Optimize lazy explanation generation.