The boolean Pythagorean Triples problem

subtitle

Aldo [family name]
Tobias John
Patrick Wienhöft

TU Dresden

date of the presentation

Example

- **Set of Integers: $\{1, \ldots, 15\}$**

Example

- **Set of Integers:** $\{1, \dots, 15\}$
- **Triples:**

$$3^2 + 4^2 = 5^2$$
$$6^2 + 8^2 = 10^2$$
$$9^2 + 12^2 = 15^2$$
$$5^2 + 12^2 = 13^2$$

Example

- **Set of Integers:** $\{1, \dots, 15\}$
- **Triples:**

$$3^2 + 4^2 = 5^2$$
$$6^2 + 8^2 = 10^2$$
$$9^2 + 12^2 = 15^2$$
$$5^2 + 12^2 = 13^2$$

- **Partition:** $\{1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 14\}, \{5, 10, 13, 15\}$

Encoding - Intuition

Idea:

- **one variable for each number**

Encoding - Intuition

Idea:

- **one variable for each number**
- **one constraint clause for each Pythagorean triple**

Encoding - Intuition

Idea:

- **one variable for each number**
- **one constraint clause for each Pythagorean triple**
- **interpretation gives partition**

Encoding

Binary Pythagorean triple problem with n numbers

Encoding

Binary Pythagorean triple problem with n numbers

Set of variables $V = \{p_k \quad | \quad 1 \leq k \leq n\}$

Encoding

Binary Pythagorean triple problem with n numbers

Set of variables $V = \{p_k \mid 1 \leq k \leq n\}$

Constraint for non-equality: $\mathit{NotEqual}(x, y, z) = (x \vee y \vee z) \wedge (\neg x \vee \neg y \vee \neg z)$

Encoding

Binary Pythagorean triple problem with n numbers

Set of variables $V = \{p_k \mid 1 \leq k \leq n\}$

Constraint for non-equality: $NotEqual(x, y, z) = (x \vee y \vee z) \wedge (\neg x \vee \neg y \vee \neg z)$

Constraint for all Pythagorean triples: $F = \bigwedge_{\substack{1 \leq x,y,z \leq n \\ x^2+y^2=z^2}} NotEqual(p_x, p_y, p_z)$

Encoding

Binary Pythagorean triple problem with $\boldsymbol{n}$ numbers

Set of variables $V = \{p_k \mid 1 \leq k \leq n\}$

Constraint for non-equality: $\mathit{NotEqual}(x, y, z) = (x \vee y \vee z) \wedge (\neg x \vee \neg y \vee \neg z)$

Constraint for all Pythagorean triples: $F = \bigwedge_{\substack{1 \leq x,y,z \leq n \\ x^2+y^2=z^2}} \mathit{NotEqual}(p_x, p_y, p_z)$

For an interpretation $I \subseteq V$ with $I \models F$, the resulting partition is:

- $P_1 = \{x \mid p_x \in I\}$
- $P_2 = \{x \mid p_x \notin I\}$

Encoding - Example

As in beginning example, $\boldsymbol{n = 15}$

Encoding - Example

As in beginning example, $n = 15$

$V = \{p_1, p_2, \ldots, p_{15}\}$

Encoding - Example

As in beginning example, $n = 15$

$V = \{p_1, p_2, \ldots, p_{15}\}$

$$
\begin{aligned}
F \quad = \quad & (p_3 \lor p_4 \lor p_5) \land (\neg p_3 \lor \neg p_4 \lor \neg p_5) \\
\land \quad & (p_6 \lor p_8 \lor p_{10}) \land (\neg p_6 \lor \neg p_8 \lor \neg p_{10}) \\
\land \quad & (p_9 \lor p_{12} \lor p_{15}) \land (\neg p_9 \lor \neg p_{12} \lor \neg p_{15}) \\
\land \quad & (p_5 \lor p_{12} \lor p_{13}) \land (\neg p_5 \lor \neg p_{12} \lor \neg p_{13})
\end{aligned}
$$

Encoding - Example

As in beginning example, $n = 15$

$V = \{p_1, p_2, \ldots, p_{15}\}$

$$
\begin{aligned}
F \quad = \quad & (p_3 \vee p_4 \vee p_5) \wedge (\neg p_3 \vee \neg p_4 \vee \neg p_5) \\
\wedge \quad & (p_6 \vee p_8 \vee p_{10}) \wedge (\neg p_6 \vee \neg p_8 \vee \neg p_{10}) \\
\wedge \quad & (p_9 \vee p_{12} \vee p_{15}) \wedge (\neg p_9 \vee \neg p_{12} \vee \neg p_{15}) \\
\wedge \quad & (p_5 \vee p_{12} \vee p_{13}) \wedge (\neg p_5 \vee \neg p_{12} \vee \neg p_{13})
\end{aligned}
$$

Possible interpretation: $I = \{p_5, p_{10}, p_{13}, p_{15}\}$

Encoding - Example

As in beginning example, $n = 15$

$V = \{p_1, p_2, \ldots, p_{15}\}$

$$\begin{aligned} F \quad = \quad & (p_3 \vee p_4 \vee p_5) \wedge (\neg p_3 \vee \neg p_4 \vee \neg p_5) \\ \wedge \quad & (p_6 \vee p_8 \vee p_{10}) \wedge (\neg p_6 \vee \neg p_8 \vee \neg p_{10}) \\ \wedge \quad & (p_9 \vee p_{12} \vee p_{15}) \wedge (\neg p_9 \vee \neg p_{12} \vee \neg p_{15}) \\ \wedge \quad & (p_5 \vee p_{12} \vee p_{13}) \wedge (\neg p_5 \vee \neg p_{12} \vee \neg p_{13}) \end{aligned}$$

Possible interpretation: $I = \{p_5, p_{10}, p_{13}, p_{15}\}$

Resulting partition:

- $P_1 = \{1, 2, 3, 4, 6, 7, 8, 9, 11, 12, 14\}$
- $P_2 = \{5, 10, 13, 15\}$

Transformation

Goal: from F, find formula F' which

- **is easier to solve**

Transformation

Goal: from F, find formula F' which

- **is easier to solve**
- **preserves satisfiability**

Transformation

Goal: from F, find formula F' which

- **is easier to solve**
- **preserves satisfiability**
- **has models that can be easily transformed into models for F**

Transformation

Goal: from F, find formula F' which

- **is easier to solve**
- **preserves satisfiability**
- **has models that can be easily transformed into models for F**

Approaches:

- **eliminate clauses with variables only occuring once**

Transformation

Goal: from F, find formula F' which

- **is easier to solve**
- **preserves satisfiability**
- **has models that can be easily transformed into models for F**

Approaches:

- **eliminate clauses with variables only occuring once**
- **break partition symmetry**

Clause elimination

$$
\begin{aligned}
F \quad = \quad & \mathit{NotEqual}(p_3, p_4, p_5) \\
\wedge \quad & \mathit{NotEqual}(p_6, p_9, p_{12}) \\
\wedge \quad & \mathit{NotEqual}(p_9, p_{12}, p_{15}) \\
\wedge \quad & \mathit{NotEqual}(p_5, p_{12}, p_{13})
\end{aligned}
$$

Note p_3 only occurs in $\mathit{NotEqual}(p_3, p_4, p_5)$

Clause elimination

$$
\begin{aligned}
F \quad = \quad & \mathit{NotEqual}(p_3, p_4, p_5) \\
\wedge \quad & \mathit{NotEqual}(p_6, p_9, p_{12}) \\
\wedge \quad & \mathit{NotEqual}(p_9, p_{12}, p_{15}) \\
\wedge \quad & \mathit{NotEqual}(p_5, p_{12}, p_{13})
\end{aligned}
$$

Note p_3 only occurs in $\mathit{NotEqual}(p_3, p_4, p_5)$

- **if $p_4^I \neq p_5^I$ then $\mathit{NotEqual}(p_3, p_4, p_5)$ is satisfied**

Clause elimination

$$
\begin{aligned}
F \quad = \quad & \mathit{NotEqual}(p_3, p_4, p_5) \\
\wedge \quad & \mathit{NotEqual}(p_6, p_9, p_{12}) \\
\wedge \quad & \mathit{NotEqual}(p_9, p_{12}, p_{15}) \\
\wedge \quad & \mathit{NotEqual}(p_5, p_{12}, p_{13})
\end{aligned}
$$

Note p_3 only occurs in $\mathit{NotEqual}(p_3, p_4, p_5)$

- **if $p_4^I \neq p_5^I$ then $\mathit{NotEqual}(p_3, p_4, p_5)$ is satisfied**
- **if $p_4^I = p_5^I = \top$ then choose $p_3^I = \bot$**

Clause elimination

$$
\begin{aligned}
F \quad = \quad & \mathit{NotEqual}(p_3, p_4, p_5) \\
\wedge \quad & \mathit{NotEqual}(p_6, p_9, p_{12}) \\
\wedge \quad & \mathit{NotEqual}(p_9, p_{12}, p_{15}) \\
\wedge \quad & \mathit{NotEqual}(p_5, p_{12}, p_{13})
\end{aligned}
$$

Note p_3 only occurs in $\mathit{NotEqual}(p_3, p_4, p_5)$

- **if $p_4^I \neq p_5^I$ then $\mathit{NotEqual}(p_3, p_4, p_5)$ is satisfied**
- **if $p_4^I = p_5^I = \top$ then choose $p_3^I = \bot$**
- **if $p_4^I = p_5^I = \bot$ then choose $p_3^I = \top$**

Clause elimination

$$
\begin{aligned}
F \quad = \quad & \mathit{NotEqual}(p_3, p_4, p_5) \\
\land \quad & \mathit{NotEqual}(p_6, p_9, p_{12}) \\
\land \quad & \mathit{NotEqual}(p_9, p_{12}, p_{15}) \\
\land \quad & \mathit{NotEqual}(p_5, p_{12}, p_{13})
\end{aligned}
$$

Note p_3 only occurs in $\mathit{NotEqual}(p_3, p_4, p_5)$

- **if $p_4^I \neq p_5^I$ then $\mathit{NotEqual}(p_3, p_4, p_5)$ is satisfied**
- **if $p_4^I = p_5^I = \top$ then choose $p_3^I = \bot$**
- **if $p_4^I = p_5^I = \bot$ then choose $p_3^I = \top$**

$\rightarrow$ clause $\mathit{NotEqual}(p_3, p_4, p_5)$ will not cause conflict

Clause elimination

$$\begin{aligned} F \quad = \quad & NotEqual(p_3, p_4, p_5) \\ \wedge \quad & NotEqual(p_6, p_9, p_{12}) \\ \wedge \quad & NotEqual(p_9, p_{12}, p_{15}) \\ \wedge \quad & NotEqual(p_5, p_{12}, p_{13}) \end{aligned}$$

Note p_3 only occurs in $NotEqual(p_3, p_4, p_5)$

- **if $p_4^I \neq p_5^I$ then $NotEqual(p_3, p_4, p_5)$ is satisfied**
- **if $p_4^I = p_5^I = \top$ then choose $p_3^I = \bot$**
- **if $p_4^I = p_5^I = \bot$ then choose $p_3^I = \top$**

$\rightarrow$ clause $NotEqual(p_3, p_4, p_5)$ will not cause conflict

$$\begin{aligned} F' \quad = \quad & NotEqual(p_6, p_9, p_{12}) \\ \wedge \quad & NotEqual(p_9, p_{12}, p_{15}) \\ \wedge \quad & NotEqual(p_5, p_{12}, p_{13}) \end{aligned}$$

Clause elimination

Notes:

- ***F'* has more models than *F***

Clause elimination

Notes:

- **F' has more models than F**
- **choice for variables occurring only once in F is important but not represented in F'**

Clause elimination

Notes:

- **F' has more models than F**
- **choice for variables occurring only once in F is important but not represented in F'**
- **$\rightarrow$ remember deleted clauses and modify interpretation of F' accordingly**

Clause elimination

Notes:

- **F' has more models than F**
- **choice for variables occurring only once in F is important but not represented in F'**
- **$\rightarrow$ remember deleted clauses and modify interpretation of F' accordingly**

Example:

- **deleted clause $\mathit{NotEqual}(p_3, p_4, p_5)$ because p_3 only occurred once**

Clause elimination

Notes:

- **F' has more models than F**
- **choice for variables occurring only once in F is important but not represented in F'**
- **$\rightarrow$ remember deleted clauses and modify interpretation of F' accordingly**

Example:

- **deleted clause $NotEqual(p_3, p_4, p_5)$ because p_3 only occurred once**
- **$I' \models F'$ and $\{p_3, p_4, p_5\} \subseteq I'$**

Clause elimination

Notes:

- **F' has more models than F**
- **choice for variables occurring only once in F is important but not represented in F'**
- **$\rightarrow$ remember deleted clauses and modify interpretation of F' accordingly**

Example:

- **deleted clause $NotEqual(p_3, p_4, p_5)$ because p_3 only occurred once**
- **$I' \models F'$ and $\{p_3, p_4, p_5\} \subseteq I'$**
- **$I = I' \setminus p_3$**
- **while $I' \not\models F$, we have $I \models F$**

Breaking Symmetry

- formulas F and F' are symmetric

Breaking Symmetry

- **formulas F and F' are symmetric**
- **if $I \models F'$ then $V \setminus I \models F'$**

Breaking Symmetry

- **formulas F and F' are symmetric**
- **if $I \models F'$ then $V \setminus I \models F'$**
- **solution: introduce unit clause for variable occurring in F'**
- $F'' = F' \wedge p_x$

Cube-and-conquer solving

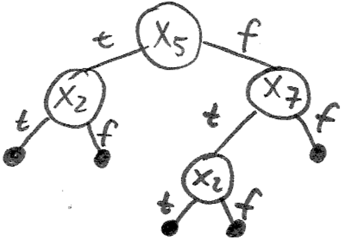
- **Problem: solving with conflict-driven clause leraning (CDCL) is too slow**

Cube-and-conquer solving

- **Problem: solving with conflict-driven clause leraning (CDCL) is too slow**
- **Solution: use different heuristics $\Rightarrow$ cube-and-conquer solver (C&C)**

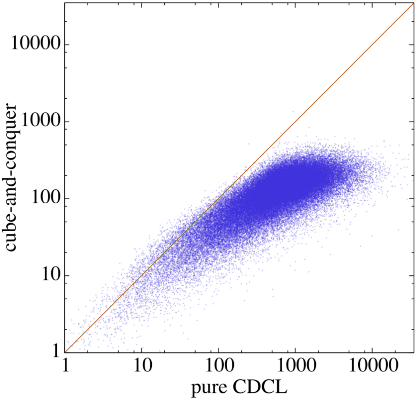
Cube-and-conquer solving

- **Problem: solving with conflict-driven clause leraning (CDCL) is too slow**
- **Solution: use different heuristics ⇒ cube-and-conquer solver (C&C)**



Runtime

- **x time**



Validation of the program