Explainable Product Configuration for Telecom using Constraint Reasoning

UPPSALA UNIVERSITET

Master Thesis Presentation

Paul Häusner
Supervisor: Alessandro Previti (Ericsson Research)
Reviewer: Jens Sjölund

February 6, 2022

Department of Information Technology, Uppsala University

Outline

UPPSALA UNIVERSITET

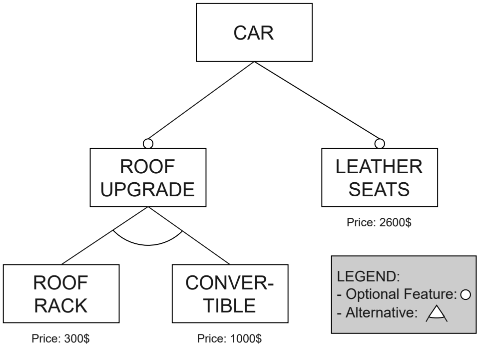
Product Configuration

UPPSALA UNIVERSITET

Feature Model for describing product lines

- Set of available features with attributes and relationship between them
- Technical and physical constraints, product design choices, ...
- A user has preferences of the final product

Can be described as a Constraint Satisfaction Problem which allows automated reasoning



Constraint Satisfaction Problems

A Constraint Satisfaction Problem (CSP) is a triplet $(\mathcal{V}, \mathcal{D}, \mathcal{C})$:

- $\mathcal{V}$ is a sequence of variables
- $\mathcal{D}$ is a corresponding sequence of domains
- $\mathcal{C}$ is a set of constraints: Every constraint is a relation over a subset of variables and describes which values are allowed to be assigned to its variables

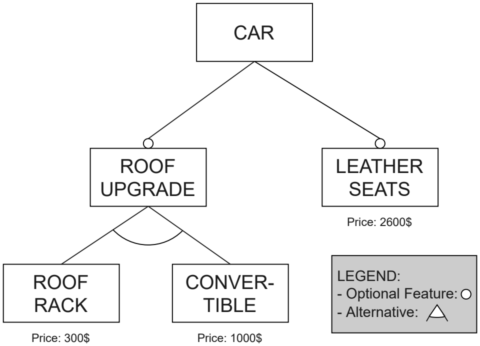
Solving a CSP: Assigning every variable a single value from its domain without violating the constraints (this is in general an $\mathcal{NP}$-hard problem)

If no such assignment exist we say the problem is *unsatisfiable* else *satisfiable*

Model the features as variables of the CSP:

- Boolean variables s_1, s_2, s_3
- Technical constraint: $f_1 = \neg(s_1 \wedge s_2)$
- User selecting a feature: $c_i = s_i$ (forcing the variable to *TRUE*).
- User preference: $c_4 = price \leq 3000$

Assume a user selects all features and the price constraint $\Rightarrow$ *unsatisfiable* model



Outline

UPPSALA UNIVERSITET

Explanations

UPPSALA UNIVERSITET

Goal: Find an explanation for the unsatisfiability of the model (based on the user preferences $\mathcal{F}$):

MUS (Minimal Unsatisfiable Subset): A subset of relevant constraints $\mathcal{U} \subseteq \mathcal{F}$ such that the constraints $\mathcal{U}$ are unsatisfiable but no proper subset of $\mathcal{U}$ is unsatisfiable

MCS (Minimal Correction Subset): A subset of relevant constraints $\mathcal{C} \subseteq \mathcal{F}$ such that the set of constraints $\mathcal{F} \setminus \mathcal{C}$ is satisfiable but by adding any constraint c back to the problem, the problem becomes unsatisfiable again.

MSS (Maximal Satisfiable Set): Complement of the MCS.

The number of MUSes and MCSes of a problem is exponential wrt the number of constraints.

Enumeration of Explanations

Finding all (or as many as possible) explanations for a problem: **MARCO** algorithm

- Partial and Full enumeration MUSes and MCSes possible
- Keep track of explored subsets with *map* formula (one variable per constraint)

Core algorithm:

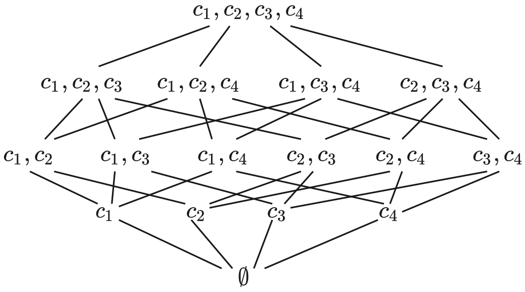
1. Start with an arbitrary unexplored subset *seed*
2. If *seed* is satisfiable: add constraints until it is maximal (grow)
3. If *seed* is unsatisfiable: remove constraints until it is minimal (shrink)
4. Add a clause to the *map* formula to block subsets where satisfaiability is known
5. If *map* is still satisfiable go back to step 1, otherwise full enumeration is achieved

Reminder:

- c_1, c_2, c_3 are for selecting the features (prices: 300, 1000, 2600)
- c_4 restricts the price ≤ 3000
- s_1 and s_2 can not be true at the same time (technical constraint)

The *map* formula gets initialized empty: *map* $\models \top$

Visualizing the power set as a Hasse diagram:

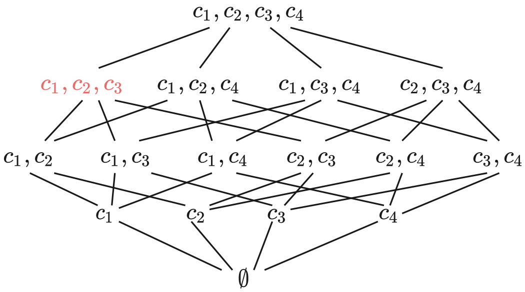


MARCO - Car Configuration (cont.)

Get a *seed*: any satisfiable model from the *map*

- Assume: $seed = \{c_1, c_2, c_3\}$
- Call solver $\Rightarrow$ unsatisfiable
- Shrink to extract MUS $\{c_1, c_2\}$
- Blocking up (setting all supersets of the MUS to explored)
- $map \models \neg c_1 \vee \neg c_2$

Updated Hasse diagram (after initial satisfiability check):

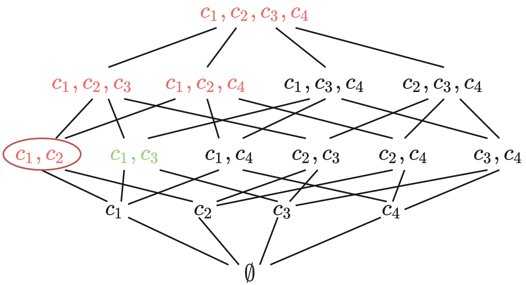


MARCO - Car Configuration (cont.)

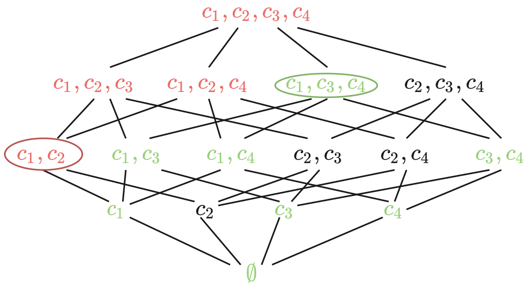
Get a *seed*: any satisfiable model from the *map*

- Assume: $seed = \{c_1, c_3\}$
- Call solver $\Rightarrow$ satisfiable
- Grow to MSS $\{c_1, c_3, c_4\}$
- Blocking down (setting all subsets of the MSS to explored)
- $map \models (\neg c_1 \lor \neg c_2) \land c_2$

Updated Hasse diagram (after first MUS extraction):



Updated Hasse diagram (after first MSS extraction):



Repeat this procedure until either

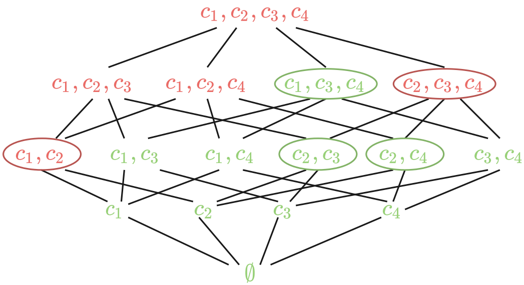
- the *map* formula is unsatisfiable
- a timeout is reached

UPPSALA UNIVERSITET

Advantages of the MARCO algorithm

- Explanations are created early
- Both full and partial enumeration possible
- Agnostic of grow/shrink algorithm

Updated Hasse diagram (after full enumeration):



Outline

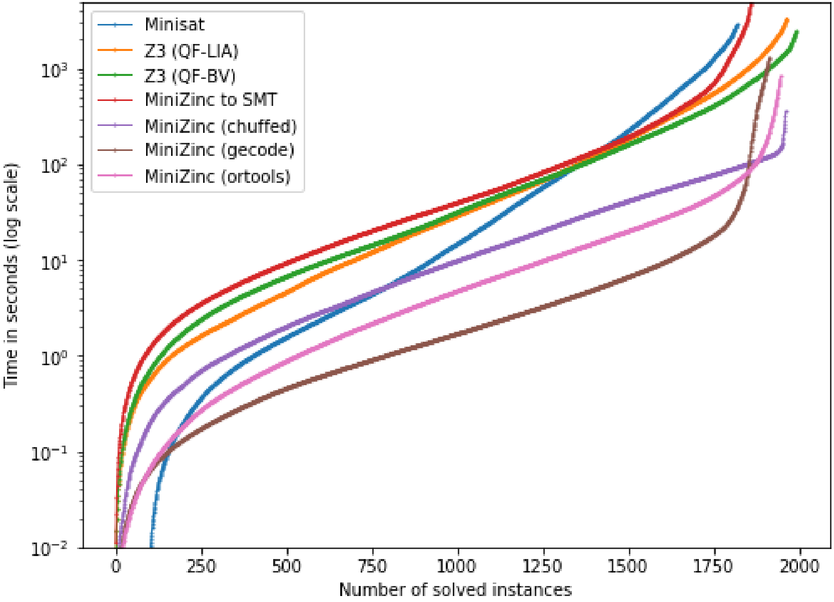
General: Compare performance of different encodings of the product configuration problem from the *telecom* domain

- Compare performance on satisfiability checks
- Compare performance on explanation computation

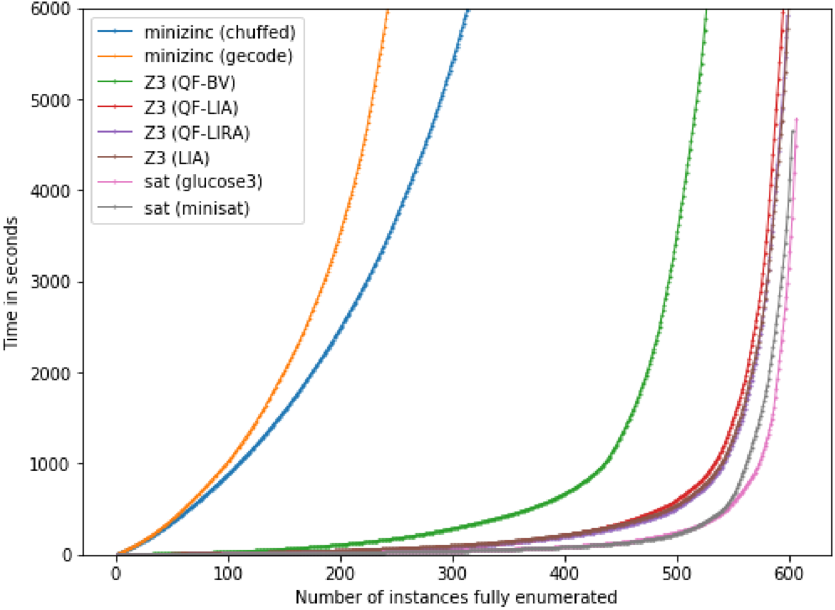
Outline

- Core telecom problem: 2-dimensional bin-packing and table constraints
- Randomly generated problem instances
- Generated user preferences make the problem unsatisfiable
- Main modeling techniques: MiniZinc, Z3
- Solving techniques: CP, SMT, SAT, (KC)

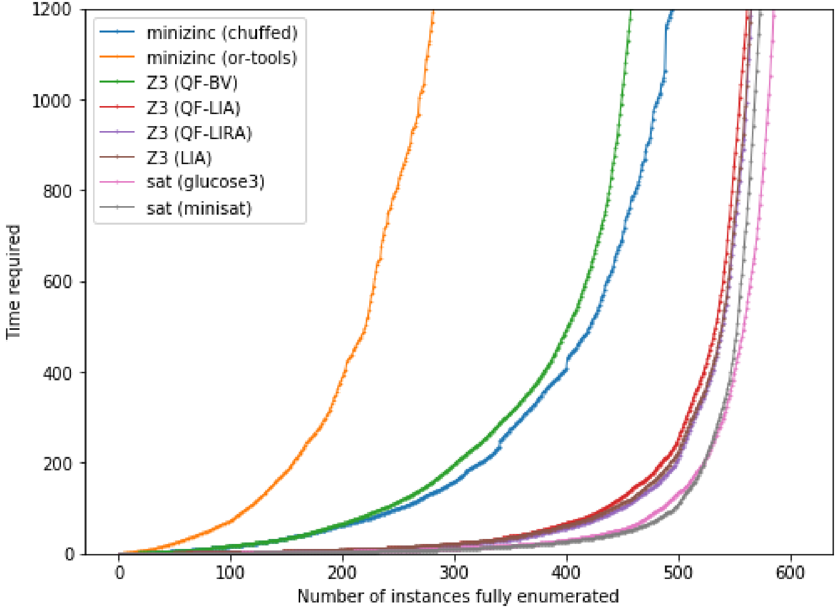
Satisfiability checks

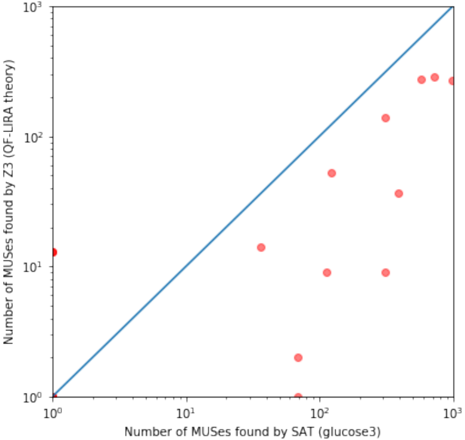


Full explanation enumeration

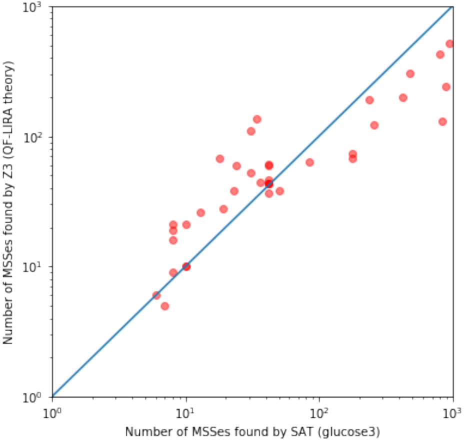


Full explanation enumeration (cont.)





(a) MUSes



(b) MSSes

Figure 1: SAT vs SMT partial enumeration

Outline

Conclusion and Future Work

- There is no single best method for this use-case
- Different tasks require different behavior
- Currently no pre-solving optimization used
- Making use of symmetry-breaking and implied constraints
- Definition of a good explanation is non-trivial

Thank you for your attention