

#### **Thesis Topics: Product-Line Analysis**

Chico Sundermanr Product-Line Analysis | 01.02.2023







# Your Thesis or Project with SoftVarE

- Support
  - Weekly meetings
  - Mattermost
- Feedback
  - ► Reviews for each part of your thesis before submission
  - For each chapter by supervisor
  - Intermediate presentation
  - Feedback by Thomas





- Thesis topics relevant to research
- Possibility of participating in future publications
- For Everyone! B (Bachelor) P (Project) M (Master)

### **About Me**

- My name is Chico
- 2015–2020: studied Computer Science in Braunschweig
- 2020–?: PhD student at Institute of Software Engineering and Programming Languages in Ulm
- Main research:
  - Product-line analysis
  - ► #SAT
  - Knowledge Compilation



#### **Product Lines**



#### **Feature Models**



Navigation ⇒ USB

## Complexity



F\_6B23DD86CE4C ⇒ F\_287D032B6EC5 F\_75372381F32E ⇒ F\_8A42FBF6E175 F\_E976C9C42885 ⇒ F\_E2F1C99E45CF ¬F\_1289140DEC8F ⇒ ¬F\_477DA48AC4E3 F\_71ED362684B0 ⇒ ¬F\_E0E6FF46990C F 71ED362684B0 ⇒ ¬F\_2994A4D5A93B









Thesis Topics: Product-Line Analysis - Product-Line Analysis

#### **Current State**

Insights

#### There are plenty of applications Computations are expensive

How to improve performance?

#### How can we do better?



# Topic 1: Meta #SAT Solver B/P/M

- Problem
  - ► #SAT solvers perform differently depending on instance
  - Beneficial to switch between solvers
  - Strong performance in regular SAT

# Topic 1: Meta #SAT Solver B/P/M

- Problem
  - ► #SAT solvers perform differently depending on instance
  - Beneficial to switch between solvers
  - Strong performance in regular SAT
- Tasks
  - Gather promising #SAT solvers
  - Analyze structural properties of feature models
  - Develop heuristic(s) to select solvers depending on structure
  - Empirically evaluate heuristic(s)

### **Similar Queries**

Feature Model Formula =  $F_{FM}$ 



### **Similar Queries**



### **Similar Queries**



## Topic 2: Variational #SAT Solver M

- Problem
  - ► With existing #SAT solvers: separate invocations
  - Possibly large overhead
  - ► Goal: reuse information from single run for multiple computations

## Topic 2: Variational #SAT Solver M

Problem

- ► With existing #SAT solvers: separate invocations
- Possibly large overhead
- ► Goal: reuse information from single run for multiple computations
- Tasks
  - Develop concept for variational #SAT
  - Choice:
    - Develop your own tooling
    - Adapt existing solver
  - Empirically evaluate your concept and tooling

## Knowledge Compilation: d-DNNF



= decomposable Deterministic Negation Normal Form

## Knowledge Compilation: d-DNNF



- = decomposable Deterministic Negation Normal Form
  - Expensive offline phase

## Knowledge Compilation: d-DNNF



- = decomposable Deterministic Negation Normal Form
  - Expensive offline phase
  - Fast online queries: e.g. SAT or #SAT
  - Shown to be very effective for #SAT on feature models

# Topic 3: SAT-based Analysis with d-DNNF P/B/M

- Problem
  - d-DNNF efficient for counting
  - Many SAT-based analyses on feature models
  - ► Goal: Evaluate performance of d-DNNFs for SAT

# Topic 3: SAT-based Analysis with d-DNNF P/B/M

- Problem
  - d-DNNF efficient for counting
  - Many SAT-based analyses on feature models
  - ► Goal: Evaluate performance of d-DNNFs for SAT
- Tasks
  - Identify suitable analyses
  - Replace ordinary SAT solvers with d-DNNF
  - Empirically evaluate advantages of applying d-DNNF

### How to represent? Universal Variability Language



. . .

# Topic 4–X: Advancing Support for UVL P(/B/M)

- Language Server Protocol (LSP)
  - Features
    - Parsing, completion, error messages, simple analysis
    - What else?
  - Currently supported in
    - Visual Studio Code, Vim
- Simplify complex language construct
- Translations from other languages
- . . .

# Wrap-Up

- Topics:
  - ► Topic 1: Meta #SAT Solver P/B/M
  - ► Topic 2: Variational #SAT Solver M
  - ► Topic 3: SAT-based Analysis with d-DNNF P/B/M
  - ► Topic 4-X: Advancing Support for UVL P/B/M
- Research not only for the books
- Compete with the state of the art
- Implement your prototypes
- chico.sundermann@uni-ulm.de