



[Weinbergsquelle, Hohenwarthe]

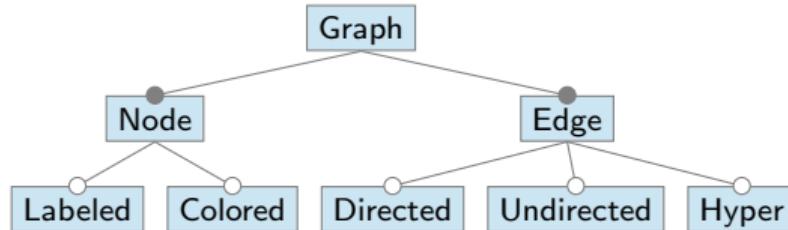
How Easy is SAT-Based Analysis of a Feature Model?

VaMoS 2024 — February 7–9 — Bern, Switzerland

Elias Kuiter¹, Tobias Heß², Chico Sundermann², Sebastian Krieter², Thomas Thüm², Gunter Saake¹

University of Magdeburg¹, Ulm², Germany

Recap: Product-Line and Feature-Model Analysis



$\neg(Directed \wedge Undirected)$

$Hyper \rightarrow Undirected$

$Directed \nleftrightarrow (Undirected \wedge Hyper)$

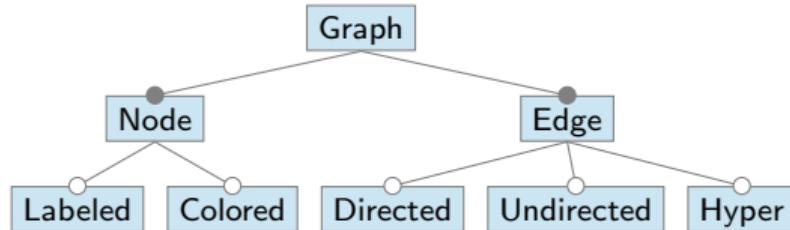
node.cpp

```
class Node {  
    #ifdef LABELED  
    std::string label;  
    #endif  
    #ifdef COLORED  
    std::string color;  
    #endif  
};
```

edge.cpp

```
class Edge {  
    #ifdef DIRECTED  
    Node from, to;  
    #elif UNDIRECTED &&  
        HYPER  
    std::set<Node> nodes;  
    #endif  
};
```

Recap: Product-Line and Feature-Model Analysis



$\neg(Directed \wedge Undirected)$

$Hyper \rightarrow Undirected$

$Directed \nleftrightarrow (Undirected \wedge Hyper)$

Product-Line Analyses ...

[ref]

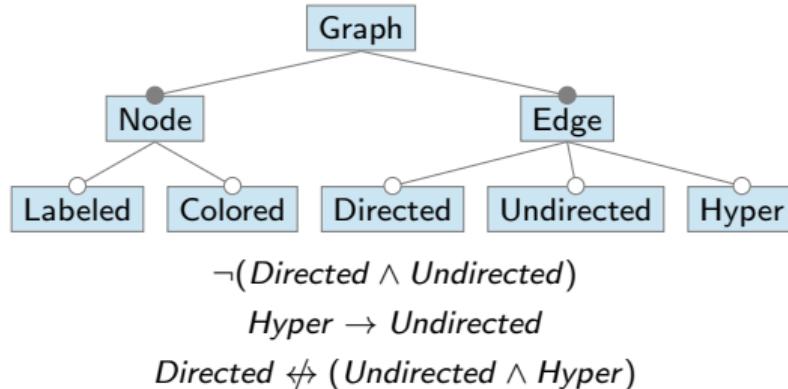
node.cpp

```
class Node {  
    #ifdef LABELED  
    std::string label;  
    #endif  
    #ifdef COLORED  
    std::string color;  
    #endif  
};
```

edge.cpp

```
class Edge {  
    #ifdef DIRECTED  
    Node from, to;  
    #elif UNDIRECTED &&  
        HYPER  
    std::set<Node> nodes;  
    #endif  
};
```

Recap: Product-Line and Feature-Model Analysis



Product-Line Analyses ...

[ref]

Which product(s) ...

- ... has the most **lines of code**? [ref]

- ... is the **fastest** or **smallest**? [ref, ref]

- ... have type or **logic errors**? [ref, ref]

- ... have unsafe **data flows**? [ref]

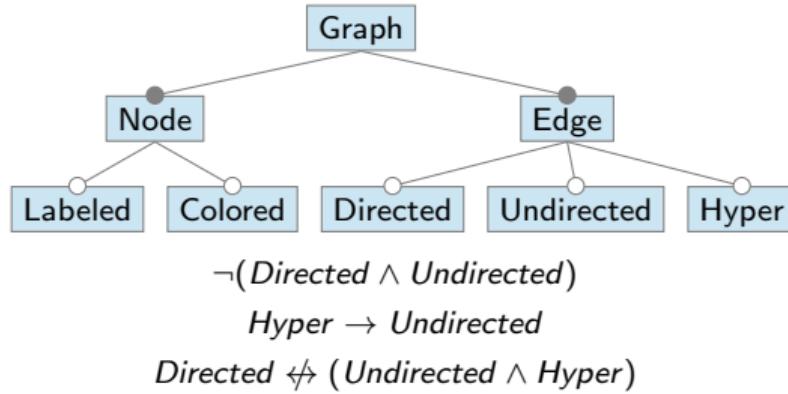
node.cpp

```
class Node {  
    #ifdef LABELED  
        std::string label;  
    #endif  
    #ifdef COLORED  
        std::string color;  
    #endif  
};
```

edge.cpp

```
class Edge {  
    #ifdef DIRECTED  
        Node from, to;  
    #elif UNDIRECTED &&  
        HYPER  
        std::set<Node> nodes;  
    #endif  
};
```

Recap: Product-Line and Feature-Model Analysis



node.cpp

```
class Node {  
    #ifdef LABELED  
    std::string label;  
    #endif  
    #ifdef COLORED  
    std::string color;  
    #endif  
};
```

edge.cpp

```
class Edge {  
    #ifdef DIRECTED  
    Node from, to;  
    #elif UNDIRECTED &&  
        HYPER  
    std::set<Node> nodes;  
    #endif  
};
```

Product-Line Analyses ...

Which product(s) ...

- ... has the most **lines of code**? [ref]

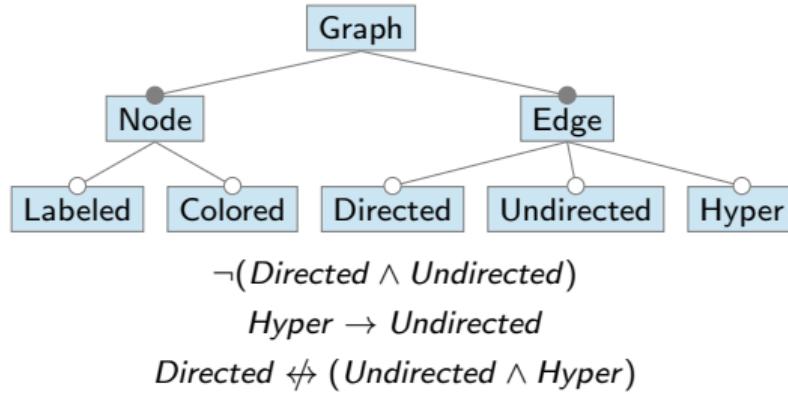
- ... is the **fastest** or **smallest**? [ref, ref]

- ... have type or **logic errors**? [ref, ref]

- ... have unsafe **data flows**? [ref]

... Often Rely on Feature-Model Analyses ...

Recap: Product-Line and Feature-Model Analysis



node.cpp

```
class Node {  
    #ifdef LABELED  
    std::string label;  
    #endif  
    #ifdef COLORED  
    std::string color;  
    #endif  
};
```

edge.cpp

```
class Edge {  
    #ifdef DIRECTED  
    Node from, to;  
    #elif UNDIRECTED && HYPER  
    std::set<Node> nodes;  
    #endif  
};
```

Product-Line Analyses ...

Which product(s) ...

- ... has the most **lines of code**? [ref]

- ... is the **fastest** or **smallest**? [ref, ref]

- ... have type or **logic errors**? [ref, ref]

- ... have unsafe **data flows**? [ref]

... Often Rely on Feature-Model Analyses ...

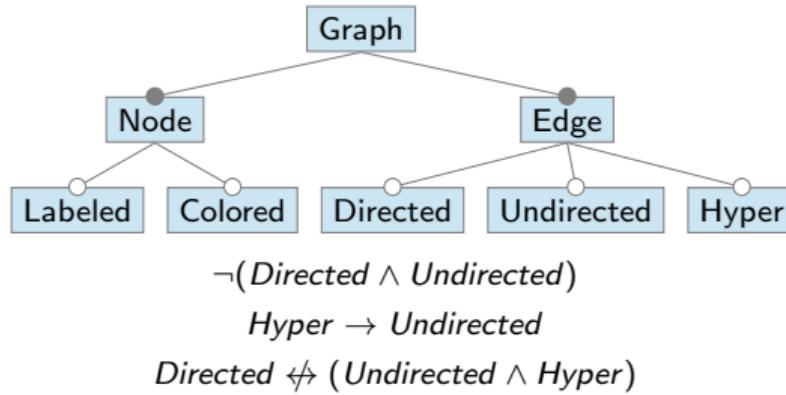
- Which features are **core/dead**? [ref]

- A sample covering all **t-wise interactions**? [ref]

- How large** is the configuration space? [ref]

- How has the configuration space **evolved**? [ref]

Recap: Product-Line and Feature-Model Analysis



node.cpp

```
class Node {  
    #ifdef LABELED  
    std::string label;  
    #endif  
    #ifdef COLORED  
    std::string color;  
    #endif  
};
```

edge.cpp

```
class Edge {  
    #ifdef DIRECTED  
    Node from, to;  
    #elif UNDIRECTED && HYPER  
    std::set<Node> nodes;  
    #endif  
};
```

Product-Line Analyses ...

Which product(s) ...

- ... has the most **lines of code**? [ref]

- ... is the **fastest** or **smallest**? [ref, ref]

- ... have type or **logic errors**? [ref, ref]

- ... have unsafe **data flows**? [ref]

... Often Rely on Feature-Model Analyses ...

- Which features are **core/dead**? [ref]

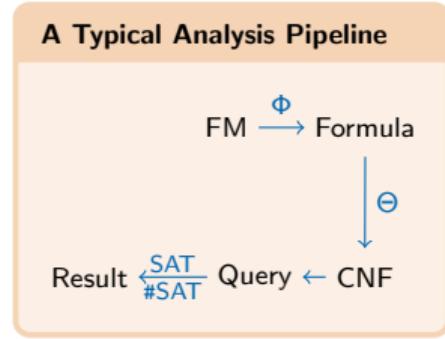
- A sample covering all **t-wise interactions**? [ref]

- How large** is the configuration space? [ref]

- How has the configuration space **evolved**? [ref]

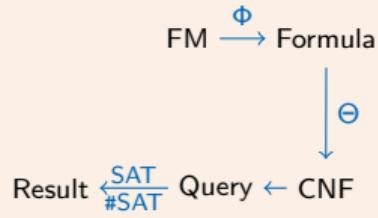
... which often rely on **SAT solving** (et al.)!

Recap: Product-Line and Feature-Model Analysis

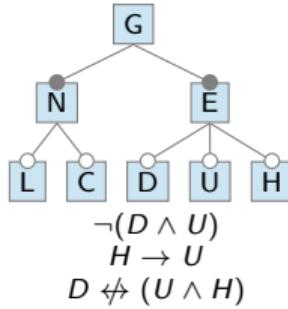


Recap: Product-Line and Feature-Model Analysis

A Typical Analysis Pipeline

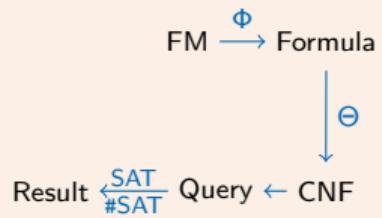


A Feature Model FM

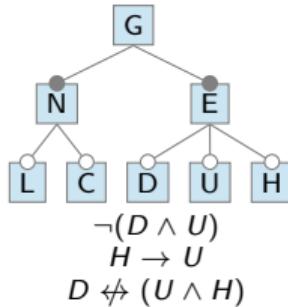


Recap: Product-Line and Feature-Model Analysis

A Typical Analysis Pipeline



A Feature Model FM

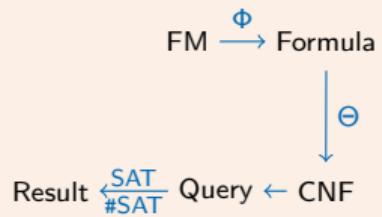


As a Formula $\Phi(FM)$

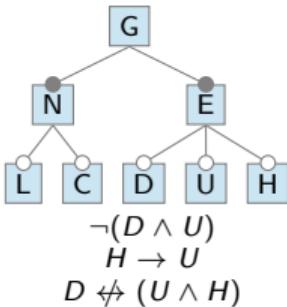
$$\begin{aligned} & G \\ & \wedge (N \leftrightarrow G) \wedge (E \leftrightarrow G) \\ & \wedge ((L \vee C) \rightarrow N) \\ & \wedge ((D \vee U \vee H) \rightarrow E) \\ & \wedge \neg(D \wedge U) \wedge (H \rightarrow U) \\ & \wedge (D \nleftrightarrow (U \wedge H)) \end{aligned}$$

Recap: Product-Line and Feature-Model Analysis

A Typical Analysis Pipeline



A Feature Model FM



As a Formula $\Phi(FM)$

$$\begin{aligned} &G \\ &\wedge (N \leftrightarrow G) \wedge (E \leftrightarrow G) \\ &\wedge ((L \vee C) \rightarrow N) \\ &\wedge ((D \vee U \vee H) \rightarrow E) \\ &\wedge \neg(D \wedge U) \wedge (H \rightarrow U) \\ &\wedge (D \nleftrightarrow (U \wedge H)) \end{aligned}$$

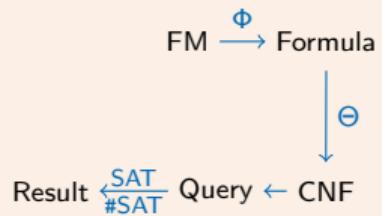
$\downarrow \Theta$

As a CNF $\Theta(\Phi(FM))$

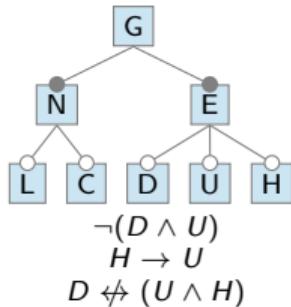
$$\begin{aligned} &\{\{G\}, \{\neg N, G\}, \{N, \neg G\}, \\ &\{\neg E, G\}, \{E, \neg G\}, \{\neg L, N\}, \\ &\{\neg C, N\}, \{\neg D, E\}, \{\neg U, E\}, \\ &\{\neg H, E\}, \{\neg D, \neg U\}, \{\neg H, U\}, \\ &\{\{D, U\}, \{D, H\}, \{\neg D, \neg U, \neg H\}\} \} \end{aligned}$$

Recap: Product-Line and Feature-Model Analysis

A Typical Analysis Pipeline



A Feature Model FM



As a Formula $\Phi(FM)$

$$\begin{aligned}
 & G \\
 & \wedge (N \leftrightarrow G) \wedge (E \leftrightarrow G) \\
 & \wedge ((L \vee C) \rightarrow N) \\
 & \wedge ((D \vee U \vee H) \rightarrow E) \\
 & \wedge \neg(D \wedge U) \wedge (H \rightarrow U) \\
 & \wedge (D \nleftrightarrow (U \wedge H))
 \end{aligned}$$

$\downarrow \Theta$

Core Features

$$\{G, N, E\}$$

←

Core Feature F ?

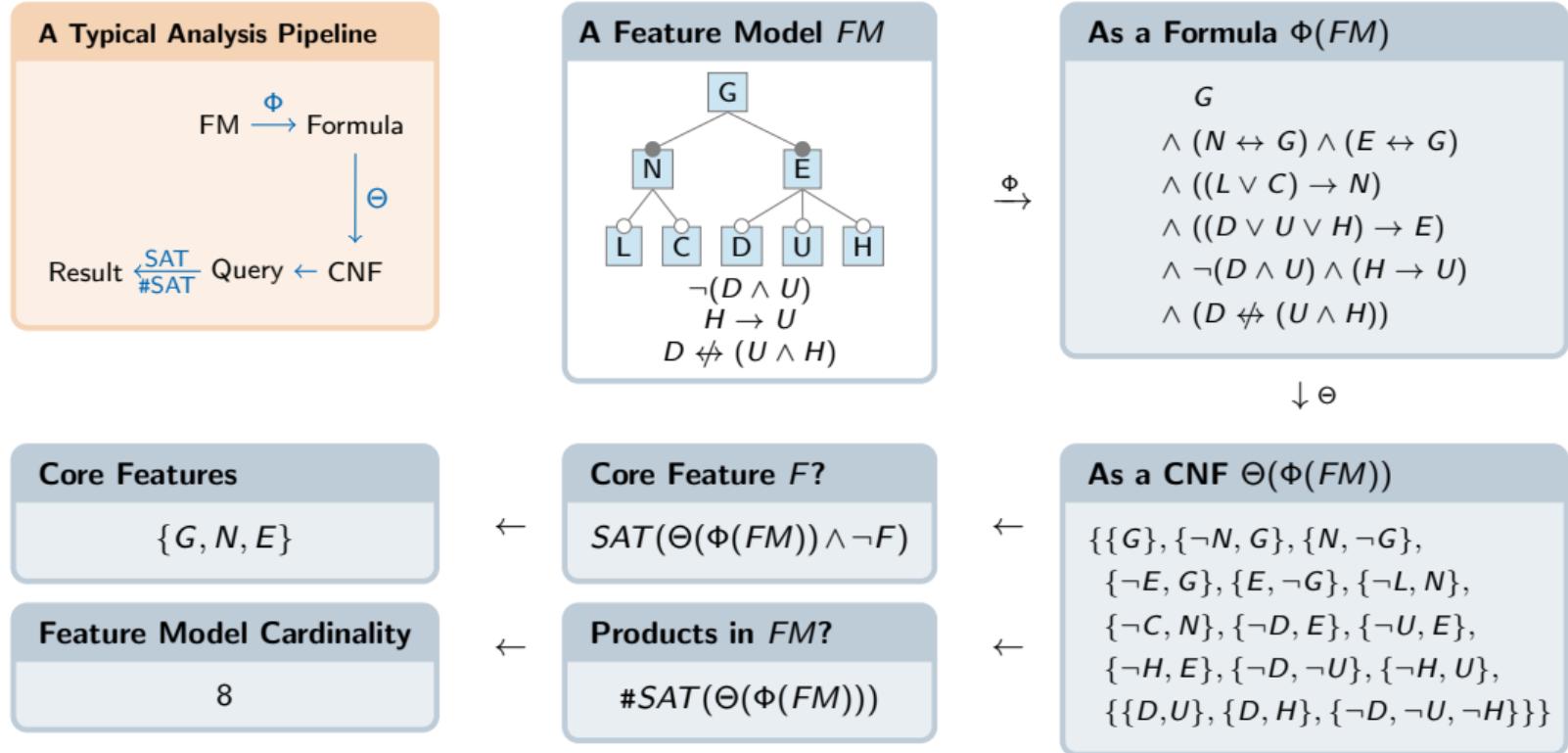
$$SAT(\Theta(\Phi(FM)) \wedge \neg F)$$

←

As a CNF $\Theta(\Phi(FM))$

$$\begin{aligned}
 & \{\{G\}, \{\neg N, G\}, \{N, \neg G\}, \\
 & \{\neg E, G\}, \{E, \neg G\}, \{\neg L, N\}, \\
 & \{\neg C, N\}, \{\neg D, E\}, \{\neg U, E\}, \\
 & \{\neg H, E\}, \{\neg D, \neg U\}, \{\neg H, U\}, \\
 & \{\{D, U\}, \{D, H\}, \{\neg D, \neg U, \neg H\}\}
 \end{aligned}$$

Recap: Product-Line and Feature-Model Analysis



SAT-Based Analysis of Feature Models is Easy

SAT-Based Analysis of Feature Models is Easy

SAT-based Analysis of Feature Models is Easy

Marcilio Mendonca
University of Waterloo, Canada
marcilio@csg.uwaterloo.ca

Andrzej Wąsowski
IT University of Copenhagen, Denmark
wasowski@itu.dk

Krzysztof Czarnecki
University of Waterloo, Canada
kczarnec@gsd.uwaterloo.ca

SAT-Based Analysis of Feature Models is Easy

SAT-based Analysis of Feature Models is Easy

Marcilio Mendonca
University of Waterloo, Canada
marcilio@csg.uwaterloo.ca

Andrzej Wąsowski
IT University of Copenhagen, Denmark
wasowski@itu.dk

Krzysztof Czarnecki
University of Waterloo, Canada
kczarnec@gsd.uwaterloo.ca

SAT-based Analysis of Large Real-world Feature Models is Easy

Jia Hui Liang
University of Waterloo,
Canada

Vijay Ganesh
University of Waterloo,
Canada

Krzysztof Czarnecki
University of Waterloo,
Canada

Venkatesh Raman
Institute of Mathematical
Sciences, India

SAT-Based Analysis of Feature Models is Easy

SAT-based Analysis of Feature Models is Easy

Marcilio Mendonca
University of Waterloo, Canada
marcilio@csg.uwaterloo.ca

Andrzej Wąsowski
IT University of Copenhagen, Denmark
wasowski@itu.dk

Krzysztof Czarnecki
University of Waterloo, Canada
kczarnev@gsd.uwaterloo.ca

“easy”

as in

**“performs much better than expected
despite being NP-complete”**

(because no phase transition is observed on typical
feature models)

SAT-based Analysis of Large Real-world Feature Models is Easy

Jia Hui Liang
University of Waterloo,
Canada

Vijay Ganesh
University of Waterloo,
Canada

Krzysztof Czarnecki
University of Waterloo,
Canada

Venkatesh Raman
Institute of Mathematical
Sciences, India

SAT-Based Analysis of Feature Models is Easy

SAT-based Analysis of Feature Models is Easy

Marcilio Mendonca
University of Waterloo, Canada
marcilio@csg.uwaterloo.ca

Andrzej Wąsowski
IT University of Copenhagen, Denmark
wasowski@itu.dk

Krzysztof Czarnecki
University of Waterloo, Canada
kczarnev@gsd.uwaterloo.ca

“easy”

as in

“performs much better than expected
despite being NP-complete”

(because no phase transition is observed on typical
feature models)

SAT-based Analysis of Large Real-world Feature Models is Easy

Jia Hui Liang
University of Waterloo,
Canada

Krzysztof Czarnecki
University of Waterloo,
Canada

Vijay Ganesh
University of Waterloo,
Canada

Venkatesh Raman
Institute of Mathematical
Sciences, India

Yes, But . . .

- easy = **fast**?
 - what about **pre-solving steps**?
 - what about **repeated solver calls**?
- are **non-SAT analyses** also easy?
- are all feature models **equally** easy?

SAT-Based Analysis of Feature Models is Easy*

SAT-based Analysis of Feature Models is Easy

Marcilio Mendonca
University of Waterloo, Canada
marcilio@csg.uwaterloo.ca

Andrzej Wąsowski
IT University of Copenhagen, Denmark
wasowski@itu.dk

Krzysztof Czarnecki
University of Waterloo, Canada
kczarnev@gsd.uwaterloo.ca

SAT-based Analysis of Large Real-world Feature Models is Easy

Jia Hui Liang
University of Waterloo,
Canada

Krzysztof Czarnecki
University of Waterloo,
Canada

Vijay Ganesh
University of Waterloo,
Canada

Venkatesh Raman
Institute of Mathematical
Sciences, India

“easy”

as in

“performs much better than expected
despite being NP-complete”

(because no phase transition is observed on typical
feature models)

Yes, But . . .

- 👎 easy = **fast**?
 - what about **pre-solving steps**?
 - what about **repeated solver calls**?
- 👎 are **non-SAT analyses** also easy?
- 👎 are all feature models **equally** easy?

*on most instances, for most purposes

How Bad is It?

How Bad is It?

How Bad is It?

(REDACTED)

- Linux kernel
 - number of configurations **unknown** past 2007
 - family-based type checking **infeasible** [ref]
 - as is uniform random sampling, slicing, diffing, ...
 - even core/dead features are **challenging**
- same for Freetz-NG, Buildroot, Automotive02, ...

(REDACTED)

How Bad is It?

- Linux kernel
 - number of configurations **unknown** past 2007
 - family-based type checking **infeasible**
 - as is uniform random sampling, slicing, diffing, ...
 - even core/dead features are **challenging**
- same for Freetz-NG, Buildroot, Automotive02, ...
 - solvers get **faster** – but feature models grow **more complex!**
 - ⇒ need better understanding of **feature-model complexity**
 - ⇒ perspective shift to **instance-based meta-analysis**

- Linux kernel

- number of configurations **unknown** past 2007
 - family-based type checking **infeasible**
 - as is uniform random sampling, slicing, diffing, ...
 - even core/dead features are **challenging**
 - same for Freetz-NG, Buildroot, Automotive02, ...

solvers get **faster** – but feature models grow **more complex**!

- ⇒ need better understanding of **feature-model complexity**
 - ⇒ perspective shift to **instance-based meta-analysis**

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions**
about feature-model analyses

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering questions
about feature-model analyses

⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering questions
about feature-model analyses

⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

... to Instance-Based Meta-Analysis

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

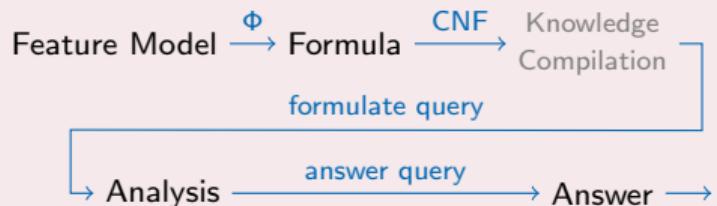
[ref, ref]

- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis



Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

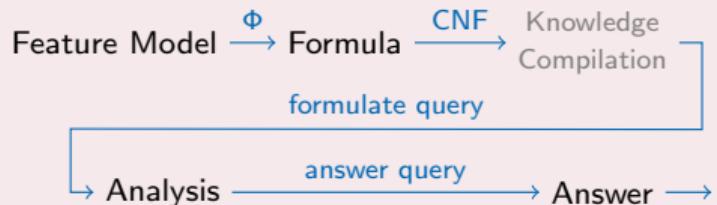
- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness



Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

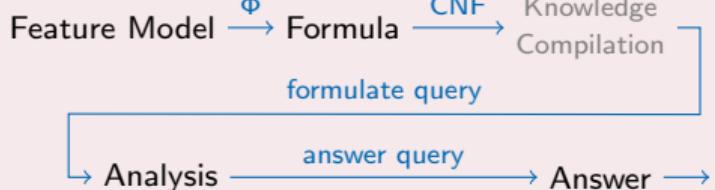
... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness

extractor
non-Boolean variability
preprocessing



Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

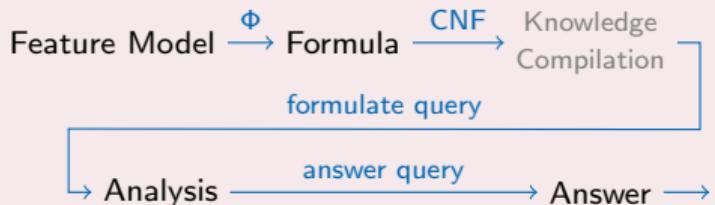
- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness extractor
non-Boolean variability
preprocessing BDD
d-DNNF



Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

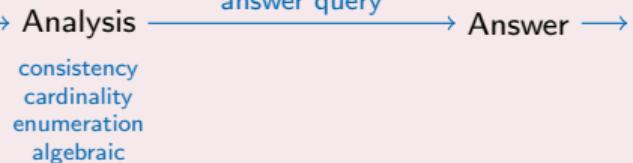
- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness extractor
non-Boolean variability
preprocessing BDD
d-DNNF



Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

- "Is SAT-based analysis of feature models easy?"
- "Is SAT-based analysis of large real-world feature models easy?"

... to Instance-Based Meta-Analysis

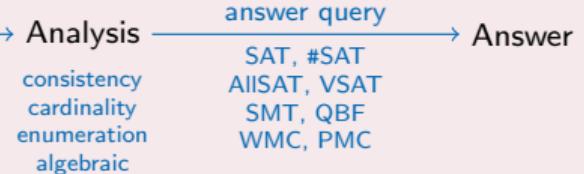
- "How much time does analysis X need on feature model Y when using solver Z?"
- "Which algorithm is most memory-efficient for computing X on Y?"

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness extractor
non-Boolean variability
preprocessing BDD
d-DNNF



formulate query



Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

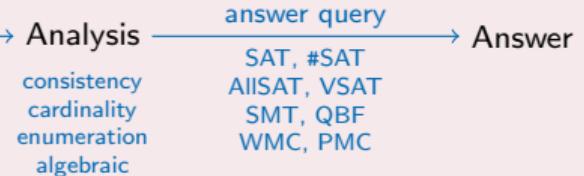
- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness extractor
non-Boolean variability
preprocessing BDD
d-DNNF



formulate query



- + **parametrization:** extractor, compiler, analysis, solver
- + **prior information:** incremental analysis, interfaces
- + **execution environment:** CPU, RAM, deep variability

Asking Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of **asking** and answering **questions about** feature-model analyses

- ⇒ ask for **(non-)functional requirements**
(e.g., correctness, runtime, memory, energy)

From Class-Based Meta-Analysis ...

[ref, ref]

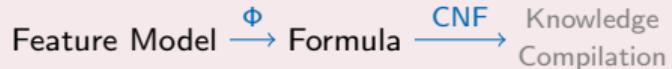
- “Is SAT-based analysis of feature models easy?”
- “Is SAT-based analysis of large real-world feature models easy?”

... to Instance-Based Meta-Analysis

- “How much time does analysis X need on feature model Y when using solver Z?”
- “Which algorithm is most memory-efficient for computing X on Y?”

Influence Factors for Feature-Model Analysis

origin, domain
size
expressiveness extractor
non-Boolean variability
preprocessing BDD
d-DNNF



formulate query

Analysis answer query
consistency SAT, #SAT
cardinality AIISAT, VSAT
enumeration SMT, QBF
algebraic WMC, PMC

- + **parametrization:** extractor, compiler, analysis, solver
- + **prior information:** incremental analysis, interfaces
- + **execution environment:** CPU, RAM, deep variability

for practical analysis tasks, there are many **analysis plans**

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions
about feature-model analyses

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions
about feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the
question (either exactly or an estimate)

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions
about feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

Choosing Criteria & an Algorithm

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Discussion & Outlook

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Discussion & Outlook

- what meta-analysis questions are **worth asking**?

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Discussion & Outlook

- what meta-analysis questions are **worth asking**?
- which factors are relevant, how do they **interact**?

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Discussion & Outlook

- what meta-analysis questions are **worth asking**?
- which factors are relevant, how do they **interact**?
- is feature-model complexity **intrinsic**?

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Discussion & Outlook

- what meta-analysis questions are **worth asking**?
- which factors are relevant, how do they **interact**?
- is feature-model complexity **intrinsic**?
- when do advanced techniques (knowledge computation, incremental analysis, ...) **pay off**?

Answering Meta-Analysis Questions

Feature-Model Meta-Analysis

the practice of asking and **answering** questions **about** feature-model analyses

⇒ define **criteria** and an **algorithm** to answer the question (either exactly or an estimate)

Surrogate Metrics for Avoiding the Computation

- **syntactic** (e.g., number of features, variables, constraints, clauses, literals; constraint size, density)
- **semantic** (e.g., phase transition, community structure, self-similarity)

Choosing Criteria & an Algorithm

- **binary** (e.g., “yes/no” for phase transition)
- **linear** (e.g., number off features \sim analysis time)
- **machine learning** techniques

- syntactic metrics only give rough estimates
- semantic metrics probably better, but NP-hard
⇒ try approximating
- ML is sensitive to small changes in the input

Discussion & Outlook

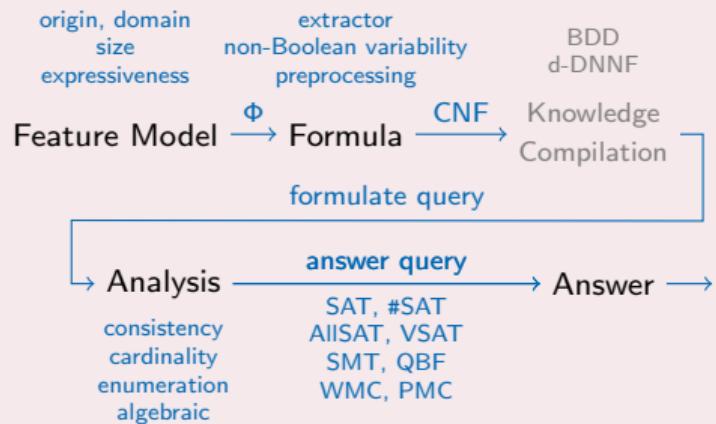
- what meta-analysis questions are **worth asking**?
- which factors are relevant, how do they **interact**?
- is feature-model complexity **intrinsic**?
- when do advanced techniques (knowledge computation, incremental analysis, ...) **pay off**?
- long-term goal: a **meta-analyzer** that finds the best analysis plan for a given analysis task
 - reify analyses as first-class objects
 - inspired by portfolio solving and relational query optimization

Conclusion

Feature-Model Meta-Analysis

the practice of **asking** and **answering** questions
about feature-model analyses

Influence Factors for Feature-Model Analysis



Your opinion?

Does feature-model complexity matter for your work?
Are you doing meta-analysis?
How would you answer a meta-analysis question?



[/SoftVarE-Group/Papers](#)

[/SoftVarE-Group/Slides](#)