Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Pooya Saadatpanah, Michalis Famelis, Jan Gorzny, Nathan Robinson, Marsha Chechik, Rick Salay

University of Toronto

September 30th, 2012

MoDeVVa’12
Uncertainty in software modeling

- Uncertainty: pervasive in MDE

- Models with uncertainty:
  - Represent choice among many possibilities
  - Can be refined to many different classical models

- Our goal:
  Handle models with uncertainty in MDE without having to remove it [MoDeVVa’11].
Existing Work

Reasoning [ICSE'12]

Position [MoDeVe'11]

Requirements [RE'12]

Refinement [VOLT12]

Partiality Types [FASE'12]

Transformation [MiSE'12]
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction
Designer Uncertainty
Modeling Uncertainty
Property Checking Process
Verification Technologies
Experiments Results
Conclusion

In This Paper

RQ: What is the most efficient formalism for checking properties of models with Uncertainty?

Experiment with alternative verification technologies.
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Outline

Introduction

What is uncertainty?

How to represent uncertainty with partial models (MAVO).

Process for checking properties

Alternative verification technologies

Experiments

Results

Conclusion
Introduction to Uncertainty

What the designer **knows**.
Introduction to Uncertainty

What the designer does not know.

- Vehicle
- LandVehicle
- numDoors

Unknown if or how many LandVehicles will be there.
What the designer does not know.
Introduction to Uncertainty

What the designer does not know.
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Introduction
Designer
Uncertainty
Modeling Uncertainty
Property Checking
Process
Verification Technologies
Experiments
Results
Conclusion

Uncertainty: a Set of Possible Refinements.

If we remove all uncertainty, we have a concrete refinement.
Modeling Uncertainty with Partial Models

Explicating uncertainty in a partial model.

Unknown if or how many LandVehicles will be subclass of Vehicle

Unknown if or how many LandVehicles will be there

Vehicle

Unknown which class will contain this attribute

LandVehicle

numDoors
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction
Designer
Uncertainty
Modeling Uncertainty
Property Checking
Process
Verification Technologies
Experiments
Results
Conclusion

Modeling Uncertainty with Partial Models

Explicating uncertainty in a partial model.

In a refinement, a May element is optional.
Modeling Uncertainty with Partial Models

Explicating uncertainty in a partial model.

In a refinement, a May element is optional.

In a refinement, a Set element can be multiplied to many copies.
Explicating uncertainty in a partial model.

In a refinement, a May element is optional.
In a refinement, a Set element can be multiplied to many copies.
In a refinement, a Variable element can be unified with some other.
Modeling Uncertainty with Partial Models

Explicating uncertainty in a partial model.

In a refinement, a May element is optional.
In a refinement, a Set element can be multiplied to many copies.
In a refinement, a Variable element can be unified with some other.
In a refinement, an Open world model can be expanded with some other elements.
Partial models: Syntactic annotations of the points of uncertainty.
Outline

Introduction

What is uncertainty?

How to represent uncertainty with partial models (MAVO).

Process for checking properties

Alternative verification technologies

Experiments

Results

Conclusion
Property Checking in Partial Models

![Diagram showing the process of property checking in partial models.]

- Partial Model
- Verification Tool
- Result
- Property

Introduction
Designer Uncertainty
Modeling Uncertainty
Property Checking Process
Verification Technologies
Experiments
Results
Conclusion
Property Checking in Partial Models

ICSE'12

- Partial Model
- May Model
- Verification Tool
- SAT
- Property
- Result
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction

Designer Uncertainty

Modeling Uncertainty

Property Checking Process

Verification Technologies

Experiments

Results

Conclusion

Property Checking in Partial Models

ICSE'12 FASE'12

Partial Model

May Model

MAVO Model

Verification Tool

SAT

Alloy

Result
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction
Designer Uncertainty
Modeling Uncertainty
Property Checking Process
Verification Technologies
Experiments
Results
Conclusion

Property Checking in Partial Models

ICSE'12  FASE'12  MoDDeVa'12

Partial Model
- May Model
- MAVO Model
- MAV Model

Property

Verification Tool
- SAT
- Alloy
- Alloy, CSP
- SMT, ASP

Result
Verification Technologies I

• Alloy
  • Lightweight formal methods
  • Model finder based on SAT
  • First order logic specifications expressed in relational logic
  • Grounded to CNF representation
  • Finds counter examples

• Constraint Satisfaction Problem (Minizinc/Flatzinc)
  • Assign value to variables to satisfy all constraints
  • Constraint modeling language
  • Easily translatable to the form required by other CSP solvers
Verification Technologies II

- Satisfiability Modulo Theory (Z3)
  - Constraint satisfaction search with richer theories
  - Theorem prover
  - Check the satisfiability logical formulas

- Answer Set Programming (Clingo=Gringo+Clasp)
  - Answer set solvers
  - Conflict-driven nogood learning
  - Normal logic programs
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Outline

Introduction

What is uncertainty?

How to represent uncertainty with partial models (MAVO).

Process for checking properties

Alternative verification technologies

Experiments

Results

Conclusion
Experimental Setup
Experimental Setup

- Experimental Parameters
- Model
- Relational Encoding
- Verification Tool (Alloy, CSP, SMT, ASP)
- Property
- Result
Random Input Generation

- Meta-model: directed graphs
  - Minimal meta-model
  - A few constraints
  - Most difficult one for solvers

- Randomly decorated with MAVO annotations.

- Parameters are based on real case studies.
  - Graph density
  - Percentage of MAVO annotated elements
  - Percentages of M-, S- and V-annotated elements

- 3 Model Size: Small, Medium, Large, X-Large
Experimental Setup

Experimental Parameters

Random Model Generator

Model

Relational Encoding

Verification Tool

Result

Property

Alloy, CSP
SMT, ASP
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction

Designer

Uncertainty

Modeling

Uncertainty

Property

Checking

Process

Verification

Technologies

Experiments

Results

Conclusion

Relational Encoding I

- Relational Algebra used in DBMSs
- Directly translatable into the different formalisms
- Intermediate representation
  - FOL semantics of MAVO
  - Reasoning formalisms
- Meaningful comparison
  - Most efficient encoding in each formalism: impossible!
  - Solution: common encoding
(Very) high level overview.

- The metamodel encoded as a schema.
- Partial model FOL semantics encoded as constraints over the schema.
- Creating a concrete refinement populates the database.
Experimental Setup
Translation To Formalisms

- **Alloy**
  - Relations: Alloy signatures
  - Instances: Atoms
  - MAVO constraints: quantified predicates over signatures
  - Bound is required

- **CSP**
  - Relations: Finite set of Integers
  - Instances: Integers
  - MAVO constraints: cardinality and intersection of sets
  - Bound is required

- **SMT**
  - Relations: Uninterpreted boolean functions
  - Instances: Abstract values
  - MAVO constraints: Quantified logic over truth table of functions

- **ASP**
  - Program rules for both instances and relations
  - Bound is required
Experimental Setup

Experimental Parameters

Random Model Generator → Model → Relational Encoding → Verification Tool

Property

Alloy, CSP, SMT, ASP → Result
Properties Checked

- Inspired by real metamodel constraints.
- No transitive closure, since it is expensive to check.

\begin{tabular}{|l|}
\hline
\textbf{P1}: There exists a node with a self-loop. \\
\textbf{P2}: All nodes have outgoing edges. \\
\textbf{P3}: All nodes have outgoing or incoming edges. \\
\textbf{P4}: For all pairs of nodes $\langle n_1, n_2 \rangle$ there exists at most one edge $e$ such that $n_1 \rightarrow n_2$. \\
\textbf{P5}: For every pair of nodes $\langle n_1, n_2 \rangle$, $n_1 \neq n_2$ there exist two edges $\langle e_1, e_2 \rangle$ such that $n_1 \rightarrow n_2$ and $n_2 \rightarrow n_1$. \\
\hline
\end{tabular}
Experimental Setup
Experimental Parameters

- **Bound (2, 4, 6)**
  - Solvers (except SMT) use bound for grounding expressions to atoms.
  - How many times can an 'S'-annotated element be replicated in a refinement.

- **Repetitions**
  - 5 times

- **Cutoff time/memory**
  - less than 10 minutes
  - less than 5 gigabyte
  - otherwise timeout

- **What we measure**
  - How long does it take for each solver to return an answer
  - A score out of 1200
  - if timeout : zero!
Findings

![Graph showing the effectiveness of reasoning formalisms for partial models]

- **Bound = 2**
- **SMT**, **ASP**, **Alloy**, **CSP**

**Axes:**
- **X-axis:** Model size (Small, Medium, Large, X-Large)
- **Y-axis:** Timing score

Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay
Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction
Designer
Uncertainty
Modeling
Uncertainty
Property
Checking
Process
Verification
Technologies
Experiments
Results
Conclusion

Findings

Bound = 4

Timing score

SMT
ASP
Alloy
CSP

Model size
Small Medium Large X-Large
Findings

Comparing the Effectiveness of Reasoning Formalisms for Partial Models

Saadatpanah, Famelis, Gorzny, Robinson, Chechik, Salay

Introduction
Designer
Uncertainty
Modeling
Uncertainty
Property
Checking
Process
Verification
Technologies
Experiments
Results
Conclusion
SMT the champion?

- Unaffected by bounds!
- Works at higher level of abstraction (Theory of uninterpreted functions.)
- Unaffected by expensive grounding phase.

Caveat:
- SMT can theoretically return “I don’t know”.
- (However: we didn’t observe that.’)
Threats to Validity

Randomly generated graphs.
- Tuned the generator with realistic graph properties.
- Values of graph properties from case studies.

Fairness of comparisons.
- Common encoding to level the field.

Choice of specific reasoning engines
- When available: winners of competitions.
- CSP: most convenient input language.
Research Question

What is the most efficient formalism for verifying models containing uncertainty?
Other results:

- Framework for running experiments (now full tool support).
- Random generator for arbitrary type graphs.
- Relational encoding.
- Translations of the RA encoding to different formalisms.
Future Work

- Implement symmetry breaking in the SMT encoding.

- Experiment with properties that require transitive closure.

- Experiment with partial models containing OW.
Questions?
“Partial Models: A Position Paper”.
In Proceedings of MoDeVVa’11, pages 1–6.

“Partial Models: Towards Modeling and Reasoning with Uncertainty”.
In Proceedings of ICSE’12.

“Language Independent Refinement using Partial Modeling”.
In Proceedings of FASE’12.