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

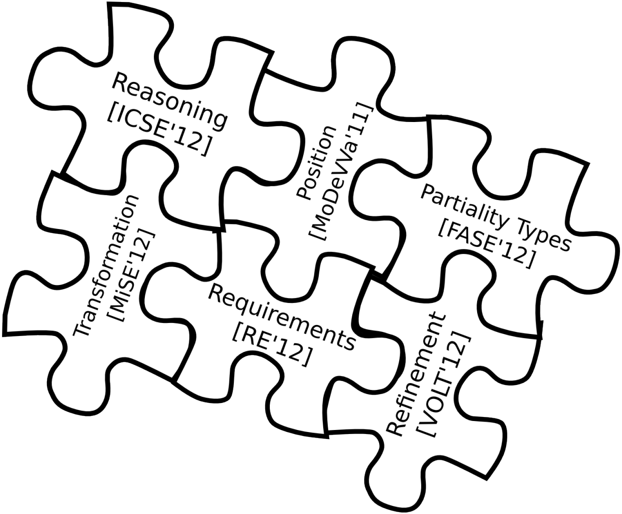
MoDeVVa'12

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



In This Paper

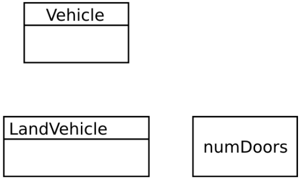
RQ: What is the most efficient formalism for checking properties of models with Uncertainty?

Experiment with alternative verification technologies.

Outline

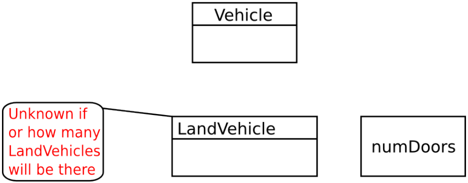
Introduction to Uncertainty

What the designer **knows.**



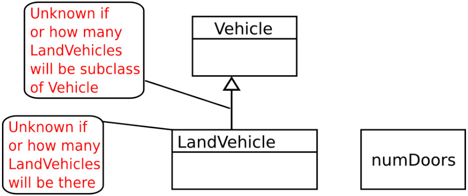
Introduction to Uncertainty

What the designer **does not know**.



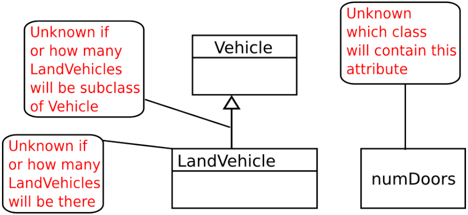
Introduction to Uncertainty

What the designer **does not know**.

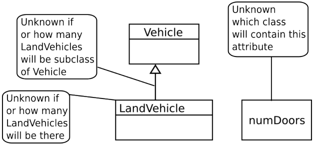


Introduction to Uncertainty

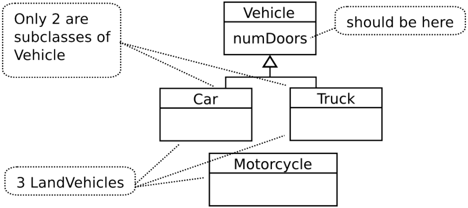
What the designer **does not know**.



Uncertainty: a Set of Possible Refinements.

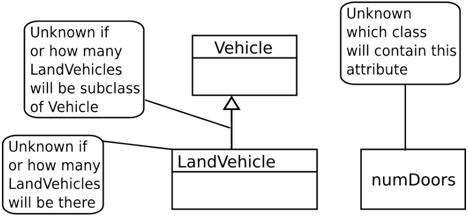


If we remove all uncertainty, we have a concrete *refinement*.



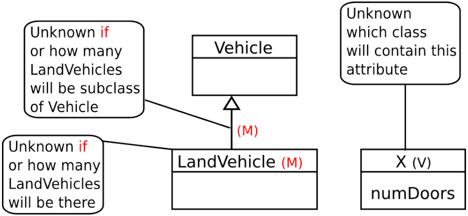
Modeling Uncertainty with Partial Models

Explicating uncertainty in a partial model.



Modeling Uncertainty with Partial Models

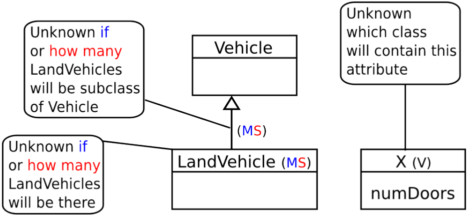
Explicating uncertainty in a partial model.



In a refinement, a May element is optional.

Modeling Uncertainty with Partial Models

Explicating uncertainty in a partial model.

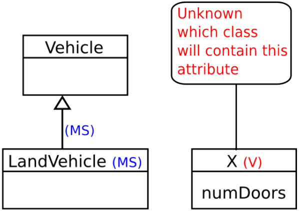


In a refinement, a May element is optional.

In a refinement, a Set element can be multiplied to many copies.

Modeling Uncertainty with Partial Models

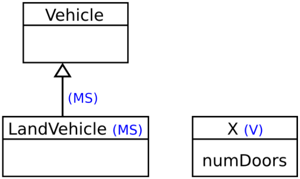
Explicating uncertainty in a partial model.



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.

Modeling Uncertainty with Partial Models

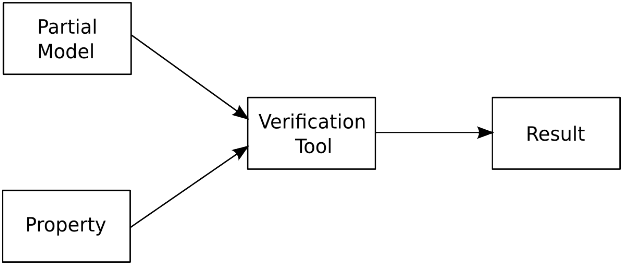
Explicating uncertainty in a partial model.



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.

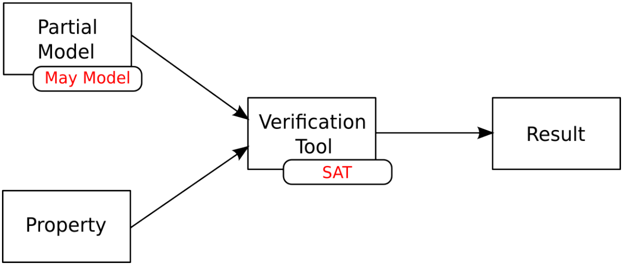
Outline

Property Checking in Partial Models

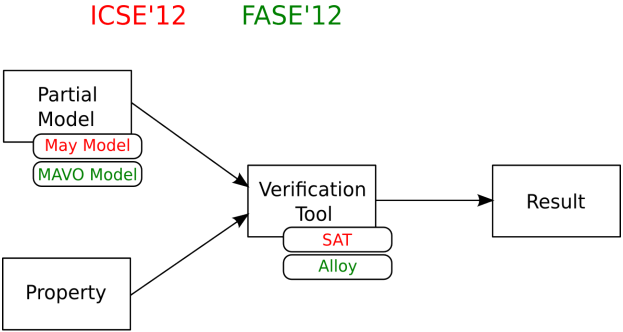


Property Checking in Partial Models

ICSE'12

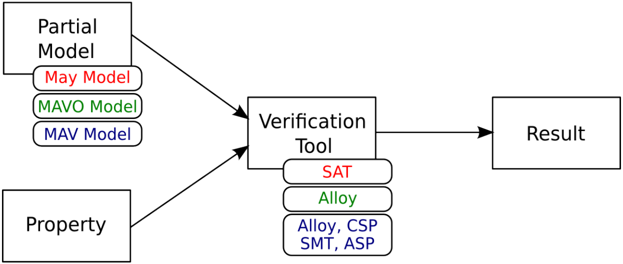


Property Checking in Partial Models



Property Checking in Partial Models

ICSE'12 FASE'12 MoDDeVa'12



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

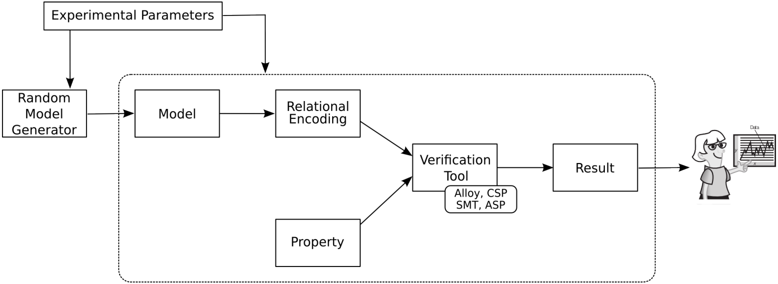
Verification Technologies II

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

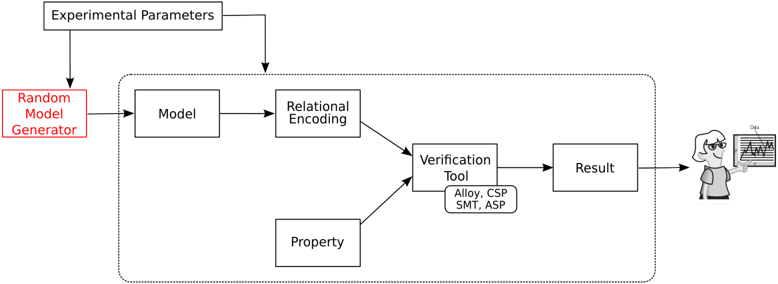
- Answer Set Programming (Clingo=Gringo+Clasp)
  - Answer set solvers
  - Conflict-driven nogood learning
  - Normal logic programs

Outline

Experimental Setup



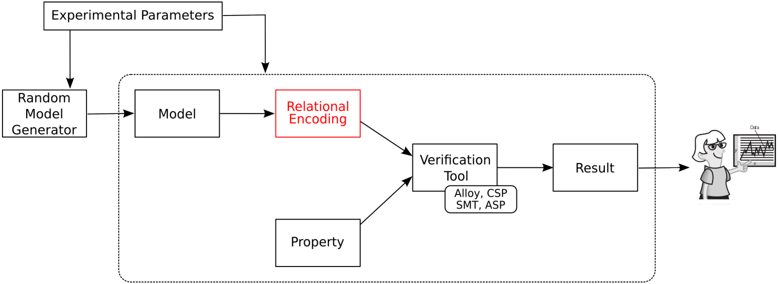
Experimental Setup



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



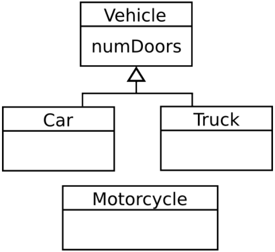
Relational Encoding I

- 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

Relational Encoding II

(Very) high level overview.

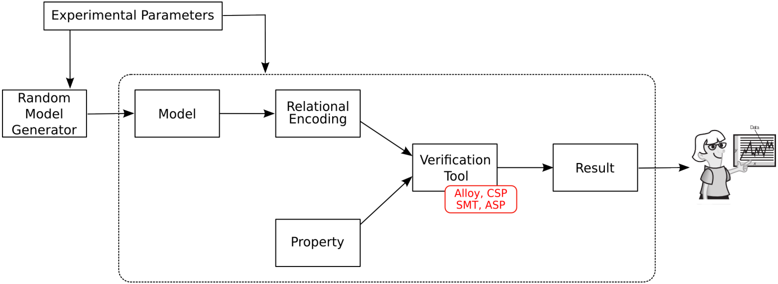
- The metamodel encoded as a schema.
- Partial model FOL semantics encoded as constraints over the schema.
- Creating a concrete refinement populates the database.



CLASS:

MAVO element	Instance element
`Vehicle`	*Vehicle*
`LandVehicle`	*LandVehicle/Car*
`LandVehicle`	*LandVehicle/Truck*
`LandVehicle`	*LandVehicle/Moto*
X	*Vehicle*

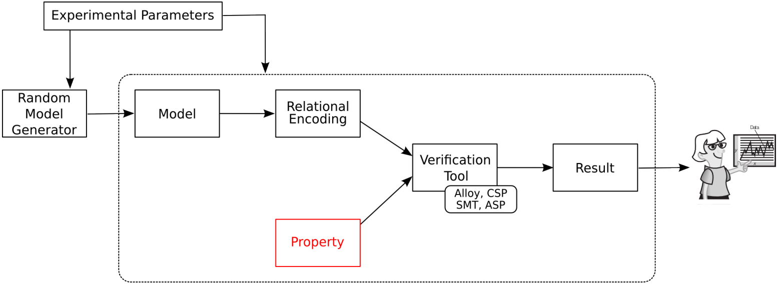
Experimental Setup



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



Properties Checked

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

P1: There exists a node with a self-loop.

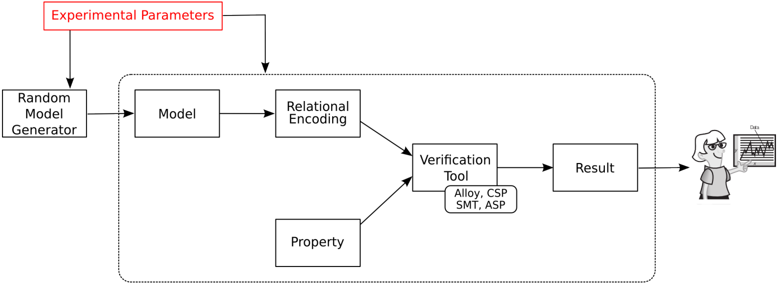
P2: All nodes have outgoing edges.

P3: All nodes have outgoing or incoming edges.

P4: For all pairs of nodes $\langle n_1, n_2 \rangle$ there exists at most one edge e such that $n_1 \xrightarrow{e} n_2$

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 $n_1 \xrightarrow{e_1} n_2$ and $n_2 \xrightarrow{e_2} n_1$.

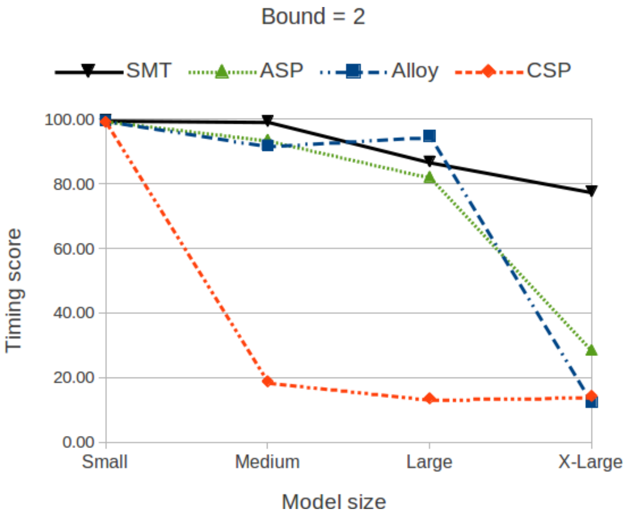
Experimental Setup



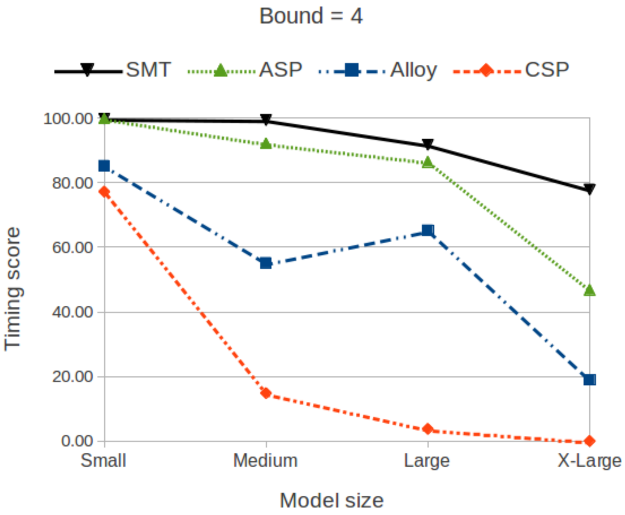
Experimental Parameters

- 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!

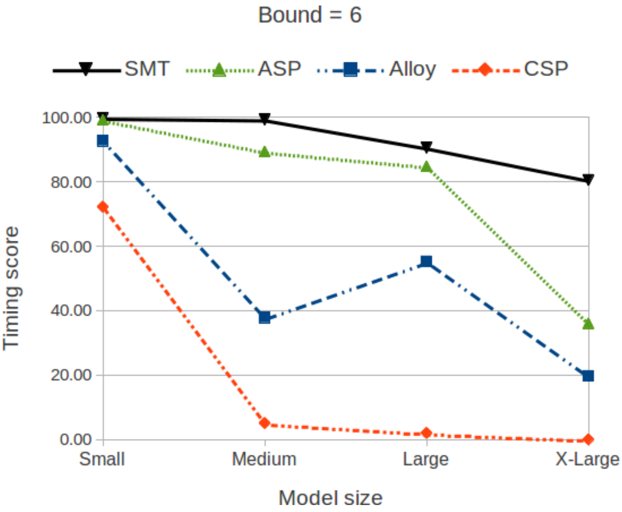
Findings


Bound = 2
SMT
ASP
Alloy
CSP
Timing score
100.00
80.00
60.00
40.00
20.00
0.00
Small
Medium
Large
X-Large
Model size

Findings


Bound = 4
SMT
ASP
Alloy
CSP
100.00
80.00
60.00
40.00
20.00
0.00
Timing score
Small
Medium
Large
X-Large
Model size

Findings


Bound = 6
SMT
ASP
Alloy
CSP
100.00
80.00
60.00
40.00
20.00
0.00
Timing score
Small
Medium
Large
X-Large
Model size

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.

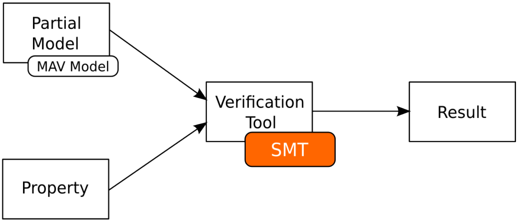
Choice of specific reasoning engines

- When available: winners of competitions.
- CSP: most convenient input language.

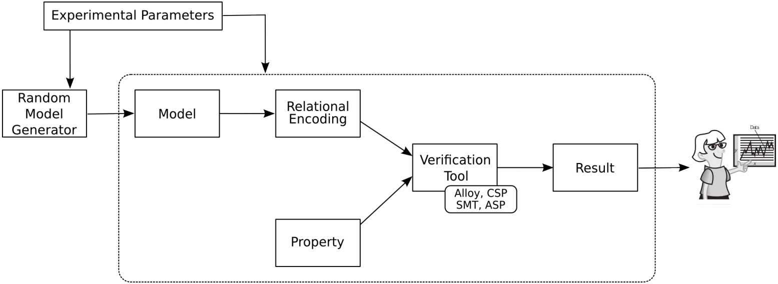
Summary

Research Question

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



Summary



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?

Bibliography I

Famelis, M., Ben-David, S., Chechik, M., and Salay, R. (2011).
"Partial Models: A Position Paper".
In *Proceedings of MoDeVVa'11*, pages 1–6.

Famelis, M., Chechik, M., and Salay, R. (2012).
"Partial Models: Towards Modeling and Reasoning with Uncertainty".
In *Proceedings of ICSE'12*.

Salay, R., Famelis, M., and Chechik, M. (2012).
"Language Independent Refinement using Partial Modeling".
In *Proceedings of FASE'12*.