Augmenting Model-Based Instantiation with Fast Enumeration in SMT

Lydia Kondylidou[1] Andrew Reynolds[2] Jasmin Blanchette[1]

[1]Ludwig Maximilian University of Munich

[2]The University of Iowa

What is an SMT solver?

Satisfiability modulo theories (SMT) solvers . . .

- combine a SAT solver with decision procedures for interpreted theories
- support first-order logic and partly higher-order logic

What is an SMT solver?

- Architecture:

$$\text{SMT loop} \begin{cases} \text{Quantifier-free solver} \begin{cases} \text{SAT solver} \\ \text{Theory solvers} \end{cases} \\ \\ \textbf{Instantiation module} \end{cases}$$

- Quantifier-free solver enumerates assignments $E \cup Q$
  - E is a set of ground (i.e. variable-free) atoms
  - Q is a set of quantified atoms
- Instantiation module generates instances of Q

Example — SMT with quantifiers

$$F = (\forall x.\ \mathrm{p}\ x) \wedge \neg\, \mathrm{p}\ \mathrm{a}$$

1. $E = \{\neg\, \mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall x.\ \mathrm{p}\ x\}$

Example — SMT with quantifiers

$$F = (\forall x.\ \mathrm{p}\ x) \land \neg\, \mathrm{p\ a}$$

1. $E = \{\neg\, \mathrm{p\ a}\}$ and $Q = \{\forall x.\ \mathrm{p}\ x\}$
2. Model from SAT solver: atom p a is false and atom $\forall x.\ \mathrm{p}\ x$ is true

Example — SMT with quantifiers

$$F = (\forall x.\ \mathrm{p}\ x) \land \neg\, \mathrm{p\ a}$$

1. $E = \{\neg\, \mathrm{p\ a}\}$ and $Q = \{\forall x.\ \mathrm{p}\ x\}$
2. Model from SAT solver: atom p a is false and atom $\forall x.\ \mathrm{p}\ x$ is true
3. **Instantiation lemma**: $(\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p\ a}$

Example — SMT with quantifiers

$$F = (\forall x.\ \mathrm{p}\ x) \wedge \neg\, \mathrm{p}\ \mathrm{a}$$

1. $E = \{\neg\, \mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall x.\ \mathrm{p}\ x\}$
2. Model from SAT solver: atom $\mathrm{p}\ \mathrm{a}$ is false and atom $\forall x.\ \mathrm{p}\ x$ is true
3. **Instantiation lemma**: $(\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p}\ \mathrm{a}$
4. $F \leftarrow F \wedge ((\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p}\ \mathrm{a})$

Example — SMT with quantifiers

$$F = (\forall x.\ \mathrm{p}\ x) \land \neg\, \mathrm{p\ a}$$

1. $E = \{\neg\, \mathrm{p\ a}\}$ and $Q = \{\forall x.\ \mathrm{p}\ x\}$
2. Model from SAT solver: atom $\mathrm{p\ a}$ is false and atom $\forall x.\ \mathrm{p}\ x$ is true
3. **Instantiation lemma**: $(\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p\ a}$
4. $F \leftarrow F \land ((\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p\ a})$
5. $E \leftarrow \{\neg\, \mathrm{p\ a}, \mathrm{p\ a}\}$ and $Q \leftarrow \{\forall x.\ \mathrm{p}\ x\}$

Example — SMT with quantifiers

$$F = (\forall x.\ \mathrm{p}\ x) \land \neg\, \mathrm{p\ a}$$

1. $E = \{\neg\, \mathrm{p\ a}\}$ and $Q = \{\forall x.\ \mathrm{p}\ x\}$
2. Model from SAT solver: atom p a is false and atom $\forall x.\ \mathrm{p}\ x$ is true
3. **Instantiation lemma**: $(\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p\ a}$
4. $F \leftarrow F \land ((\forall x.\ \mathrm{p}\ x) \Longrightarrow \mathrm{p\ a})$
5. $E \leftarrow \{\neg\, \mathrm{p\ a},\ \mathrm{p\ a}\}$ and $Q \leftarrow \{\forall x.\ \mathrm{p}\ x\}$
6. No model can be found; SAT solver says unsat

Instantiation strategies — state of the art

MBQI …

- iterately constructs a model of the quantifier-free part
- uses the model to **build suitable terms** to instantiate quantified variables

SyQI …

- synthesizes terms for each quantified variable **using a grammar**

Instantiation strategies — disadvantages

MBQI …

- instantiates quantifiers only with terms that denote values in a theory

- cannot handle higher-order problems well

SyQI …

- ignores contextual information

MBQI example

$$(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \wedge \mathrm{p}\ \mathrm{a}$$

1. $E = \{\mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
2. Model from SAT solver: atoms p a and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true

MBQI example

$$(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \wedge \mathrm{p}\ \mathrm{a}$$

1. $E = \{\mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
2. Model from SAT solver: atoms $\mathrm{p}\ \mathrm{a}$ and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true
3. MBQI substitution: $\{y \mapsto \lambda x.\, 0\}$
4. Instantiation lemma:

   $(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \Longrightarrow \neg\, \mathrm{p}\ ((\lambda x.\, 0)\ \mathrm{a}) \rightarrow_{\beta} \neg\, \mathrm{p}\ 0$

MBQI example

$$(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \wedge \mathrm{p}\ \mathrm{a}$$

1. $E = \{\mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
2. Model from SAT solver: atoms p a and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true
3. MBQI substitution: $\{y \mapsto \lambda x.\, 0\}$
4. Instantiation lemma:
   $(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \Longrightarrow \neg\, \mathrm{p}\ ((\lambda x.\, 0)\ \mathrm{a}) \rightarrow_{\beta} \neg\, \mathrm{p}\ 0$
5. $E \leftarrow \{\mathrm{p}\ \mathrm{a}, \neg\, \mathrm{p}\ 0\}$ and $Q \leftarrow \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
6. Model from SAT solver: atoms p a and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true and atom p 0 is false

MBQI example

$$(\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})) \wedge \mathsf{p}\ \mathsf{a}$$

1. $E = \{\mathsf{p}\ \mathsf{a}\}$ and $Q = \{\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})\}$
2. Model from SAT solver: atoms $\mathsf{p}\ \mathsf{a}$ and $\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})$ are true
3. MBQI substitution: $\{y \mapsto \lambda x.\, 0\}$
4. Instantiation lemma:
   $(\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})) \Longrightarrow \neg\, \mathsf{p}\ ((\lambda x.\, 0)\ \mathsf{a}) \rightarrow_\beta \neg\, \mathsf{p}\ 0$
5. $E \leftarrow \{\mathsf{p}\ \mathsf{a}, \neg\, \mathsf{p}\ 0\}$ and $Q \leftarrow \{\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})\}$
6. Model from SAT solver: atoms $\mathsf{p}\ \mathsf{a}$ and $\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})$ are true and atom $\mathsf{p}\ 0$ is false
7. MBQI substitution: $\{y \mapsto \lambda x.\, 1\}$
8. ...

Instantiation strategies — disadvantages

MBQI …

- instantiates quantifiers only with terms that denote values in a theory

- cannot handle higher-order problems well

SyQI …

- ignores contextual information

Instantiation strategies — our solution

MBQI …

- ~~instantiates quantifiers only with terms that denote values in a theory~~
  Our strategy considers uninterpreted symbols
- ~~cannot handle higher-order problems well~~
  Our strategy handles more higher-order problems

SyQI …

- ~~ignores contextual information~~
  Our strategy uses fast model-finding while enhancing the diversity of instantiations

Example with our strategy

$$(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \wedge \mathrm{p}\ \mathrm{a}$$

1. $E = \{\mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
2. Model from SAT solver: atoms $\mathrm{p}\ \mathrm{a}$ and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true

Example with our strategy

$$(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \wedge \mathrm{p}\ \mathrm{a}$$

1. $E = \{\mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
2. Model from SAT solver: atoms p a and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true
3. Substitution: $\{y \mapsto \lambda x.\, x\}$
4. Instantiation lemma: $(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \Longrightarrow (\neg\, \mathrm{p}\ ((\boldsymbol{\lambda x.\, x})\ \mathrm{a}))$
   $\rightarrow_\beta \neg\, \mathrm{p}\ \mathrm{a}$

Example with our strategy

$$(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \wedge \mathrm{p}\ \mathrm{a}$$

1. $E = \{\mathrm{p}\ \mathrm{a}\}$ and $Q = \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
2. Model from SAT solver: atoms $\mathrm{p}\ \mathrm{a}$ and $\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})$ are true
3. Substitution: $\{y \mapsto \lambda x.\, x\}$
4. Instantiation lemma: $(\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})) \Longrightarrow (\neg\, \mathrm{p}\ ((\boldsymbol{\lambda x.\, x})\ \mathrm{a}))$
   $\rightarrow_\beta \neg\, \mathrm{p}\ \mathrm{a}$
5. $E \leftarrow \{\mathrm{p}\ \mathrm{a}, \neg\, \mathrm{p}\ \mathrm{a}\}$ and $Q \leftarrow \{\forall y.\, \neg\, \mathrm{p}\ (y\ \mathrm{a})\}$
6. No model can be found; SAT solver says unsat

Our strategy

1. Initialize a SyQI enumerator within MBQI
2. Construct a grammar incorporating uninterpreted symbols from the entire formula
   - Consider bound variables as terminal symbols
3. Iteratively enumerate terms for each quantified variable
   - Consider λ-abstractions for higher-order variables
4. If the instance, according to the current model, is unsuccessful, continue to the next candidate
5. Revert to MBQI if all possibilities are exhausted

Our strategy

1. Initialize a SyQI enumerator within MBQI
2. Construct a grammar incorporating uninterpreted symbols from the entire formula
   - Consider bound variables as terminal symbols
3. Iteratively enumerate terms for each quantified variable
   - Consider λ-abstractions for higher-order variables
4. If the instance, according to the current model, is unsuccessful, continue to the next candidate
5. Revert to MBQI if all possibilities are exhausted

Our strategy

1. Initialize a SyQI enumerator within MBQI
2. Construct a grammar incorporating uninterpreted symbols from the entire formula
   - Consider bound variables as terminal symbols
3. Iteratively enumerate terms for each quantified variable
   - Consider λ-abstractions for higher-order variables
4. If the instance, according to the current model, is unsuccessful, continue to the next candidate
5. Revert to MBQI if all possibilities are exhausted

Choice of grammar

$$(\forall y.\, \neg\, \mathsf{p}\ (y\ \mathsf{a})) \wedge \mathsf{p}\ \mathsf{a}$$

- Grammar constructed for y:

$$\mathcal{A} ::= \boldsymbol{\lambda x.}\ \mathcal{B}$$

$$\mathcal{B} ::= \boldsymbol{x} \mid 0 \mid 1 \mid \mathcal{B} + \mathcal{B} \mid \mathcal{B} - \mathcal{B} \mid \cdots$$

- Our strategy enumerates terms from the grammar and creates the substitution $\{y \mapsto \lambda x.\, x\}$

Choice of grammar

Symbols included in the grammar based on three options:

- **Local symbols**: symbols from the quantified formula
- **Bound variables**: bound variables from the quantified formula
- **Global symbols**: function symbols from the entire formula

Example using our strategy with options enabled

$$\forall y.\, \neg \forall z.\, \neg\, \mathsf{p}\ (y\ z) \vee \mathsf{p}\ (\mathsf{f}\ z)$$

1. Grammar for y with set of symbols $\{\mathsf{f}, \mathsf{p}\}$:

$$\mathcal{A} ::= \boldsymbol{\lambda x.}\ \mathcal{B}$$

$$\mathcal{B} ::= \boldsymbol{x} \mid \mathbf{f}\ \mathcal{B} \mid 0 \mid 1 \mid \mathcal{B} + \mathcal{B} \mid \mathcal{B} - \mathcal{B} \mid \cdots$$

2. Instantiation lemma:
   $(\forall y.\, \neg \forall z.\, \neg\, \mathsf{p}\ (y\ z) \vee \mathsf{p}\ (\mathsf{f}\ z)) \Longrightarrow \neg \forall z.\, \neg\, \mathsf{p}\ (\mathsf{f}\ z) \vee \mathsf{p}\ (\mathsf{f}\ z)$
3. Skolemization lemma:
   $(\neg \forall z.\, \neg\, \mathsf{p}\ (\mathsf{f}\ z) \vee \mathsf{p}\ (\mathsf{f}\ z)) \Longrightarrow \mathsf{p}\ (\mathsf{f}\ \mathsf{sk}) \wedge \neg\, \mathsf{p}\ (\mathsf{f}\ \mathsf{sk})$

Example using our strategy with options enabled

$$(\forall x, y, z.\ x\ y = x\ z) \wedge \mathsf{a} \neq \mathsf{b}$$

1. Grammar for $x : (\tau \to \tau) \to \tau$

$$\mathcal{A} ::= \lambda w.\ \mathcal{B}$$

$$\mathcal{B} ::= w\ \mathcal{B} \mid \mathsf{a} \mid \mathsf{b} \mid \cdots$$

2. Grammar for $y : (\tau \to \tau)$ and $z : (\tau \to \tau)$

$$\mathcal{A} ::= \lambda w.\ \mathcal{B}$$

$$\mathcal{B} ::= w \mid \mathsf{a} \mid \mathsf{b} \mid \cdots$$

3. Instantiation lemma:

$$(\forall \boldsymbol{x}, y, z.\ x\ y = x\ z) \Longrightarrow ((\boldsymbol{\lambda w.\ w\ b})\ \lambda w.\ w) = ((\boldsymbol{\lambda w.\ w\ b})\ \lambda w.\ a)$$

Example using our strategy with options enabled

$$(\forall x, y, z.\; x\; y = x\; z) \land \mathsf{a} \neq \mathsf{b}$$

1. Grammar for $x : (\tau \to \tau) \to \tau$

$$\mathcal{A} ::= \lambda w.\; \mathcal{B}$$

$$\mathcal{B} ::= w\; \mathcal{B} \mid \mathsf{a} \mid \mathsf{b} \mid \cdots$$

2. Grammar for $y : (\tau \to \tau)$ and $z : (\tau \to \tau)$

$$\mathcal{A} ::= \lambda w.\; \mathcal{B}$$

$$\mathcal{B} ::= w \mid \mathsf{a} \mid \mathsf{b} \mid \cdots$$

3. Instantiation lemma:

   $(\forall x, \boldsymbol{y}, z.\; x\; y = x\; z) \Longrightarrow ((\boldsymbol{\lambda w.\; w})\; b) = ((\lambda w.\; a)\; b)$

Example using our strategy with options enabled

$$(\forall x, y, z.\ x\ y = x\ z) \wedge \mathrm{a} \neq \mathrm{b}$$

1. Grammar for $x : (\tau \to \tau) \to \tau$

$$\mathcal{A} ::= \lambda w.\ \mathcal{B}$$

$$\mathcal{B} ::= w\ \mathcal{B} \mid \mathrm{a} \mid \mathrm{b} \mid \cdots$$

2. Grammar for $y : (\tau \to \tau)$ and $z : (\tau \to \tau)$

$$\mathcal{A} ::= \lambda w.\ \mathcal{B}$$

$$\mathcal{B} ::= w \mid \mathrm{a} \mid \mathrm{b} \mid \cdots$$

3. Instantiation lemma:

   $(\forall x, y, \boldsymbol{z}.\ x\ y = x\ z) \Longrightarrow b = ((\boldsymbol{\lambda w.\ a})\ b)$

Example using our strategy with options enabled

$$(\forall x, y, z.\ x\ y = x\ z) \wedge \mathsf{a} \neq \mathsf{b}$$

1. Grammar for $x : (\tau \to \tau) \to \tau$

$$\mathcal{A} ::= \lambda w.\ \mathcal{B}$$

$$\mathcal{B} ::= w\ \mathcal{B} \mid \mathsf{a} \mid \mathsf{b} \mid \cdots$$

2. Grammar for $y : (\tau \to \tau)$ and $z : (\tau \to \tau)$

$$\mathcal{A} ::= \lambda w.\ \mathcal{B}$$

$$\mathcal{B} ::= w \mid \mathsf{a} \mid \mathsf{b} \mid \cdots$$

3. Instantiation lemma:
   $(\forall x, y, z. x\ \ y = x\ z) \Longrightarrow b = a$

Evaluation on higher-order TPTP benchmarks

	Vampire	Zipperposition	cvc5[s]	cvc5[m]	cvc5[M]
Satisfiable	6	0	78	121	**129**
Unsatisfiable	**1757**	1499	1304	1637	1670
Total	1763	1499	1382	1758	**1799**

cvc5[s]: SyQI
cvc5[m]: MBQI
cvc5[M]: Our strategy

Our strategy solves 36 more problems

Evaluation on higher-order TPTP benchmarks

	Vampire	Zipperposition	cvc5[s]	cvc5[m]	cvc5[M]
Satisfiable	6	0	78	121	**129**
Unsatisfiable	**1757**	1499	1304	1637	1670
Total	1763	1499	1382	1758	**1799**

cvc5[s]: SyQI

cvc5[m]: MBQI

cvc5[M]: Our strategy

Our strategy solves 36 more problems; **now** 137 more problems

Evaluation on first-order SMT-LIB benchmarks

	Boolector	Bitwuzla	Z3	cvc5[s]	cvc5[m]	cvc5[M]
Bit-vectors	5565	**5708**	5548	5224	5446	5670
Non-arithmetic logics		3420	2959	3417	3732	**3891**
Arithmetic logics			**6509**	5874	5814	6052
Total	5565	9128	15016	14515	14992	**15613**

cvc5[s]: SyQI
cvc5[m]: MBQI
cvc5[M]: Our strategy

Our strategy solves 597 more problems

Conclusion

Our quantifier instantiation strategy …

- uses fast model-finding capabilities of MBQI
- uses diversity of terms considered by SyQI
- helps solve many problems cvc5 could not solve

Future work ideas are to …

- generate dynamic grammars
- enumerate specific axioms

Augmenting Model-Based Instantiation with Fast Enumeration in SMT

Lydia Kondylidou[1] Andrew Reynolds[2] Jasmin Blanchette[1]

[1]Ludwig Maximilian University of Munich

[2]The University of Iowa