

Logikbasierte Systeme der KI; Wissensverarbeitung

Prädikatenlogik, SMT-Solver

Prof. Dr. M. Schmidt-Schauß

SoSe 2025

SMT = SAT mit extra Theorien

Erweiterung der Aussagenlogik mit **quantorenfreien** Formeln
UND: Funktionssymbole, Prädikate mit vordefinierter Bedeutung.
Satisfiability **M**odulo **T**heories (SMT).

Idee: Aussagenlogische Formeln und Ausdrücke mit reellen Zahlen, Integer, Listen, Array, Bitvektoren usw.

Schlüsse: sollen aus den Eingaben gezogen werden können.

Test: (Un)Erfüllbarkeit von (speziellen) Formeln

SMT Solver: Es gibt Implementierungen: zB Z3 and CVC4,
...

SMT Beispiele

Theorie: Ganze Zahlen mit $=, <, >, \leq, \geq$
Operatoren; $+, -, *$ und evtl. weitere;
Keine Quantoren,
Keine (nicht-interpretierten) Funktionssymbole

Beispielformel, z.B. über den ganzen Zahlen

$$a < b \wedge b < c \wedge c * c < d \wedge a = c - 1$$

Frage: Erfüllbar oder unerfüllbar?

SMT Beispiele

Theorie: Ganze Zahlen mit $=, <, >, \leq, \geq$
Operatoren; $+, -, *$ und evtl. weitere;
Keine Quantoren,
Keine (nicht-interpretierten) Funktionssymbole

Beispielformel, z.B. über den ganzen Zahlen

$$a < b \wedge b < c \wedge c * c < d \wedge a = c - 1$$

Frage: Erfüllbar oder unerfüllbar?

1): $a + 2 \leq c$ gilt wegen $a < b \wedge b < c$

2): Das widerspricht $a = c - 1$

Also: Eingabeformel unerfüllbar

SMT Beispiele: Theorie: Ganze Zahlen

Allgemeinere Beispielformel:

aussagenlogisch		Formelanteil zu ganzen Zahlen
$(A \vee \neg B$	$\vee$	$a < b \vee b \neq c)$
$\wedge (\neg A$	$\vee$	$d < c * c + a)$
$\wedge (B$	$\vee$	$a = c - 1)$

A, B, C sind aussagenlogisch (d.h. wahr oder falsch;)

a, b, c, d sind Variablen zu ganzen Zahlen;

$+, *, -, >, \neq$ sind Operatoren auf ganzen Zahlen.

Fragen: Bedeutung der Formeln / Semantik?

Welche syntaktischen / semantischen Beschränkungen?

Wie kann man korrekte Schlüsse ziehen?

Wie erweitert man den DPLL-Algorithmus?

Formelformat

Klausel: $F \vee G$

F Aussagenlogische Klausel

G : Theorie-Formel Z.B.

Konjunktion von (Un-)Gleichungen über $\mathbb{Z}$

Formel: Konjunktion von Klauseln

Deduktions-Algorithmen spezifisch für Theorien:

- Entscheidungsverfahren für Konjunktionen von Theorie-Klauseln
- oder: Herleitung von einfachen Formeln aus einer gegebenen Konjunktion von Literalen.

SMT-Beispiel.

Theorie = nicht-interpretierte Funktionssymbole

- Formeln:
Aussagenlogische Formel mit Gleichungen und Ungleichungen zwischen Termen
- Die Werte der Terme sind unbekannt.
- gleich geschriebene Terme sind auch gleich unter =.
- Und es gilt: $x = y \implies f(x) = f(y)$.

SMT-Beispiel.

Theorie = nicht-interpretierte Funktionssymbole

$$\Phi = \underbrace{\{g(a) = c\}}_A \wedge \underbrace{(f(g(a)) \neq f(c) \vee g(a) = d)}_{\neg B} \wedge \underbrace{c \neq d}_{\neg D}$$

Vorgehen: **SAT**-Solver und **Theorie-Solver** abwechselnd.

Regel: Wenn F widersprüchlich, dann kann man $\neg F$ hinzufügen!

1. **SAT**: $A, \neg B \vee C, \neg D$ hat Modell $A, C, \neg D$
2. **Theorie-Solver**: $\{A, C, \neg D\}$ ist widersprüchlich!

SMT-Beispiel.

Theorie = nicht-interpretierte Funktionssymbole

$$\Phi = \underbrace{\{g(a) = c\}}_A \wedge \underbrace{(f(g(a)) \neq f(c) \vee g(a) = d)}_{\neg B \vee C} \wedge \underbrace{c \neq d}_{\neg D}$$

Vorgehen: **SAT**-Solver und **Theorie-Solver** abwechselnd.

Regel: Wenn F widersprüchlich, dann kann man $\neg F$ hinzufügen!

1. **SAT**: $A, \neg B \vee C, \neg D$ hat Modell $A, C, \neg D$
2. **Theorie-Solver**: $\{A, C, \neg D\}$ ist widersprüchlich!
3. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D$ hat als Modell $A, \neg B, \neg C, \neg D$.
4. **Theorie-Solver**: $\{A, \neg B\}$ ist bereits widersprüchlich!

SMT-Beispiel.

Theorie = nicht-interpretierte Funktionssymbole

$$\Phi = \underbrace{\{g(a) = c\}}_A \wedge \underbrace{(f(g(a)) \neq f(c) \vee g(a) = d)}_{\neg B} \wedge \underbrace{c \neq d}_{\neg D}$$

Vorgehen: **SAT**-Solver und **Theorie-Solver** abwechselnd.

Regel: Wenn F widersprüchlich, dann kann man $\neg F$ hinzufügen!

1. **SAT**: $A, \neg B \vee C, \neg D$ hat Modell $A, C, \neg D$
2. **Theorie-Solver**: $\{A, C, \neg D\}$ ist widersprüchlich!
3. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D$ hat als Modell $A, \neg B, \neg C, \neg D$.
4. **Theorie-Solver**: $\{A, \neg B\}$ ist bereits widersprüchlich!
5. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D, \neg A \vee B$ ist widersprüchlich

SMT-Beispiel.

Theorie = nicht-interpretierte Funktionssymbole

$$\Phi = \underbrace{\{g(a) = c\}}_A \wedge \underbrace{(f(g(a)) \neq f(c) \vee g(a) = d)}_{\neg B} \wedge \underbrace{c \neq d}_{\neg D}$$

Vorgehen: **SAT**-Solver und **Theorie-Solver** abwechselnd.

Regel: Wenn F widersprüchlich, dann kann man $\neg F$ hinzufügen!

1. **SAT**: $A, \neg B \vee C, \neg D$ hat Modell $A, C, \neg D$
2. **Theorie-Solver**: $\{A, C, \neg D\}$ ist widersprüchlich!
3. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D$ hat als Modell $A, \neg B, \neg C, \neg D$.
4. **Theorie-Solver**: $\{A, \neg B\}$ ist bereits widersprüchlich!
5. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D, \neg A \vee B$ ist widersprüchlich
6. $\Rightarrow$ Eingabe-Formel ist widersprüchlich

SMT-Beispiel.

Theorie = nicht-interpretierte Funktionssymbole

$$\Phi = \underbrace{\{g(a) = c\}}_A \wedge \underbrace{(f(g(a)) \neq f(c) \vee g(a) = d)}_{\neg B} \wedge \underbrace{c \neq d}_{\neg D}$$

Vorgehen: **SAT**-Solver und **Theorie-Solver** abwechselnd.

Regel: Wenn F widersprüchlich, dann kann man $\neg F$ hinzufügen!

1. **SAT**: $A, \neg B \vee C, \neg D$ hat Modell $A, C, \neg D$
2. **Theorie-Solver**: $\{A, C, \neg D\}$ ist widersprüchlich!
3. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D$ hat als Modell $A, \neg B, \neg C, \neg D$.
4. **Theorie-Solver**: $\{A, \neg B\}$ ist bereits widersprüchlich!
5. **SAT**: $A, \neg B \vee C, \neg D, \neg A \vee \neg C \vee D, \neg A \vee B$ ist widersprüchlich
6. $\Rightarrow$ Eingabe-Formel ist widersprüchlich

Erfolg!:

$$g(a) = c \wedge (f(g(a)) \neq f(c) \vee g(a) = d) \text{ impliziert } c = d$$