ПОЛИТЕХ
Санкт-Петербургский
политехнический университет
Петра Великого

ИКНТ

Верификация и анализ программ SMT и SAT

2020

КСПТ
КОМПЬЮТЕРНЫЕ СИСТЕМЫ И ПРОГРАММНЫЕ ТЕХНОЛОГИИ

Задача SAT-разрешимости

- Задача выполнимости булевой формулы (SAT или CNF-SAT)
- Впервые введена С. Куком и Л. Левиным в 1971-1972 годах
- Первая NP-полная задача
- Состоит в определении, может ли логическая формула быть выполнена:
  - Нужно определить, существует ли такой набор значений входящий в неё переменных, при котором она становится истинной

Задача SAT-разрешимости

- Логическая формула в логике 1-го порядка
  - Содержит логические переменные
  - Содержит операции конъюнкции, дизъюнкции, отрицания
  - Может содержать скобки
  - Записана в конъюнктивной нормальной форме
  - $(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge \neg x_3$
- Два возможных исхода:
  - SAT – выполнима
  - UNSAT – невыполнима

Задача SAT-разрешимости

- Одна из первых NP-полных задач, **которые научились эффективно решать**
- Все NP-полные задачи сводимы друг к другу
- Формула:
  - $(x_1 \vee \neg x_5 \vee x_4) \wedge (\neg x_1 \vee x_5 \vee x_3 \vee x_4)$
- Решение:
  - SAT
  - $x_1 = True$
  - $x_3 = False$
  - $x_4 = True$
  - $x_5 = False$

SAT-солверы

- SAT-солверы - средства эффективного решения задачи SAT
- С появлением SAT-солверов NP-сложность задачи перестала быть непреодолимой преградой
- Известные солверы
  - MiniSAT / CryptoMiniSAT
  - Lingeling (PicoSAT)
  - Sat4j
  - Glucose
  - …

Задачи выполнимости и доказательства

- Задачи выполнимости и доказательства *дуальны* и сводятся друг к другу:
  - Формула $F(X)$ выполнима, **iff** $\neg F(X)$ неверна
  - Формула $G(X)$ верна, **iff** $\neg G(X)$ невыполнима

Задача SMT

- Satisfiability modulo theories, SMT – задача выполнимости формул **с учётом лежащих в их основе теорий**
- Отличие от SAT-разрешимости – вместо булевых переменных могут использоваться логические предикаты, реализующие другие теории
  - $(x > 2) \wedge (2*y > x) \wedge (z < 0)$
- Задача выполнимости решается над формулами логики первого порядка
- Результат решения задачи SMT – вектор значений переменных, при которых формула становится истинной

Задача SMT. Теории

- Теория - набор возможных значений и операций над ними
- Теории можно комбинировать в новые теории
- В новых теориях задача разрешимости может перестать быть NP-полной или вообще разрешимой

Задача SMT. Теории

- Поддерживаемые в разных солверах возможности
  - Поддерживаемые теории
    - Булева теория
    - Теория целых чисел
    - Теория рациональных чисел
    - Теория вещественных чисел
    - Теория массивов
    - Теория битовых векторов
    - Теория вероятностной логики
    - Теория строк
    - Теория неинтерпретируемых функций
  - Кванторы

SMT. Булева теория

- Код: нет
- Поддерживается: всеми
- Операции: те же, что и в задаче SAT

SMT. Теория целых чисел

- Код: (L|NL)IA
- Поддерживается: Z3, CVC4, MathSAT, etc.
- Операции: линейной (нелинейной) арифметики над целыми числами
- Теория целых чисел в общем случае **неразрешима**

SMT. Теория вещественных чисел

- Код: (L|NL)RA
- Поддерживается: Z3, CVC4, MathSAT, etc.
- Операции: линейной (нелинейной) арифметики над действительными числами
- По факту иррациональные числа не поддерживаются ни в задачах, ни в результатах из-за невозможности их представления
- Теория вещественных чисел разрешима

SMT. Теория неинтерпретируемых функций

- Код: UF
- Поддерживается: почти всеми
- Операции: объявление функций, применение функций
- Теория «пустых» функций, содержащих только объявления, работает за счёт определения понятия «функция»
- При комбинировании теорий функции могут принимать на вход и возвращать объекты из других теорий
- Разрешима, в чистом виде сводится к задаче SAT

SMT. Теория битовых векторов

- Код: BV
- Поддерживается: почти всеми
- Операции: машинная арифметика, в том числе битовая, сравнение машинных слов
- Представляют собой набор бит ограниченного размера, сводится к задаче SAT, поэтому разрешима. Процесс сведения называется bit-blasting

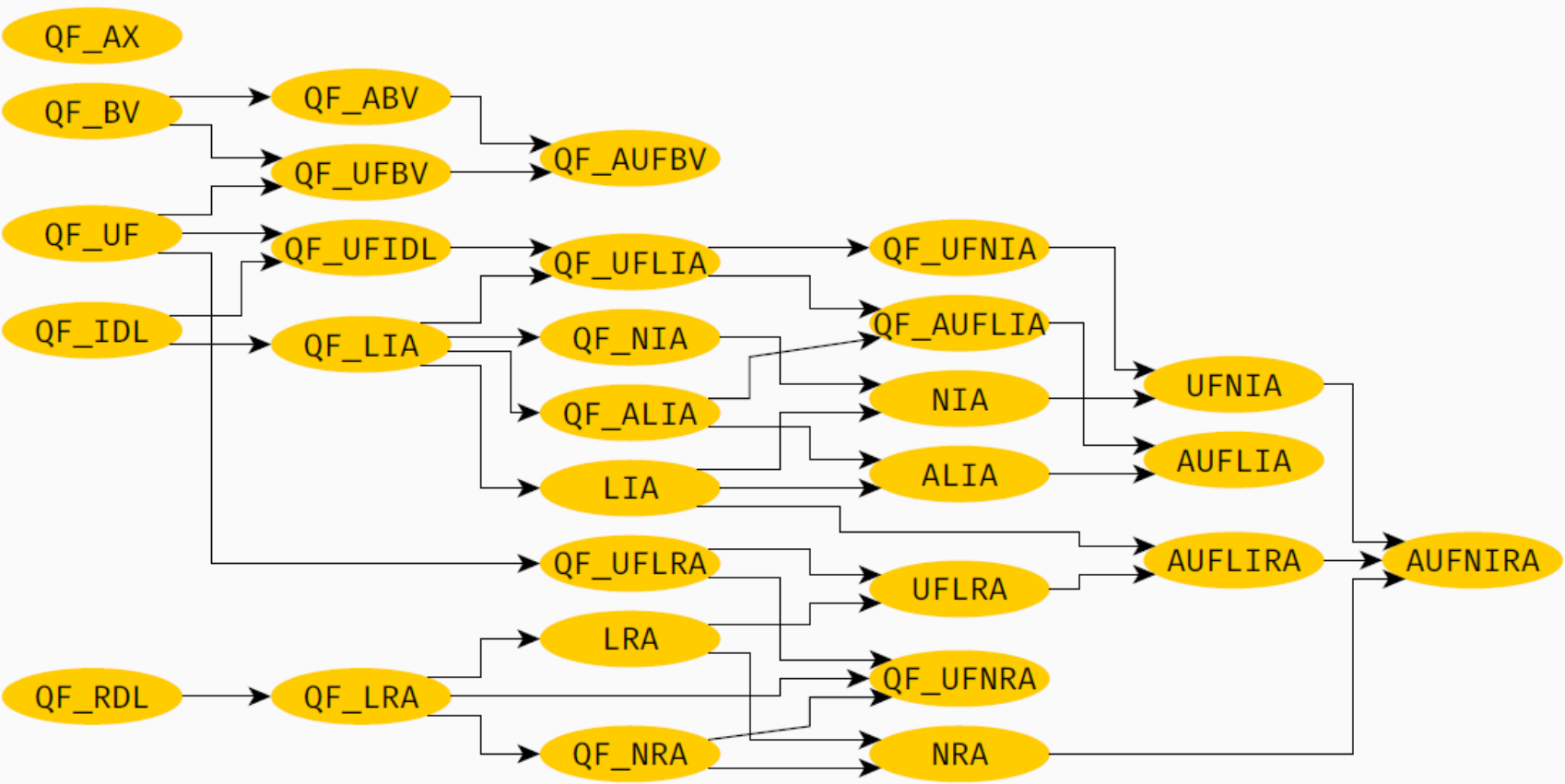
SMT. Теория массивов

- Код: A(X)
- Поддерживается: Z3, MathSAT, CVC4, Boolector, STP, etc.
- Операции: создание массива, чтение массива, запись в массив
- Массив это сложный объект, привязанный к теории индексов (объекты которой используются как индексы) и теории элементов (объекты которой используются как элементы)
- Разрешима, если разрешимы входящие в неё теории.

SMT. Теории с кванторами

- По умолчанию код любых теорий позволяет использовать кванторы.
- Теории без кванторов имеют префикс QF_
- Примеры:
  - QF_ABV — теория бит-векторов и массивов, без кванторов
  - AUFLIA — теория линейной арифметики над целыми числами с массивами и функциями

SMT. Теории


QF_AX
QF_BV
QF_ABV
QF_AUFBV
QF_UFBV
QF_UF
QF_UFIDL
QF_UFLIA
QF_UFNIA
QF_IDL
QF_LIA
QF_NIA
QF_AUFLIA
QF_ALIA
NIA
UFNIA
AUFLIA
LIA
ALIA
QF_UFLRA
AUFLIRA
AUFNIRA
UFLRA
LRA
QF_UFNRA
QF_RDL
QF_LRA
QF_NRA
NRA

SMT-рещатели

- Существуют эффективные SMT-решатели (SMT-солверы):
  - Boolector
  - CVC-3, CVC-4
  - MatSAT
  - OpenSMT
  - Yices
  - Z3 (Microsoft Research)
  - …
- Проходят регулярные чемпионаты солверов SMT-COMP: http://www.smtcomp.org

Взаимодействие с солверами

- Имеются стандарты для представления входных данных для SMT-солверов: SMT-LIB и SMT-LIB2
  - Плюс: легко заменять солверы
  - Минус: дополнительную информацию не извлечь
- Существуют биндинги для некоторых языков программирования: C, C++, Java, OCaml
  - Плюс: вся функциональность доступна
  - Минус: у каждого солвера свои особенности
- Но: не существует единого стандарта для представления выходных данных

Пример SMT-LIB

```
; Integer arithmetic
; x - y = x - y + 1
(set-logic QF_LIA)
(declare-const x Int)
(declare-const y Int)
(assert (= (- x y) (+ x (- y) 1)))
(check-sat)
; unsat
(exit)
```

Пример SMT-LIB

```
; Modeling sequential code in SSA form
;; Swap
; int x, y;
; int t = x;
; x = y;
; y = x;
(set-logic QF_UFLIA)
(set-option :produce-models true)
(declare-fun x (Int) Int)
(declare-fun y (Int) Int)
(declare-fun t (Int) Int)
(assert (= (t 1) (x 0)))
(assert (= (x 1) (y 0)))
(assert (= (y 1) (x 1)))
(assert (not (and (= (x 1) (y 0)) (= (y 1) (x 0)))))
(check-sat)
```

Что можно решать с помощью SMT-солверов? Загадка Эйнштейна

- 5 человек разных национальностей живут в 5 домах разного цвета, курят 5 разных марок сигарет, выращивают 5 разных видов животных и пьют 5 разных видов напитков.

- Норвежец живет в первом доме.
- Англичанин живет в красном доме.
- Зеленый дом находится левее белого.
- Датчанин пьет чай.
- Тот, кто курит Rothmans, живет рядом с тем, кто выращивает кошек.
- Тот, кто живет в желтом доме, курит Dunhill.
- Немец курит Marlboro.
- Тот, кто живет в центре, пьет молоко.
- Сосед того, кто курит Rothmans, пьет воду.
- Тот, кто курит Pall Mall, выращивает птиц.
- Швед выращивает собак.
- Норвежец живет рядом с синим домом.
- Тот, кто выращивает лошадей, живет в синем доме.
- Тот, кто курит Philip Morris, пьет пиво.
- В зеленом доме пьют кофе.

- Вопрос: Кто выращивает рыбок?
- Эту задачу (и подобные ей) можно решить с помощью SMT-солвера