

Автоматический вывод индуктивных инвариантов программ с алгебраическими типами данных

Костюков Юрий Олегович

Научный руководитель:
д. т. н., доцент Кознов Дмитрий Владимирович

2023



Санкт-Петербургский
государственный
университет

Содержание

Обзор предметной области

Постановка задачи

Результаты

Научная новизна

Публикации и выступления

Результаты

Верификация программ путём вывода индуктивных инвариантов

```
x, y := 0, 0
```

```
while * do
```

```
    y := y + x
```

```
    x := x + 1
```

```
assert(y ≥ 0)
```

Верификация программ путём вывода индуктивных инвариантов

$\{x = 0 \wedge y = 0\}$

while * **do**

$y := y + x$

$x := x + 1$

$\{y \geq 0\}$

Верификация программ путём вывода индуктивных инвариантов

$\{x = 0 \wedge y = 0\}$

while * **do**

$y := y + x$

$x := x + 1$

$\{y \geq 0\}$

Как доказать корректность этой тройки Хоара?

Верификация программ путём вывода индуктивных инвариантов

$\{x = 0 \wedge y = 0\}$

while * **do**

$y := y + x$

$x := x + 1$

$\{y \geq 0\}$

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского индуктивного инварианта* φ

$$x = 0 \wedge y = 0 \rightarrow \varphi(x, y)$$

$$\varphi(x, y) \wedge x' = x + 1 \wedge y' = y + x \rightarrow \varphi(x', y')$$

$$\varphi(x, y) \rightarrow y \geq 0$$

Верификация программ путём вывода индуктивных инвариантов

$\{x = 0 \wedge y = 0\}$

while * **do**

$y := y + x$

$x := x + 1$

$\{y \geq 0\}$

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского индуктивного инварианта* φ

Пользователь: $y \geq 0$ — индуктивный инвариант?

$x = 0 \wedge y = 0 \rightarrow \varphi(x, y)$

$\varphi(x, y) \wedge x' = x + 1 \wedge y' = y + x \rightarrow \varphi(x', y')$

$\varphi(x, y) \rightarrow y \geq 0$

Верификация программ путём вывода индуктивных инвариантов

$\{x = 0 \wedge y = 0\}$

while * **do**

$y := y + x$

$x := x + 1$

$\{y \geq 0\}$

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского индуктивного инварианта* φ

Пользователь: $y \geq 0$ — индуктивный инвариант?

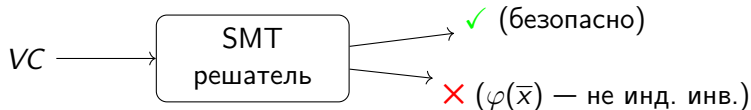
$$VC := \left\{ \begin{array}{l} \forall x, y. (x = 0 \wedge y = 0 \rightarrow y \geq 0) \wedge \\ \forall x, y, x', y'. (y \geq 0 \wedge x' = x + 1 \wedge y' = y + x \rightarrow y' \geq 0) \wedge \\ \forall x, y. (y \geq 0 \rightarrow y \geq 0) \end{array} \right.$$

Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского индуктивного инварианта* φ

Пользователь: $y \geq 0$ — индуктивный инвариант?



$$VC := \left\{ \begin{array}{l} \forall x, y. (x = 0 \wedge y = 0 \rightarrow y \geq 0) \wedge \\ \forall x, y, x', y'. (y \geq 0 \wedge x' = x + 1 \wedge y' = y + x \rightarrow y' \geq 0) \wedge \\ \forall x, y. (y \geq 0 \rightarrow y \geq 0) \end{array} \right.$$

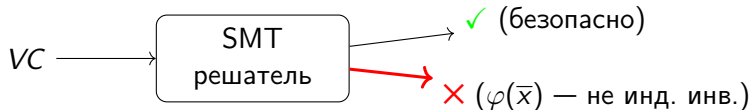
Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского индуктивного инварианта* φ

Пользователь: $y \geq 0$ — индуктивный инвариант?

SMT решатель: Нет, индуктивность нарушается при $x \mapsto -1$



$$VC := \left\{ \begin{array}{l} \forall x, y. (x = 0 \wedge y = 0 \rightarrow y \geq 0) \wedge \\ \forall x, y, x', y'. (y \geq 0 \wedge x' = x + 1 \wedge y' = y + x \rightarrow y' \geq 0) \wedge \\ \forall x, y. (y \geq 0 \rightarrow y \geq 0) \end{array} \right.$$

Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* индуктивного инварианта φ

Пользователь: $y \geq 0$ — индуктивный инвариант?

SMT решатель: Нет, индуктивность нарушается при $x \mapsto -1$

Пользователь: А усиленная формула: $x \geq 0 \wedge y \geq 0$?

$$x = 0 \wedge y = 0 \rightarrow \varphi(x, y)$$

$$\varphi(x, y) \wedge x' = x + 1 \wedge y' = y + x \rightarrow \varphi(x', y')$$

$$\varphi(x, y) \rightarrow y \geq 0$$

Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского индуктивного инварианта* φ

Пользователь: $y \geq 0$ — индуктивный инвариант?

SMT решатель: Нет, индуктивность нарушается при $x \mapsto -1$

Пользователь: А усиленная формула: $x \geq 0 \wedge y \geq 0$?

SMT решатель: Да, эта формула является индуктивным инвариантом

$$x = 0 \wedge y = 0 \rightarrow \varphi(x, y)$$

$$\varphi(x, y) \wedge x' = x + 1 \wedge y' = y + x \rightarrow \varphi(x', y')$$

$$\varphi(x, y) \rightarrow y \geq 0$$

Дизъюнкты Хорна с ограничениями

Как автоматизировать вывод индуктивных инвариантов?

$$\begin{aligned}x = 0 \wedge y = 0 &\rightarrow I(x, y) \\ I(x, y) \wedge x' = x + 1 \wedge y' = y + x &\rightarrow I(x', y') \\ I(x, y) &\rightarrow y \geq 0\end{aligned}$$

Дизъюнкты Хорна с ограничениями

Как автоматизировать вывод индуктивных инвариантов?

Заменить пользовательскую формулу на неинтерпретированный символ I

$$\begin{aligned}x &= 0 \wedge y = 0 \rightarrow I(x, y) \\ I(x, y) \wedge x' = x + 1 \wedge y' = y + x &\rightarrow I(x', y') \\ I(x, y) &\rightarrow y \geq 0\end{aligned}$$

Дизъюнкты Хорна с ограничениями

Как автоматизировать вывод индуктивных инвариантов?

Заменить пользовательскую формулу на неинтерпретированный символ I

Дизъюнкты Хорна с ограничениями

$$\begin{aligned} & x = 0 \wedge y = 0 \rightarrow I(x, y) \\ I(x, y) \wedge x' = x + 1 \wedge y' = y + x & \rightarrow I(x', y') \\ & I(x, y) \rightarrow y \geq 0 \end{aligned}$$

Дизъюнкты Хорна формально

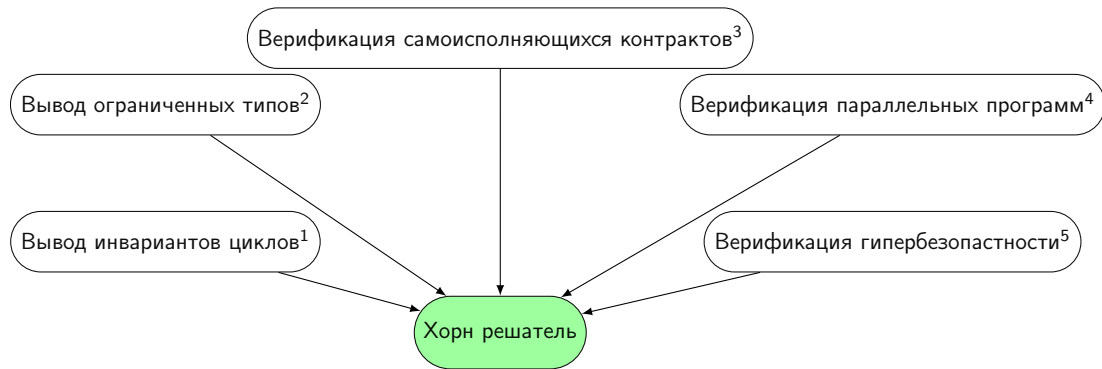
Дизъюнкт Хорна C — это формула первого порядка следующего вида:

$$\varphi \wedge P_1(\bar{x}_1) \wedge \dots \wedge P_n(\bar{x}_n) \rightarrow H$$

- ▶ **ограничение** φ — это формула теории алгебраических типов данных
- ▶ **голова** H — это либо ложь \perp , либо атом $P(\bar{x})$
- ▶ P_1, \dots, P_n, P — это неинтерпретированные символы
- ▶ все переменные (неявно) универсально квантифицированы

Система дизъюнктов Хорна — это конъюнкция дизъюнктов Хорна

Применения Хорн решателей



¹ Gurfinkel и др. The SeaHorn Verification Framework. CAV'15

² Tan и др. SolType: refinement types for arithmetic overflow in solidity. POPL'22

³ Alt и др. SolCMC: Solidity Compiler's Model Checker. CAV'22

⁴ Hoenicke и др. Thread Modularity at Many Levels. POPL'17

⁵ Shemer и др. Property Directed Self Composition. CAV'19

Дизъюнкты Хорна над АТД

Пример программы на языке HASKELL:

```
data Nat = Z | S Nat
data List = nil | cons Nat List
drop Z xs = xs
drop _ nil = nil
drop (S n) (cons(_, xs)) = drop n xs
assert (¬∃ n xs . xs /= nil && drop n xs == drop (S n) xs)
```

Условия верификации в виде дизъюнктов Хорна над АТД:

$$\top \rightarrow \text{drop}(Z, xs, xs)$$

$$\top \rightarrow \text{drop}(S(n), nil, nil)$$

$$\text{drop}(n, xs, rs) \rightarrow \text{drop}(S(n), \text{cons}(x, xs), rs)$$

$$\neg(xs = nil) \wedge \text{drop}(n, xs, ys) \wedge \text{drop}(S(n), xs, ys) \rightarrow \perp$$

Индуктивный инвариант

Пусть \mathcal{H} — модель теории АД, \mathcal{S} — система дизъюнктов Хорна.

Индуктивный инвариант \mathcal{I} — расширение модели $\mathcal{I} = \langle \mathcal{H}, \mathcal{R} \rangle$, такое что $\mathcal{I} \models \mathcal{S}$.

Индуктивный инвариант

Пусть \mathcal{H} — модель теории АТД, \mathcal{S} — система дизъюнктов Хорна.

Индуктивный инвариант \mathcal{I} — расширение модели $\mathcal{I} = \langle \mathcal{H}, \mathcal{R} \rangle$, такое что $\mathcal{I} \models \mathcal{S}$.

$$x = Z \wedge y = S(Z) \rightarrow inc(x, y)$$

$$x' = S(x) \wedge y' = S(y) \wedge inc(x, y) \rightarrow inc(x', y')$$

$$x = y \wedge inc(x, y) \rightarrow \perp$$

$$\mathcal{I}_1 = \mathcal{H} \left\{ inc \mapsto \{(x, y) \mid y = S(x)\} \right\}$$

$$\mathcal{I}_2 = \mathcal{H} \left\{ inc \mapsto \{(x, y) \mid x \neq y\} \right\}$$

$$\mathcal{I}_3 = \dots$$

Индуктивные инварианты составляют решётку

Индуктивный инвариант

Пусть \mathcal{H} — модель теории АТД, \mathcal{S} — система дизъюнктов Хорна.

Индуктивный инвариант \mathcal{I} — расширение модели $\mathcal{I} = \langle \mathcal{H}, \mathcal{R} \rangle$, такое что $\mathcal{I} \models \mathcal{S}$.

$$x = Z \wedge y = S(Z) \rightarrow inc(x, y)$$

$$x' = S(x) \wedge y' = S(y) \wedge inc(x, y) \rightarrow inc(x', y')$$

$$x = y \wedge inc(x, y) \rightarrow \perp$$

$$\mathcal{I}_1 = \mathcal{H} \left\{ inc \mapsto \{(x, y) \mid y = S(x)\} \right\}$$

$$\mathcal{I}_2 = \mathcal{H} \left\{ inc \mapsto \{(x, y) \mid x \neq y\} \right\}$$

$$\mathcal{I}_3 = \dots$$

Как представлять эти бесконечные множества?

Индуктивный инвариант

Пусть \mathcal{H} — модель теории АТД, \mathcal{S} — система дизъюнктов Хорна.

Индуктивный инвариант \mathcal{I} — расширение модели $\mathcal{I} = \langle \mathcal{H}, \mathcal{R} \rangle$, такое что $\mathcal{I} \models \mathcal{S}$.

$$x = Z \wedge y = S(Z) \rightarrow inc(x, y)$$

$$x' = S(x) \wedge y' = S(y) \wedge inc(x, y) \rightarrow inc(x', y')$$

$$x = y \wedge inc(x, y) \rightarrow \perp$$

$$\mathcal{I}_1 = \mathcal{H} \left\{ inc \mapsto \boxed{y = S(x)} \right\}$$

$$\mathcal{I}_2 = \mathcal{H} \left\{ inc \mapsto \boxed{\neg(x = y)} \right\}$$

$$\mathcal{I}_3 = \dots$$

Как представлять эти **бесконечные** множества?

Инварианты обычно представляются в логике первого порядка (ЛПП)

ЛПП задаёт т.н. *класс элементарных инвариантов*

Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow \text{even}(x)$$

$$\text{even}(y) \wedge x = S(S(y)) \rightarrow \text{even}(x)$$

$$\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp$$

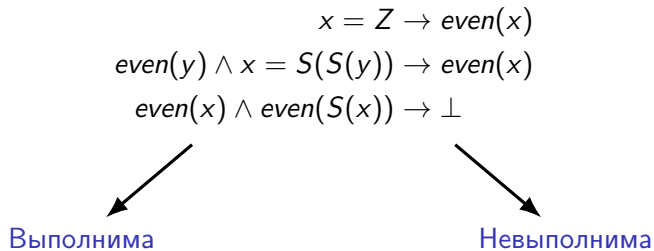
Проблема выразимости класса элементарных инвариантов

$$\begin{aligned}x &= Z \rightarrow \text{even}(x) \\ \text{even}(y) \wedge x &= S(S(y)) \rightarrow \text{even}(x) \\ \text{even}(x) \wedge \text{even}(S(x)) &\rightarrow \perp\end{aligned}$$

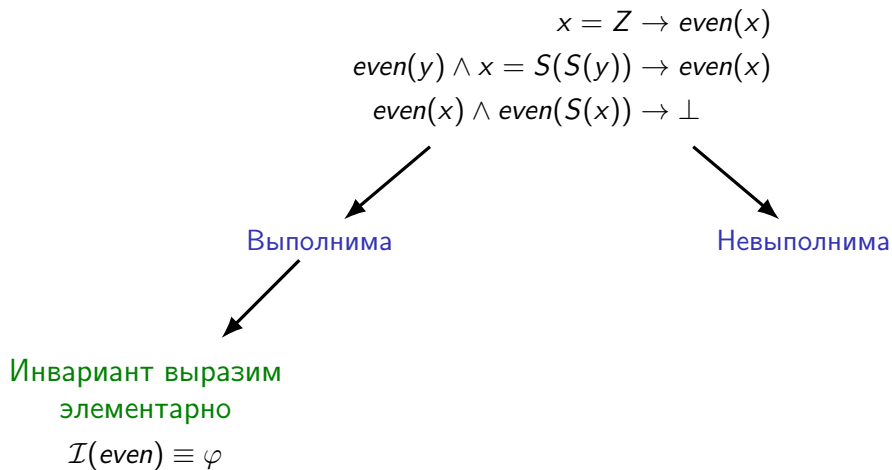


Невыполнима

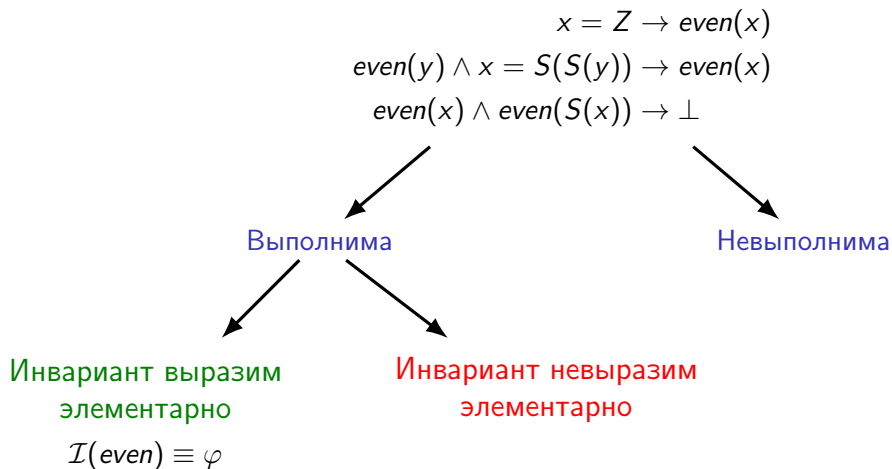
Проблема выразимости класса элементарных инвариантов



Проблема выразимости класса элементарных инвариантов



Проблема выразимости класса элементарных инвариантов



Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow \text{even}(x)$$

$$\text{even}(y) \wedge x = S(S(y)) \rightarrow \text{even}(x)$$

$$\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp$$

Язык АД первого порядка:

$$\varphi ::= t = t' \mid \neg\psi \mid \psi \wedge \psi' \mid \psi \vee \psi' \mid \forall x. \psi \mid \exists x. \psi$$

Инвариант выразим
элементарно

$$\mathcal{I}(\text{even}) \equiv \varphi$$

Инвариант невыразим
элементарно

Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow \text{even}(x)$$

$$\text{even}(y) \wedge x = S(S(y)) \rightarrow \text{even}(x)$$

$$\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp$$

Язык АД первого порядка:

$$\varphi ::= t = t' \mid \neg\psi \mid \psi \wedge \psi' \mid \psi \vee \psi' \mid \forall x.\psi \mid \exists x.\psi$$

Инвариант выразим
элементарно

$$\mathcal{I}(\text{even}) \equiv \varphi$$

Теория АД допускает устранение кванторов
инвариант невыразим
элементарно

Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow \text{even}(x)$$

$$\text{even}(y) \wedge x = S(S(y)) \rightarrow \text{even}(x)$$

$$\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp$$

Язык АД первого порядка:

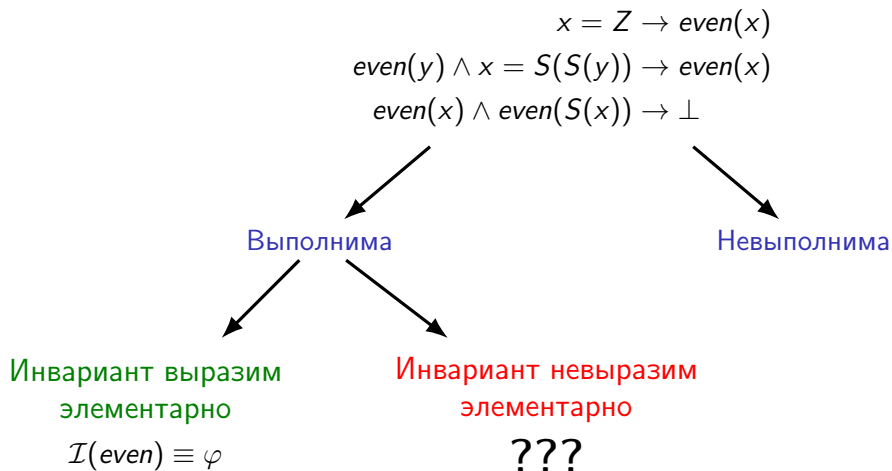
$$\varphi ::= t = t' \mid \neg \psi \mid \psi \wedge \psi' \mid \psi \vee \psi' \mid \forall x. \psi \mid \exists x. \psi$$

Инвариант выразим
элементарно

$$\mathcal{I}(\text{even}) \equiv \varphi$$

Теория АД допускает устранение кванторов
инвариант невыразим
элементарно

Проблема выразимости класса элементарных инвариантов



Проблема выразимости класса элементарных инвариантов

$x = Z \rightarrow \text{even}(x)$
 $\text{even}(y) \wedge x = S(S(y)) \rightarrow \text{even}(x)$
 $\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \text{even}(S(S(x)))$

Вывод инвариантов в языке АТД расходится!
например, Z3/SPACER расходится

Невыполнима

Инвариант выразим
элементарно

$\mathcal{I}(\text{even}) \equiv \varphi$

Инвариант невыразим
элементарно

???

Постановка задачи

Цель работы — предложение новых классов индуктивных инвариантов для программ с АТД и создание для них методов автоматического вывода. **Задачи:**

1. Предложить новые классы индуктивных инвариантов программ с АТД, позволяющие выражать рекурсивные и синхронные отношения
2. Создать методы автоматического вывода инвариантов в новых классах
3. Выполнить пилотную программную реализацию предложенных методов
4. Провести экспериментальное сопоставление реализованного инструмента с существующими на представительном тестовом наборе

Результаты

1. Предложен метод вывода регулярных инвариантов при помощи поиска конечных моделей
2. Предложен метод вывода синхронных регулярных инвариантов при помощи поиска конечных моделей
3. Предложен класс инвариантов, основанный на булевой комбинации элементарных и регулярных инвариантов
Также предложен метод совместного вывода инвариантов в этом классе посредством вывода инвариантов в подклассах
4. Проведено теоретическое сравнение рассмотренных классов инвариантов
Доказаны леммы о «накачке» для элементарных инвариантов
5. Выполнена пилотная реализация предложенных методов на языке $F\#$ в рамках инструмента RINGEN
Разработанный инструмент решил из бенчмарка «Tons of Inductive Problems» в 3.74 раза больше задач, чем наилучший из существующих инструментов

Соответствие результатов паспорту специальности 2.3.5

Результаты соответствуют направлению исследования № 1

- ▶ Модели, **методы и алгоритмы** проектирования, анализа, трансформации, **верификации** и тестирования **программ** и программных систем из паспорта специальности.

1. Впервые предложен класс инвариантов, основанный на булевой комбинации элементарных и регулярных инвариантов
2. Впервые предложен алгоритм вывода инвариантов для программ с АТД, основанный на поиске конечных моделей
3. Предложен новый алгоритм совместного вывода инвариантов в комбинации классов инвариантов на базе методов вывода инвариантов в подклассах
4. Впервые введены и доказаны леммы о «накачке» для языков первого порядка в сигнатуре теории АТД

Публикации по теме диссертации

- [1] Kostyukov Yurii, Mordvinov Dmitry, Fedyukovich Grigory. Beyond the Elementary Representations of Program Invariants over Algebraic Data Types // Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation. — PLDI 2021. — New York, NY, USA : Association for Computing Machinery, 2021. — P. 451–465.
- [2] Kostyukov Yurii, Mordvinov Dmitry, Fedyukovich Grigory. Collaborative Inference of Combined Invariants // Proceedings of 24th International Conference on Logic for Programming, Artificial Intelligence and Reasoning / Ed. by Ruzica Piskac, Andrei Voronkov. — Vol. 94 of EPIc Series in Computing. — EasyChair, 2023. — P. 288–305.
- [3] Автоматическое доказательство корректности программ с динамической памятью / Юрий Олегович Костюков, Константин Аланович Батоев, Дмитрий Александрович Мордвинов и др. // Труды Института системного программирования РАН. — 2019. — Т. 31, № 5. — С. 37–62.
- [4] Генерация слабейших предусловий программ с динамической памятью в символьном исполнении / Александр Владимирович Мисонижник, Юрий Олегович Костюков, Михаил Павлович Костицын и др. // Научно-технический вестник информационных технологий, механики и оптики. — 2022. — Т. 22, № 5. — С. 982–991.

Выступления по теме диссертации

- ▶ Международный семинар HCVS 2021 (28 марта 2021, Люксембург)
- ▶ Семинар компании Huawei (18-19 ноября 2021, Санкт-Петербург)
- ▶ Ежегодный внутренний семинар JetBrains Research (18 декабря 2021, Санкт-Петербург)
- ▶ Конференция PLDI 2021 (23-25 июня 2021, Канада)
- ▶ Внутренний семинар Венского технического университета (3 июня 2022, Австрия)
- ▶ Конференция LPAR 2023 (4-9 июня 2023, Колумбия)

Разработанный инструмент в 2021 и 2022 годах занял, соответственно, 2 и 1 место на АТД секции международных соревнований CHC-COMP.

Результаты

1. Предложен метод вывода регулярных инвариантов при помощи поиска конечных моделей
2. Предложен метод вывода синхронных регулярных инвариантов при помощи поиска конечных моделей
3. Предложен класс инвариантов, основанный на булевой комбинации элементарных и регулярных инвариантов
Также предложен метод совместного вывода инвариантов в этом классе посредством вывода инвариантов в подклассах
4. Проведено теоретическое сравнение рассмотренных классов инвариантов
Доказаны леммы о «накачке» для элементарных инвариантов
5. Выполнена пилотная реализация предложенных методов на языке $F\#$ в рамках инструмента RINGEN
Разработанный инструмент решил из бенчмарка «Tons of Inductive Problems» в 3.74 раза больше задач, чем наилучший из существующих инструментов

Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов

$$\top \rightarrow \text{even}(Z)$$

$$\text{even}(y) \rightarrow \text{even}(S(S(y)))$$

$$\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp$$

АТД ограничения устранены

Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$

Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y.(\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x.(\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y.(\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x.(\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

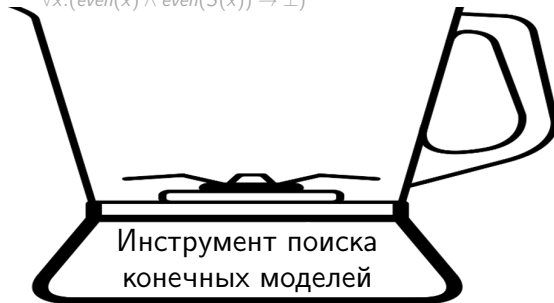
$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y.(\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x.(\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \top \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Инструмент поиска
конечных моделей

Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} & \neg \rightarrow \text{even}(Z)) \wedge \\ & \forall y. (\text{even}(y) \rightarrow \text{even}(S(S(y)))) \wedge \\ & \forall x. (\text{even}(x) \wedge \text{even}(S(x)) \rightarrow \perp) \end{aligned}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



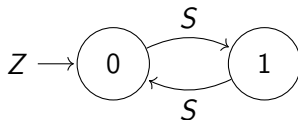
Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



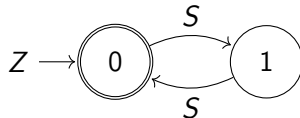
Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(\text{even}) = \{0\}$$



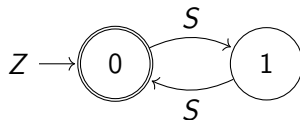
Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$

$$\mathcal{M}(Z) = 0$$

$$\mathcal{M}(S)(x) = 1 - x$$

$$\mathcal{M}(even) = \{0\}$$



$$\mathcal{I}(even) = \mathcal{L}(\mathcal{A})$$



Вывод синхронных регулярных инвариантов при помощи поиска конечных моделей

Регулярные языки не позволяют представлять синхронные отношения:

$$\begin{aligned}\top &\rightarrow It(Z, S(x)) \\ It(x, y) &\rightarrow It(S(x), S(y)) \\ It(x, y) \wedge It(y, x) &\rightarrow \perp\end{aligned}$$

Идея: породить декларативное описание синхронного автомата

$$\begin{aligned}R(q) &\rightarrow R(p(d(f, g, q), d(f, g, q))) \\ R(p(q_1, q_2)) &\rightarrow (F(q_1) \rightarrow F(d(S, S, q_2))) \\ &\dots\end{aligned}$$

Из модели можно извлечь определение синхронного автомата

$$A = \langle \{0, 1\}, \Sigma_F^{\leq 2}, \{1\}, \Delta \rangle$$

$$\begin{array}{lll}\langle Z, Z \rangle \mapsto 0 & Z \mapsto 0 & S(q) \mapsto 0 \\ \langle Z, S \rangle(q) \mapsto 1 & \langle S, Z \rangle(q) \mapsto 0 & \langle S, S \rangle(q) \mapsto q\end{array}$$

$$\mathcal{L}(A) = \{ \langle S^n(Z), S^m(Z) \rangle \mid n < m \}$$

Совместный вывод комбинированных инвариантов

FREQHORN

DUALITY

QARMC

IPROVER

CVC5

Z3/SPACER

VAMPIRE

E

ELDARICA

GOLEM

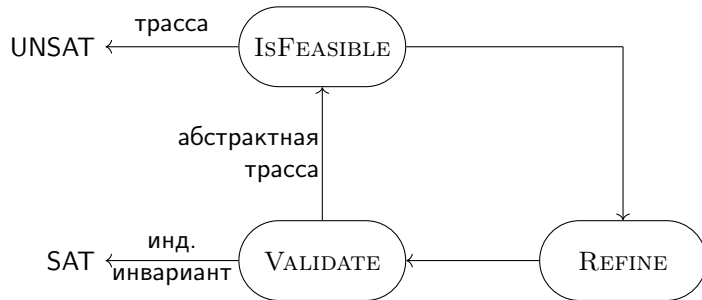
HOICE

ZIPPERPOSITION

Хорн решатели

**Инструменты
вывода теорем**

Совместный вывод комбинированных инвариантов



Хорн решатели

Инструменты
вывода теорем

IPROVER

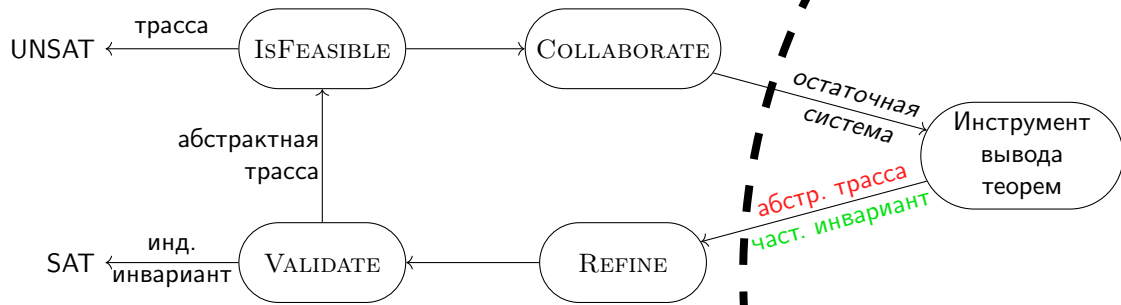
CVC5

VAMPIRE

E

ZIPPERPOSITION

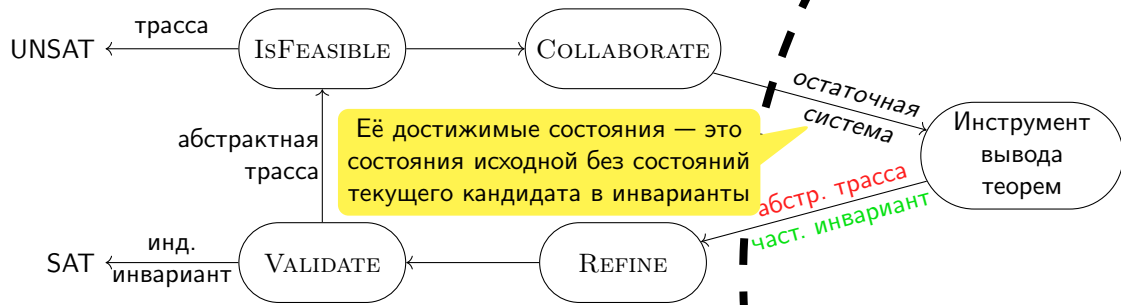
Совместный вывод комбинированных инвариантов



Хорн решатели

Инструменты
вывода теорем

Совместный вывод комбинированных инвариантов



Хорн решатели

Инструменты
вывода теорем

Теоретическое сравнение классов инвариантов

Класс Свойство	ELEM	SIZEELEM	REG	REG ₊	REG _×	ELEMREG
Замкнут по \cap	Да	Да	Да ¹	Да ²	Да ²	Да
Замкнут по \cup	Да	Да	Да ¹	Да ²	Да ²	Да
Замкнут по \setminus	Да	Да	Да ¹	Да ²	Да ²	Да
Разрешимо $\bar{t} \in I$	Да ³	Да ⁴	Да ⁵	Да ⁷	Да ⁹	Да ¹⁰
Разрешимо $I = \emptyset$	Да ³	Да ⁴	Да ⁶	Да ⁸	Да ⁹	Да ¹⁰
Выразимы рекурсивные отношения	Нет	Частично	Да	Да	Да	Да
Выразимы синхронные отношения	Да	Да	Нет	Частично	Да	Да

Класс	ELEM	SIZEELEM	REG	REG ₊	REG _×	ELEMREG
ELEM	\emptyset	\emptyset	$lr^{1,4,5}$	$lr^{1,5}$	lr^1	\emptyset
SIZEELEM	∞	\emptyset	$lr^{1,4,5}$	$lr^{1,5}$	lr^1	lt^3
REG	$even^2$	$even^2$	\emptyset	\emptyset^4	$\emptyset^{4,5}$	\emptyset
REG ₊	$even^{2,7}$	$even^{2,4}$	∞^4	\emptyset	\emptyset^5	lt^3
REG _×	$even^{2,4,5}$	$even^{2,4,5}$	$\infty^{4,5}$	∞^5	\emptyset	$lt^{3,5}$
ELEMREG	∞	$even^2$	∞	$lr^{1,5}$	lr^1	\emptyset

Реализация

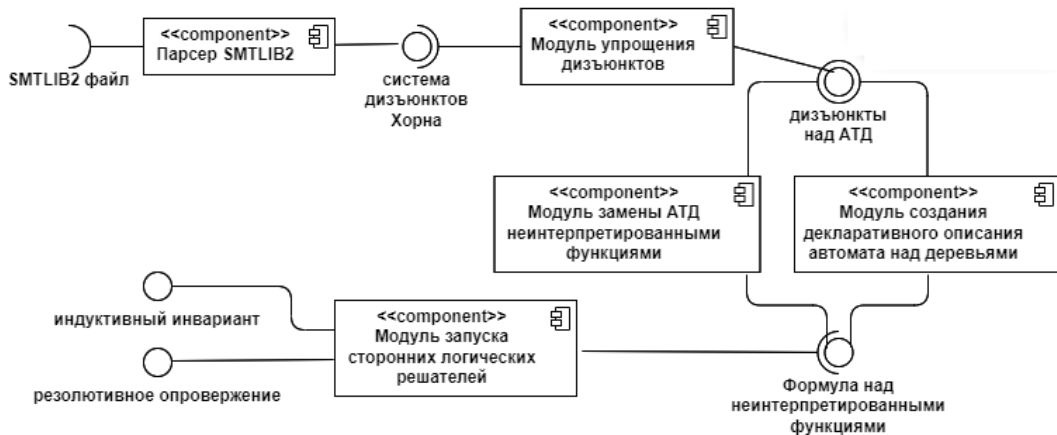


Рис.: Хорн-решатель RINGEN: <https://github.com/Columpio/RInGen>

Сравнение Хорн-решателей с поддержкой АД

Инструмент	Класс инвариантов	Метод	Возвращает инвариант	Полностью автоматический
SPACER	ELEM	IC3/PDR	Да	Да
RACER	CATELEM	IC3/PDR	Нет	Нет
ELDARICA	SIZEELEM	CEGAR	Да	Да
VERICAT	–	Трансф.	Нет	Да
HoICE	ELEM	ICE	Да	Да
RCHC	REG ₊	ICE	Да	Да
RINGEN(CVC5)	REG	Трансф. + FMF	Да	Да
RINGEN(VAMPIRE)	–	Трансф. + Насыщение	Нет	Да
RINGEN-SYNC	REG _x	Трансф. + FMF	Да	Да
RINGEN-CICI(CVC5)	ELEMREG	CEGAR(\emptyset)	Да	Да
RINGEN-CICI(VAMPIRE)	–	CEGAR(\emptyset)	Нет	Да

Эксперименты

Инструмент	SAT	UNSAT
RACER	26	22
ELДАРICA	46	12
VERICAT	16	10
CVC5-IND	0	13
RInGen(CVC5)	25	21
RInGen(VAMPIRE)	135	46
RInGen-Sync	43	21
RInGen-CICI(CVC5)	117	19
RInGen-CICI(VAMPIRE)	189	28

