# 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

# Верификация программ путём вывода индуктивных инвариантов

$$
\begin{array}{l}
x, y := 0, 0 \\
\texttt{while} \;\; * \;\; \texttt{do} \\
\qquad y := y + x \\
\qquad x := x + 1 \\
\texttt{assert}(y \geq 0)
\end{array}
$$

# Верификация программ путём вывода индуктивных инвариантов

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

```
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

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

```
while * do
    y := y + x
    x := x + 1
```

$$\{y \geq 0\}$$

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

$$\begin{aligned} x = 0 \land y = 0 \quad & \to \varphi(x, y) \\ \varphi(x, y) \land x' = x + 1 \land y' = y + x \quad & \to \varphi(x', y') \\ \varphi(x, y) \quad & \to y \geq 0 \end{aligned}$$

# Верификация программ путём вывода индуктивных инвариантов

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

```
while * do
    y := y + x
    x := x + 1
```

$$\{y \geq 0\}$$

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

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

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

# Верификация программ путём вывода индуктивных инвариантов

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

```
while * do
    y := y + x
    x := x + 1
```

$$\{y \geq 0\}$$

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

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

$$VC := \begin{cases} \forall x, y.\big(x = 0 \land y = 0 \rightarrow y \geq 0\big) \land \\ \forall x, y, x', y'.\big(y \geq 0 \land x' = x + 1 \land y' = y + x \rightarrow y' \geq 0\big) \land \\ \forall x, y.\big(y \geq 0 \rightarrow y \geq 0\big) \end{cases}$$

# Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

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

$VC \longrightarrow$ SMT решатель $\longrightarrow$ ✓ (безопасно)

SMT решатель $\longrightarrow$ ✗ ($\varphi(\overline{x})$ — не инд. инв.)

$$VC := \begin{cases} \forall x, y. \Big( x = 0 \land y = 0 \rightarrow y \geq 0 \Big) \land \\ \forall x, y, x', y'. \Big( y \geq 0 \land x' = x + 1 \land y' = y + x \rightarrow y' \geq 0 \Big) \land \\ \forall x, y. \Big( y \geq 0 \rightarrow y \geq 0 \Big) \end{cases}$$

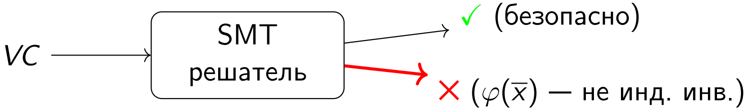
# Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

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

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




$$VC := \begin{cases} \forall x, y. \Big( x = 0 \land y = 0 \rightarrow y \geq 0 \Big) \land \\ \forall x, y, x', y'. \Big( y \geq 0 \land x' = x + 1 \land y' = y + x \rightarrow y' \geq 0 \Big) \land \\ \forall x, y. \Big( y \geq 0 \rightarrow y \geq 0 \Big) \end{cases}$$

# Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

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

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

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

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

# Верификация программ путём вывода индуктивных инвариантов

Как доказать корректность этой тройки Хоара?

При помощи *пользовательского* **индуктивного инварианта** $\varphi$

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

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

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

**SMT решатель**: Да, эта формула является индуктивным инвариантом

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

# Дизъюнкты Хорна с ограничениями

**Как автоматизировать** вывод индуктивных инвариантов?

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

# Дизъюнкты Хорна с ограничениями

**Как автоматизировать** вывод индуктивных инвариантов?

Заменить пользовательскую формулу на **неинтерпретированный** символ $I$

$$x = 0 \wedge y = 0 \to I(x, y)$$
$$I(x, y) \wedge x' = x + 1 \wedge y' = y + x \to I(x', y')$$
$$I(x, y) \to y \geq 0$$

# Дизъюнкты Хорна с ограничениями

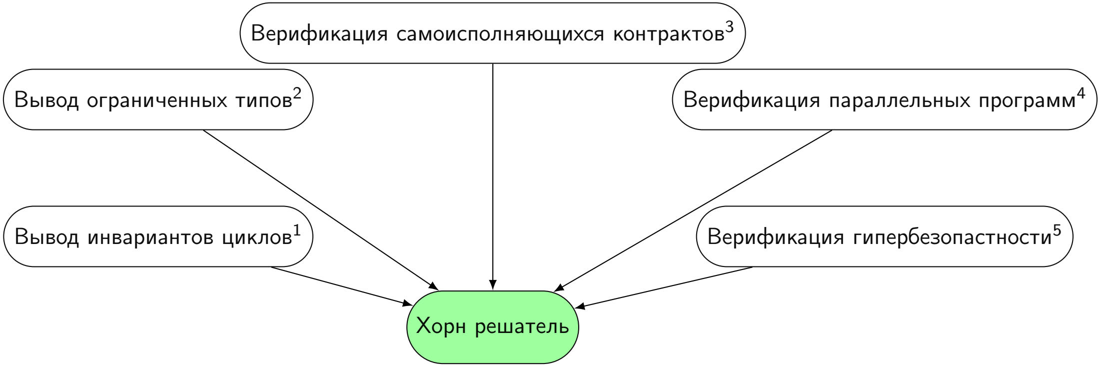
**Как автоматизировать** вывод индуктивных инвариантов?

Заменить пользовательскую формулу на **неинтерпретированный** символ $I$

Дизъюнкты Хорна с ограничениями

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

# 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:

$$\top \rightarrow drop(Z, xs, xs)$$
$$\top \rightarrow drop(S(n), nil, nil)$$
$$drop(n, xs, rs) \rightarrow drop(S(n), cons(x, xs), rs)$$
$$\neg(xs = nil) \wedge drop(n, xs, ys) \wedge drop(S(n), xs, ys) \rightarrow \bot$$

# Дизъюнкты Хорна формально

**Дизъюнкт Хорна** $C$ — это формула первого порядка следующего вида:

$$\varphi \wedge P_1(\overline{x}_1) \wedge \ldots \wedge P_n(\overline{x}_n) \rightarrow H$$

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

**Система дизъюнктов Хорна** — это конъюнкция дизъюнктов Хорна

# 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 $\mathcal{I} \models \mathcal{S}$.

# 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 $\mathcal{I} \models \mathcal{S}$.

$$x = Z \land y = S(Z) \to inc(x, y)$$
$$x' = S(x) \land y' = S(y) \land inc(x, y) \to inc(x', y')$$
$$x = y \land inc(x, y) \to \bot$$

$$\mathcal{I}_1 = \mathcal{H}\Big\{inc \mapsto \{(x, y) \mid y = S(x)\Big\}$$
$$\mathcal{I}_2 = \mathcal{H}\Big\{inc \mapsto \{(x, y) \mid x \neq y \qquad \Big\}$$
$$\mathcal{I}_3 = \ldots$$

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 $\mathcal{I} \models \mathcal{S}$.

$$x = Z \land y = S(Z) \to inc(x, y)$$

$$x' = S(x) \land y' = S(y) \land inc(x, y) \to inc(x', y')$$

$$x = y \land inc(x, y) \to \bot$$

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

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

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

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 $\mathcal{I} \models \mathcal{S}$.

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

$$\mathcal{I}_1 = \mathcal{H}\left\{ inc \mapsto \boxed{y = S(x)} \right\}$$
$$\mathcal{I}_2 = \mathcal{H}\left\{ inc \mapsto \boxed{\neg(x = y)} \right\}$$
$$\mathcal{I}_3 = \ldots$$

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

$$
\begin{aligned}
x = Z &\to even(x) \\
even(y) \wedge x = S(S(y)) &\to even(x) \\
even(x) \wedge even(S(x)) &\to \bot
\end{aligned}
$$

# Проблема выразимости класса элементарных инвариантов

$$\begin{aligned} x = Z &\rightarrow even(x) \\ even(y) \land x = S(S(y)) &\rightarrow even(x) \\ even(x) \land even(S(x)) &\rightarrow \bot \end{aligned}$$

Невыполнима

# Проблема выразимости класса элементарных инвариантов

$$
\begin{aligned}
x = Z &\to even(x) \\
even(y) \wedge x = S(S(y)) &\to even(x) \\
even(x) \wedge even(S(x)) &\to \bot
\end{aligned}
$$

Выполнима

Невыполнима

# Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow even(x)$$
$$even(y) \land x = S(S(y)) \rightarrow even(x)$$
$$even(x) \land even(S(x)) \rightarrow \bot$$

Выполнима

Невыполнима

Инвариант выразим элементарно

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

# Проблема выразимости класса элементарных инвариантов

$$x = Z \to even(x)$$
$$even(y) \land x = S(S(y)) \to even(x)$$
$$even(x) \land even(S(x)) \to \bot$$

Выполнима

Невыполнима

Инвариант выразим элементарно

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

Инвариант невыразим элементарно

# Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow even(x)$$
$$even(y) \wedge x = S(S(y)) \rightarrow even(x)$$
$$even(x) \wedge even(S(x)) \rightarrow \bot$$

Язык АТД 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

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

Инвариант невыразим элементарно

# Проблема выразимости класса элементарных инвариантов

$$x = Z \to even(x)$$
$$even(y) \land x = S(S(y)) \to even(x)$$
$$even(x) \land even(S(x)) \to \bot$$

Язык АТД первого порядка:

$$\varphi ::= t = t' \mid \neg\psi \mid \psi \land \psi' \mid \psi \lor \psi' \mid \cancel{\forall x . \psi \mid \exists x . \psi}$$

Теория АТД допускает устранение кванторов

Инвариант выразим элементарно

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

Инвариант невыразим элементарно

# Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow even(x)$$
$$even(y) \land x = S(S(y)) \rightarrow even(x)$$
$$even(x) \land even(S(x)) \rightarrow \bot$$

а

Язык АТД первого порядка:
$$\varphi ::= t = t' \mid \neg\psi \mid \psi \land \psi' \mid \psi \lor \psi' \mid \cancel{\forall x \,.\, \psi \mid \exists x \,.\, \psi}$$

Теория АТД допускает устранение кванторов

Инвариант выразим элементарно

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

Инвариант невыразим элементарно

# Проблема выразимости класса элементарных инвариантов

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

Выполнима

Невыполнима

Инвариант выразим элементарно

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

Инвариант невыразим элементарно

???

# Проблема выразимости класса элементарных инвариантов

$$x = Z \rightarrow even(x)$$
$$even(y) \land x = S(S(y)) \rightarrow even(x)$$
$$even(x) \land even(S(x)) \rightarrow \bot$$

Выполнима

Невыполнима

Инвариант выразим элементарно

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

Инвариант невыразим элементарно

???

**Вывод инвариантов в языке АТД расходится!**

например, Z3/Spacer расходится

# 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.

# Научная новизна

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 внутренней 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

$$
\begin{aligned}
\top &\rightarrow \mathit{even}(Z) \\
\mathit{even}(y) &\rightarrow \mathit{even}(S(S(y))) \\
\mathit{even}(x) \wedge \mathit{even}(S(x)) &\rightarrow \bot
\end{aligned}
$$

АТД ограничения устранены

# Вывод регулярных инвариантов при помощи поиска конечных моделей

## Система дизъюнктов как формула ЛПП

$$
\begin{aligned}
\top \rightarrow \mathit{even}(Z)) \land \\
\forall y.(\mathit{even}(y) \rightarrow \mathit{even}(S(S(y)))) \land \\
\forall x.(\mathit{even}(x) \land \mathit{even}(S(x)) \rightarrow \bot)
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

## Система дизъюнктов как формула ЛПП

$$\begin{aligned}
\top &\to even(Z)) \land \\
\forall y.(even(y) &\to even(S(S(y)))) \land \\
\forall x.(even(x) \land even(S(x)) &\to \bot)
\end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

## Система дизъюнктов как формула ЛПП

$$
\begin{aligned}
\top \rightarrow even(Z)) \land \\
\forall y.(even(y) \rightarrow even(S(S(y)))) \land \\
\forall x.(even(x) \land even(S(x)) \rightarrow \bot)
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

## Система дизъюнктов как формула ЛПП

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

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\top \to even(Z)) \land$$
$$\forall y.(even(y) \to even(S(S(y)))) \land$$
$$\forall x.(even(x) \land even(S(x)) \to \bot)$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} \top \rightarrow even(Z)) \land \\ \forall y.(even(y) \rightarrow even(S(S(y)))) \land \\ \forall x.(even(x) \land even(S(x)) \rightarrow \bot) \end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} \top \rightarrow even(Z)) \land \\ \forall y.(even(y) \rightarrow even(S(S(y)))) \land \\ \forall x.(even(x) \land even(S(x)) \rightarrow \bot) \end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} \top \rightarrow even(Z)) \land \\ \forall y.(even(y) \rightarrow even(S(S(y)))) \land \\ \forall x.(even(x) \land even(S(x)) \rightarrow \bot) \end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

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

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

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

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} \top \rightarrow \mathit{even}(Z)) \wedge \\ \forall y.(\mathit{even}(y) \rightarrow \mathit{even}(S(S(y)))) \wedge \\ \forall x.(\mathit{even}(x) \wedge \mathit{even}(S(x)) \rightarrow \bot) \end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

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

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned}\top \rightarrow \mathit{even}(Z)) \land \\ \forall y.(\mathit{even}(y) \rightarrow \mathit{even}(S(S(y)))) \land \\ \forall x.(\mathit{even}(x) \land \mathit{even}(S(x)) \rightarrow \bot)\end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} \top \rightarrow \mathit{even}(Z)) \wedge \\ \forall y.(\mathit{even}(y) \rightarrow \mathit{even}(S(S(y)))) \wedge \\ \forall x.(\mathit{even}(x) \wedge \mathit{even}(S(x)) \rightarrow \bot) \end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$
\begin{aligned}
\top \rightarrow even(Z)) \land \\
\forall y.(even(y) \rightarrow even(S(S(y)))) \land \\
\forall x.(even(x) \land even(S(x)) \rightarrow \bot)
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\top \rightarrow even(Z)) \wedge$$
$$\forall y.(even(y) \rightarrow even(S(S(y)))) \wedge$$
$$\forall x.(even(x) \wedge even(S(x)) \rightarrow \bot)$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$
\begin{aligned}
\top \to even(Z)) \wedge \\
\forall y.(even(y) \to even(S(S(y)))) \wedge \\
\forall x.(even(x) \wedge even(S(x)) \to \bot)
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\begin{aligned} \top \to even(Z)) \land \\ \forall y.(even(y) \to even(S(S(y)))) \land \\ \forall x.(even(x) \land even(S(x)) \to \bot) \end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$
\begin{aligned}
\top \to even(Z)) \wedge \\
\forall y.(even(y) \to even(S(S(y)))) \wedge \\
\forall x.(even(x) \wedge even(S(x)) \to \bot)
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$\top \rightarrow even(Z)) \land$$
$$\forall y.(even(y) \rightarrow even(S(S(y)))) \land$$
$$\forall x.(even(x) \land even(S(x)) \rightarrow \bot)$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

Система дизъюнктов как формула ЛПП

$$
\begin{aligned}
\top \to \mathit{even}(Z)) \land \\
\forall y.(\mathit{even}(y) \to \mathit{even}(S(S(y)))) \land \\
\forall x.(\mathit{even}(x) \land \mathit{even}(S(x)) \to \bot)
\end{aligned}
$$

# 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

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# 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

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0,1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# 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

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$
\begin{aligned}
|\mathcal{M}|_{Nat} &= \{0, 1\} \\
\mathcal{M}(Z) &= 0 \\
\mathcal{M}(S)(x) &= 1 - x \\
\mathcal{M}(even) &= \{0\}
\end{aligned}
$$

# Вывод регулярных инвариантов при помощи поиска конечных моделей

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

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

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

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

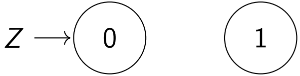

# Вывод регулярных инвариантов при помощи поиска конечных моделей

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

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

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

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



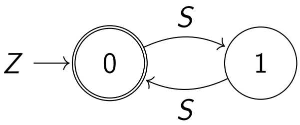

# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$
$$\mathcal{M}(Z) = 0$$
$$\mathcal{M}(S)(x) = 1 - x$$
$$\mathcal{M}(even) = \{0\}$$

$$\begin{tikzcd} Z \arrow[r] & 0 \arrow[r, bend left, "S"] & 1 \arrow[l, bend left, "S"] \end{tikzcd}$$

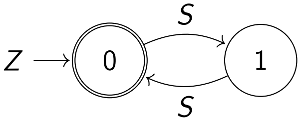
# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$
$$\mathcal{M}(Z) = 0$$
$$\mathcal{M}(S)(x) = 1 - x$$
$$\mathcal{M}(even) = \{0\}$$




# Вывод регулярных инвариантов при помощи поиска конечных моделей

$$|\mathcal{M}|_{Nat} = \{0, 1\}$$
$$\mathcal{M}(Z) = 0$$
$$\mathcal{M}(S)(x) = 1 - x$$
$$\mathcal{M}(even) = \{0\}$$




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

# Вывод синхронных регулярных инвариантов при помощи поиска конечных моделей

Регулярные языки не позволяют представлять синхронные отношения:

$$\top \to lt(Z, S(x))$$
$$lt(x, y) \to lt(S(x), S(y))$$
$$lt(x, y) \land lt(y, x) \to \bot$$

Идея: порождать декларативное описание синхронного автомата

$$R(q) \to R(p(d(f, g, q), d(f, g, q)))$$
$$R(p(q_1, q_2)) \to \Big(F(q_1) \to F(d(S, S, q_2))\Big)$$
$$\ldots$$

Из модели можно извлечь определение синхронного автомата

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

$$\langle Z, Z\rangle \mapsto 0 \qquad Z \mapsto 0 \qquad S(q) \mapsto 0$$
$$\langle Z, S\rangle(q) \mapsto 1 \qquad \langle S, Z\rangle(q) \mapsto 0 \qquad \langle S, S\rangle(q) \mapsto q$$

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

# Совместный вывод комбинированных инвариантов

FreqHorn

Duality

QARMC

Z3/Spacer

Eldarica

Golem

Hoice

Хорн решатели

iProver

cvc5

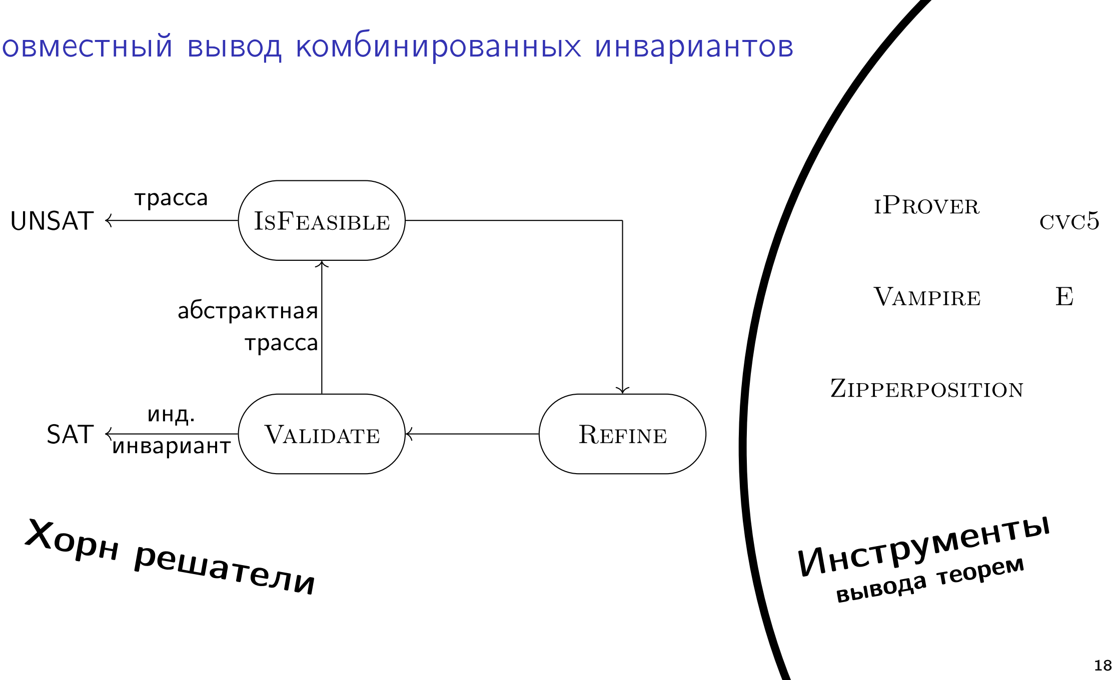
Vampire

E

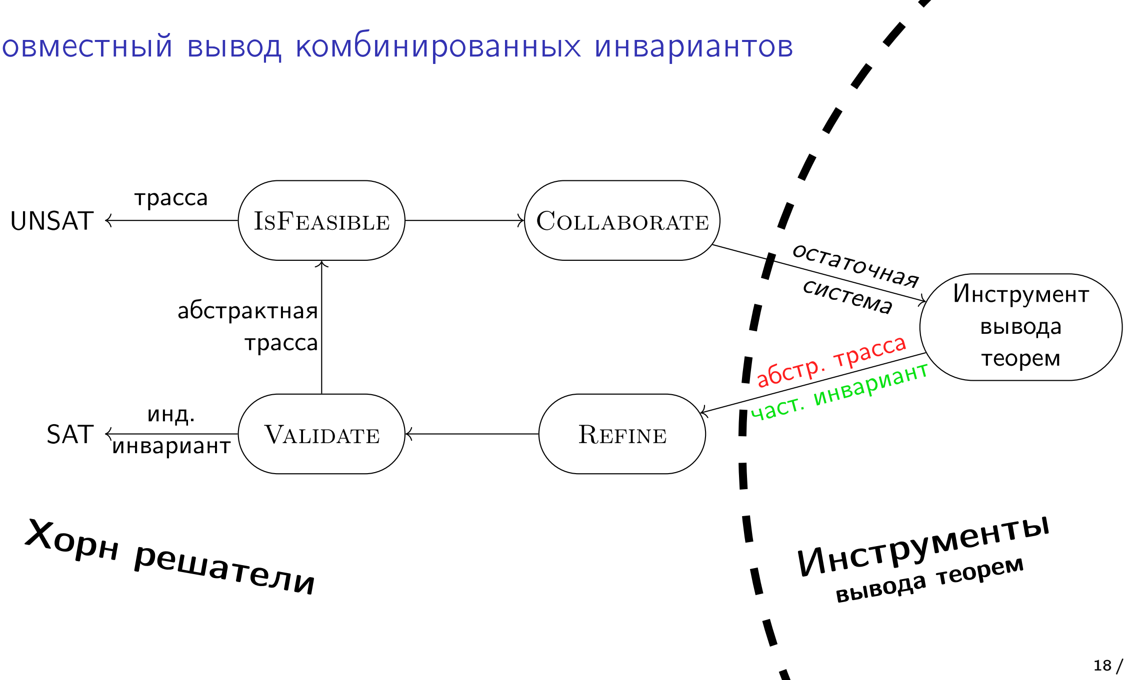
Zipperposition

Инструменты
вывода теорем

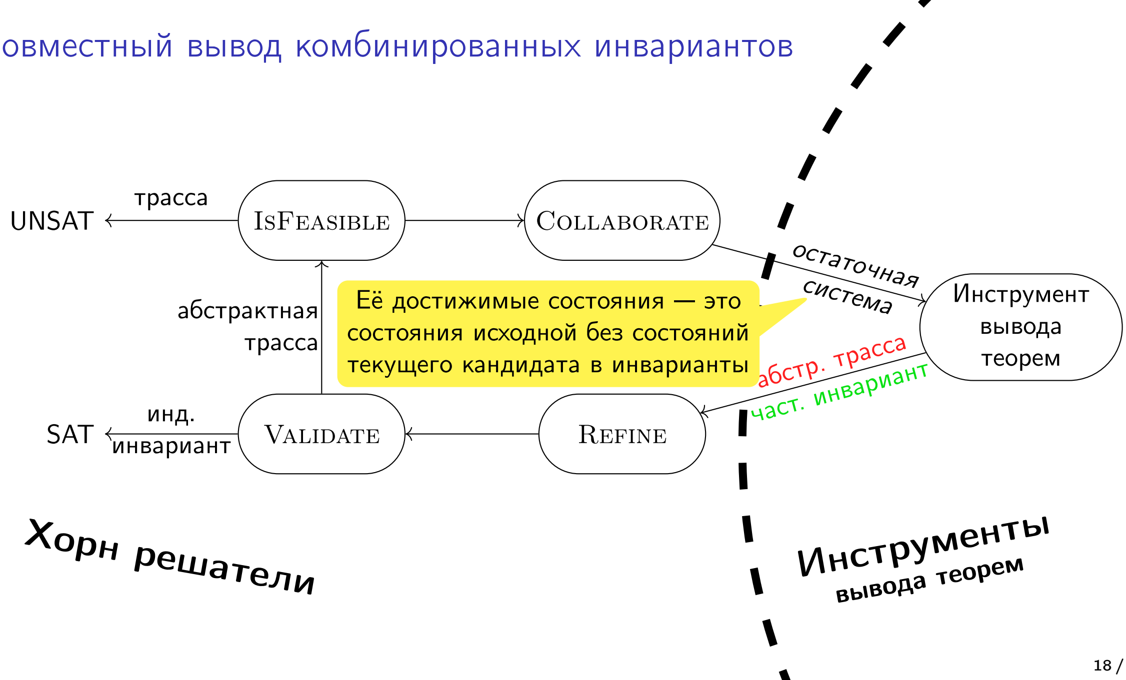
# Совместный вывод комбинированных инвариантов



# Совместный вывод комбинированных инвариантов



# Совместный вывод комбинированных инвариантов



# Теоретическое сравнение классов инвариантов

| Свойство \ Класс | Elem | SizeElem | Reg | $\text{Reg}_+$ | $\text{Reg}_\times$ | 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 $\overline{t} \in I$ | Да[3] | Да[4] | Да[5] | Да[7] | Да[9] | Да[10] |
| Разрешимо $I = \varnothing$ | 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 | $\text{Reg}_+$ | $\text{Reg}_\times$ | ElemReg |
|---|---|---|---|---|---|---|
| Elem | $\varnothing$ | $\varnothing$ | $lr$[1,4,5] | $lr$[1,5] | $lr$[1] | $\varnothing$ |
| SizeElem | $\infty$ | $\varnothing$ | $lr$[1,4,5] | $lr$[1,5] | $lr$[1] | $lt$[3] |
| Reg | $even$[2] | $even$[2] | $\varnothing$ | $\varnothing$[4] | $\varnothing$[4,5] | $\varnothing$ |
| $\text{Reg}_+$ | $even$[2,7] | $even$[2,4] | $\infty$[4] | $\varnothing$ | $\varnothing$[5] | $lt$[3] |
| $\text{Reg}_\times$ | $even$[2,4,5] | $even$[2,4,5] | $\infty$[4,5] | $\infty$[5] | $\varnothing$ | $lt$[3,5] |
| ElemReg | $\infty$ | $even$[2] | $\infty$ | $lr$[1,5] | $lr$[1] | $\varnothing$ |

# Реализация

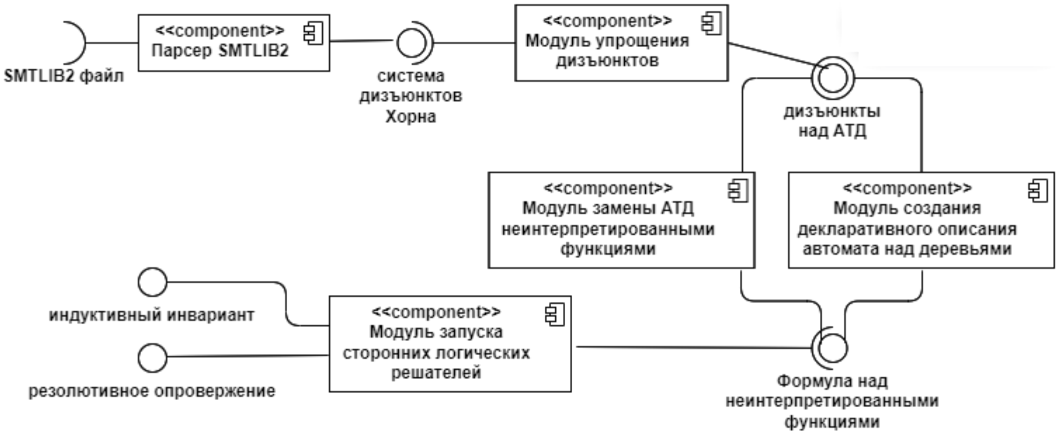



Рис.: Хорн-решатель RInGen: `https://github.com/Columpio/RInGen`

# Сравнение Хорн-решателей с поддержкой АТД

| 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 | $\text{Reg}_{+}$ | ICE | Да | Да |
| RInGen(cvc5) | Reg | Трансф. + FMF | Да | Да |
| RInGen(Vampire) | – | Трансф. + Насыщение | Нет | Да |
| RInGen-Sync | $\text{Reg}_{\times}$ | Трансф. + FMF | Да | Да |
| RInGen-CICI(cvc5) | ElemReg | CEGAR($\mathcal{O}$) | Да | Да |
| RInGen-CICI(Vampire) | – | CEGAR($\mathcal{O}$) | Нет | Да |

# Эксперименты

| Инструмент | SAT | UNSAT |
|---|---|---|
| Racer | 26 | 22 |
| Eldarica | 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 |