



УНИВЕРСИТЕТ ИТМО

Применение методов программирования в ограничениях для генерации конечно-автоматных моделей контроллеров в задачах логического управления

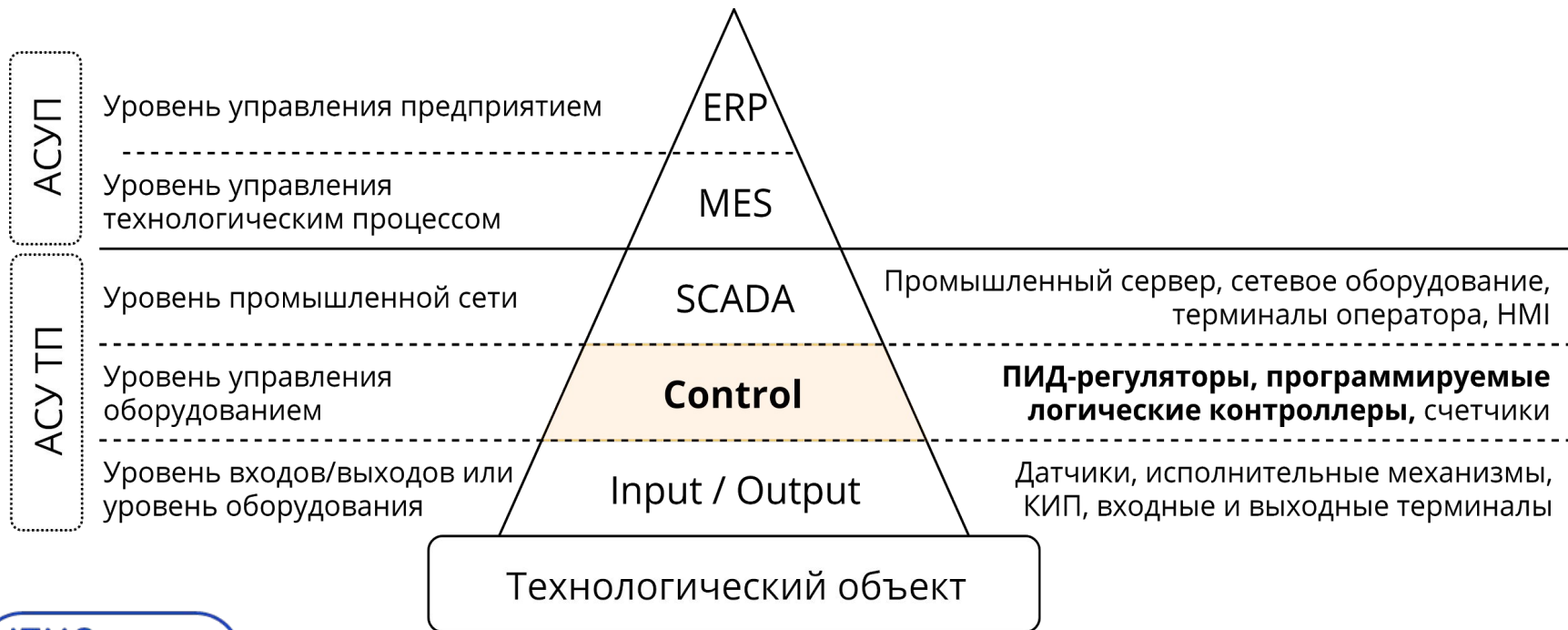
Студент: Чухарев Константин Игоревич, Р3435

Руководитель: Ульянов Владимир Игоревич, к.т.н., доцент кафедры КТ

Консультант: Чивилихин Даниил Сергеевич, к.т.н., научный сотрудник кафедры КТ

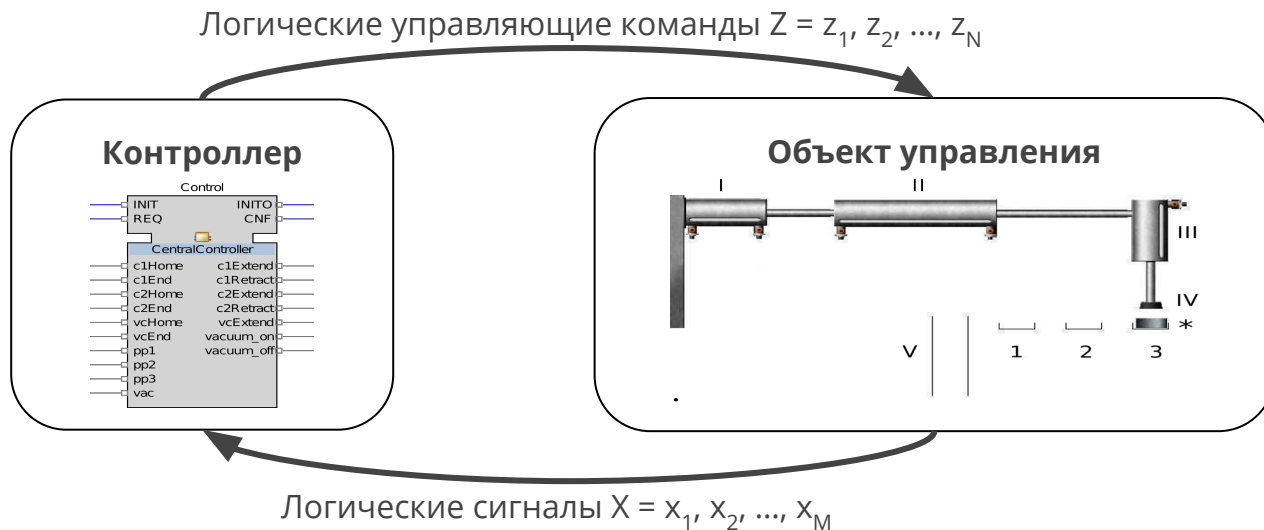
Санкт-Петербург, 2018

Пирамида автоматизации



Логическое управление

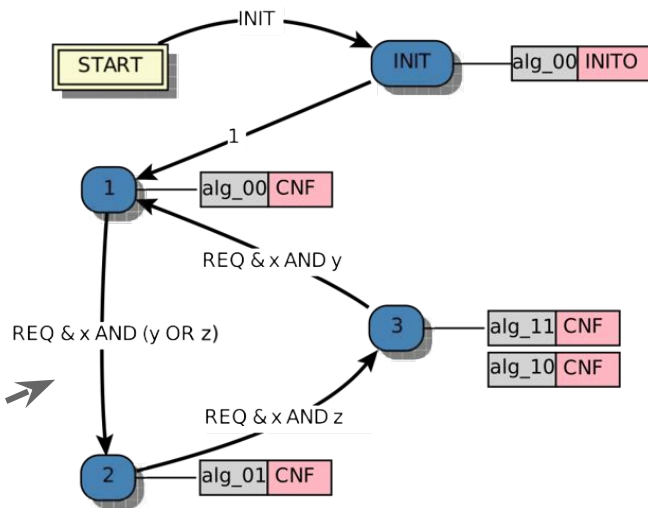
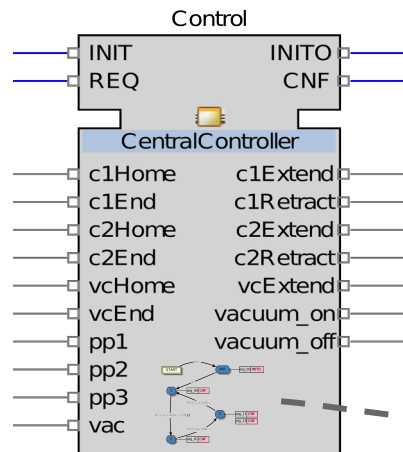
Особый вид управления*, в котором контроллер оперирует *двоичными* управляющими сигналами.



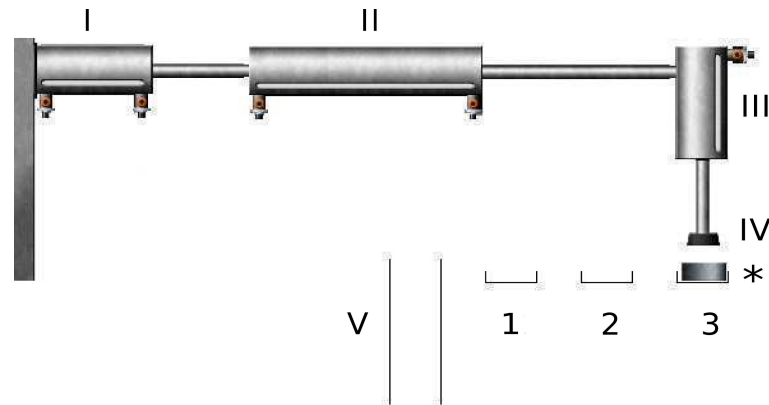
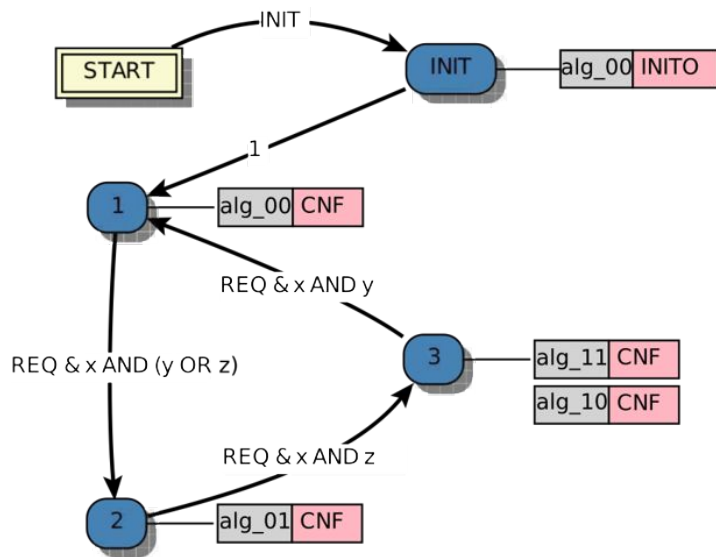
Автоматное программирование

Разработка контроллера \Rightarrow построение конечного автомата

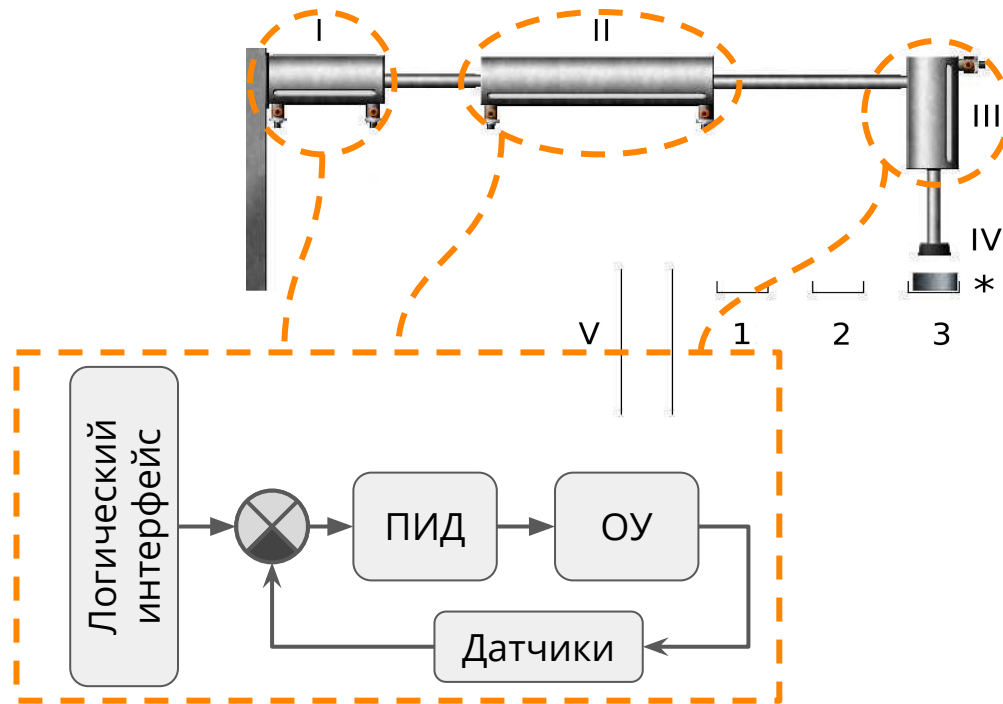
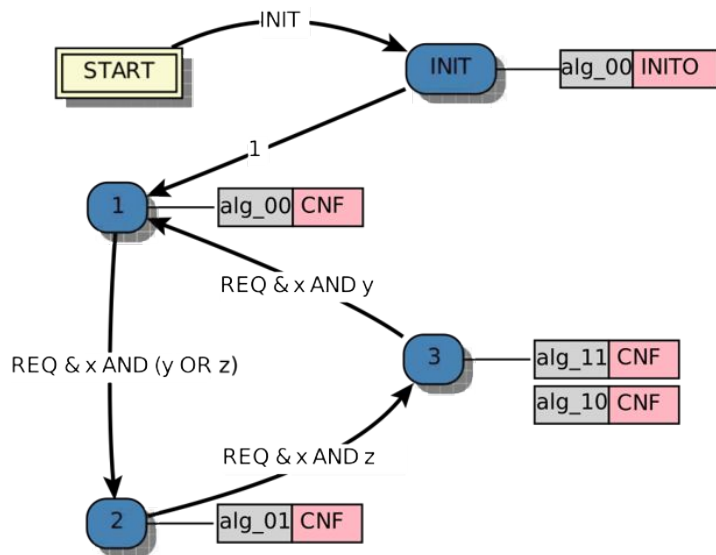
Такой подход применяется в стандарте распределенных систем управления и автоматизации IEC 61499*.



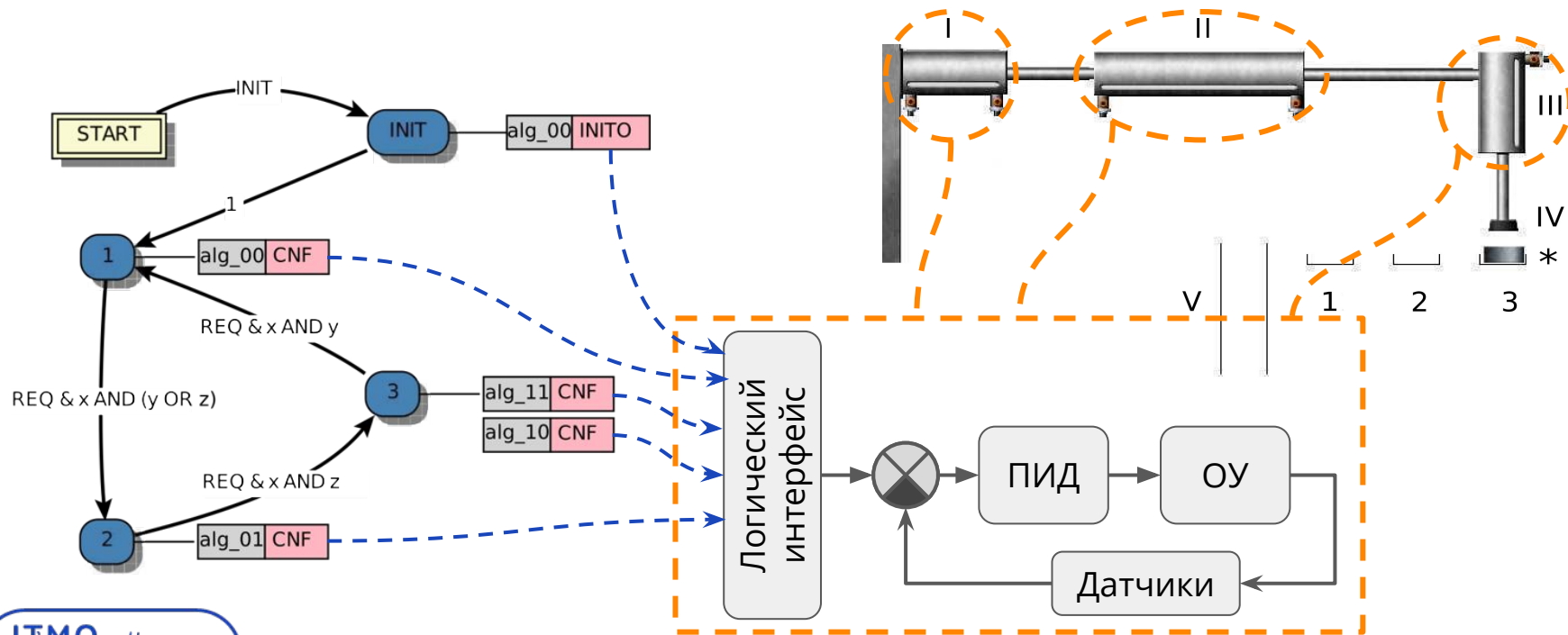
Автоматное программирование



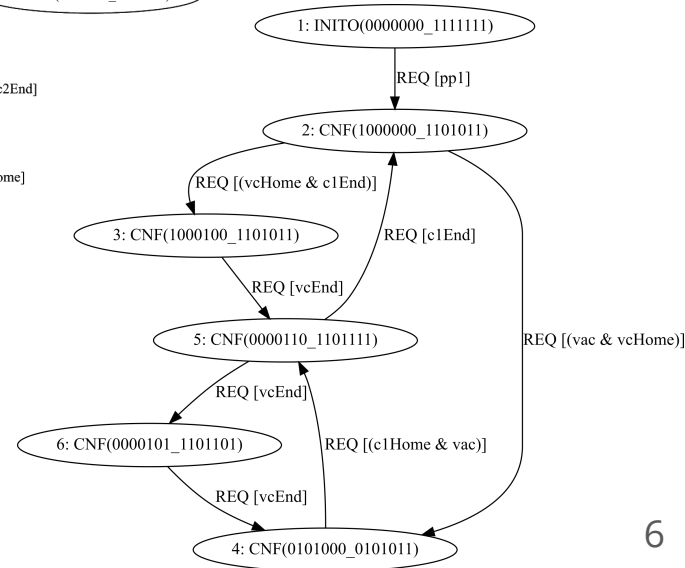
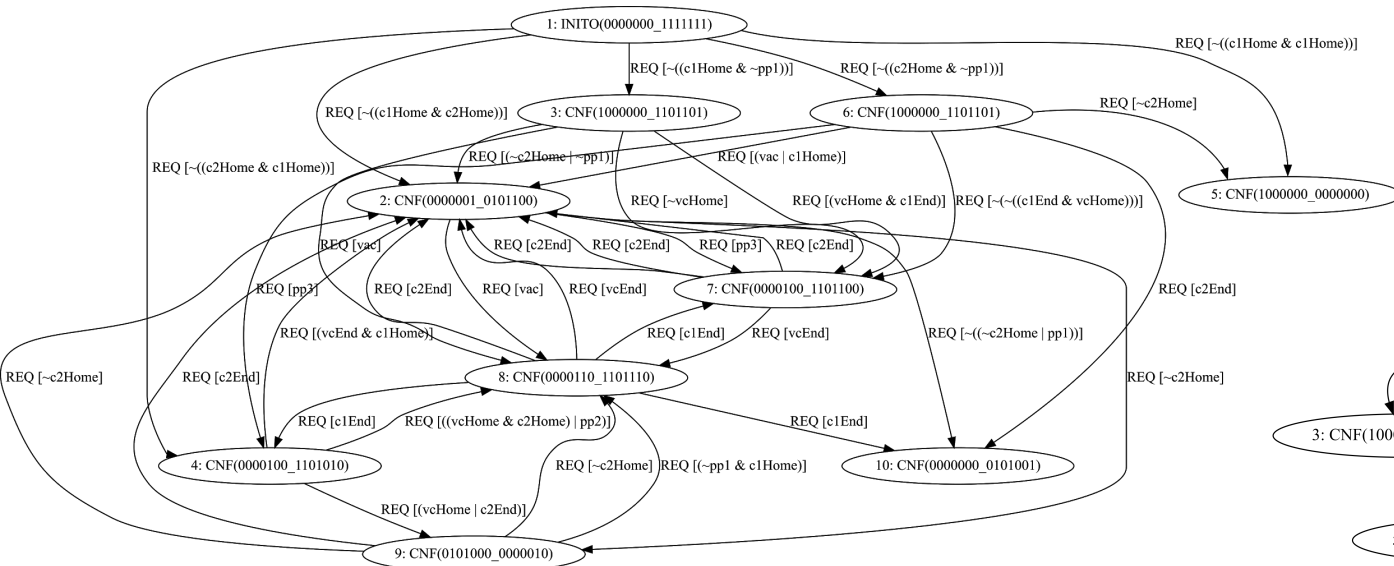
Автоматное программирование



Автоматное программирование



Минимальные автоматы



Цель и задачи

Цель: разработка методов генерации минимальных конечно-автоматных моделей контроллеров по примерам поведения.

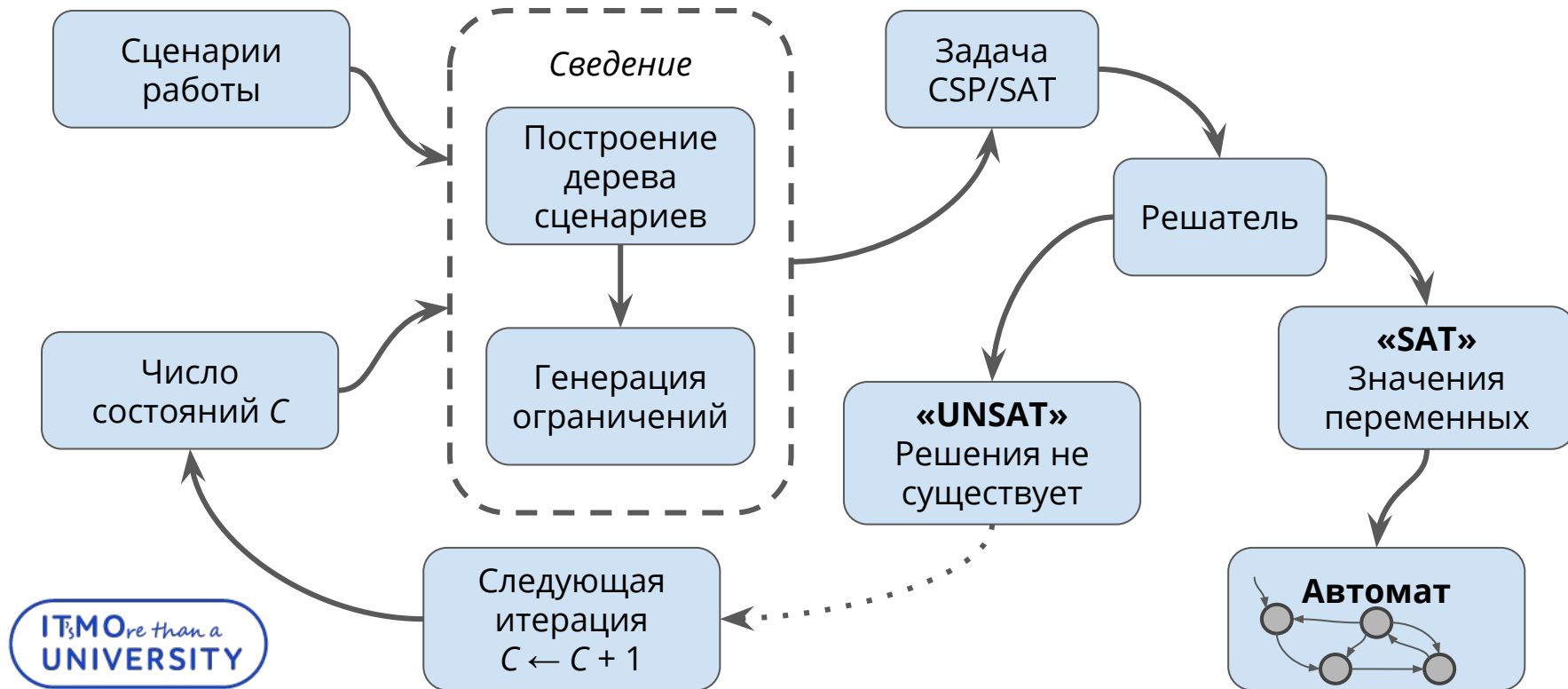
Задачи:

- 1) Расширение существующего метода fbCSP*
 - Разработка метода генерации *базовой* конечно-автоматной модели на основе сведения к задаче SAT
 - Разработка *точного* метода *минимизации* охранных условий
- 2) Разработка одноэтапного метода генерации минимальной конечно-автоматной модели на основе сведения к задаче SAT

Научная новизна

Научная новизна состоит в том, что впервые разработан **точный** метод генерации **минимальных** конечно-автоматных моделей логических контроллеров по примерам поведения.

Общий подход к синтезу минимальных моделей

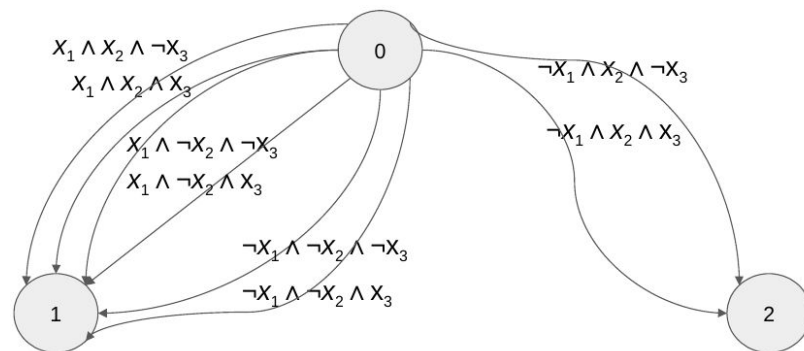


Предложенный двухэтапный метод: базовая модель

Algorithm 1: basic FB model inference

```

Data: scenarios set  $\mathcal{S}$ 
 $\mathcal{T} \leftarrow \text{buildScenarioTree}(\mathcal{S})$ 
foreach  $C \leftarrow 1$  to  $\infty$  do
     $K' \leftarrow C$ 
     $\mathcal{A}' \leftarrow \text{findModel}(\mathcal{T}, C, K')$ 
    if  $\mathcal{A}' \neq \text{null}$  then
        foreach  $K \leftarrow 1$  to  $C$  do
             $\mathcal{A} \leftarrow \text{findModel}(\mathcal{T}, C, K)$ 
            if  $\mathcal{A} \neq \text{null}$  then return  $\mathcal{A}$ 
    
```



Предложенный двухэтапный метод: минимизация

Algorithm 2: minimal guard conditions inference

Data: automata \mathcal{A} with C states and with at most K transitions from each state

foreach $c \leftarrow 1$ to C **do**

foreach $k \leftarrow 1$ to K **do**

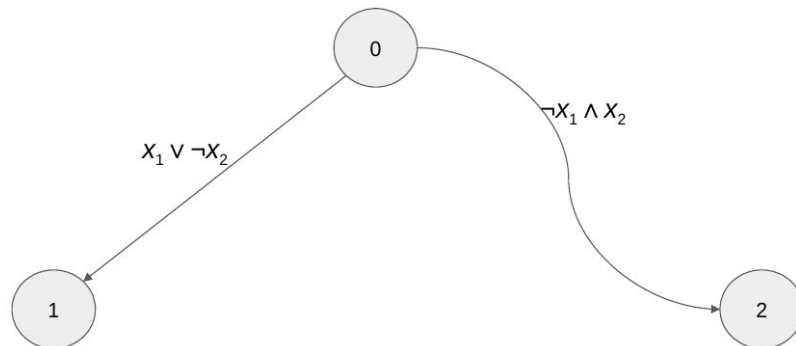
foreach $P \leftarrow 1$ to ∞ **do**

$\mathcal{G} \leftarrow \text{findGuard}(\mathcal{A}, c, k, P)$

if $\mathcal{G} \neq \text{null}$ **then**

 augment \mathcal{A} with \mathcal{G}

break



Предложенный одноэтапный метод

Algorithm 3: minimal FB model inference

Data: scenarios set \mathcal{S}

$\mathcal{T} \leftarrow \text{buildScenarioTree}(\mathcal{S})$

$C, K \leftarrow \text{estimate from SATGEN}(\mathcal{S})$

$P \leftarrow \text{estimate from TRANSITION-WISE}(\mathcal{S})$

foreach $N \leftarrow C \cdot K \cdot P$ **downto** 1 **do**

$\mathcal{A} \leftarrow \text{buildModel}(\mathcal{T}, C, K, P, N)$

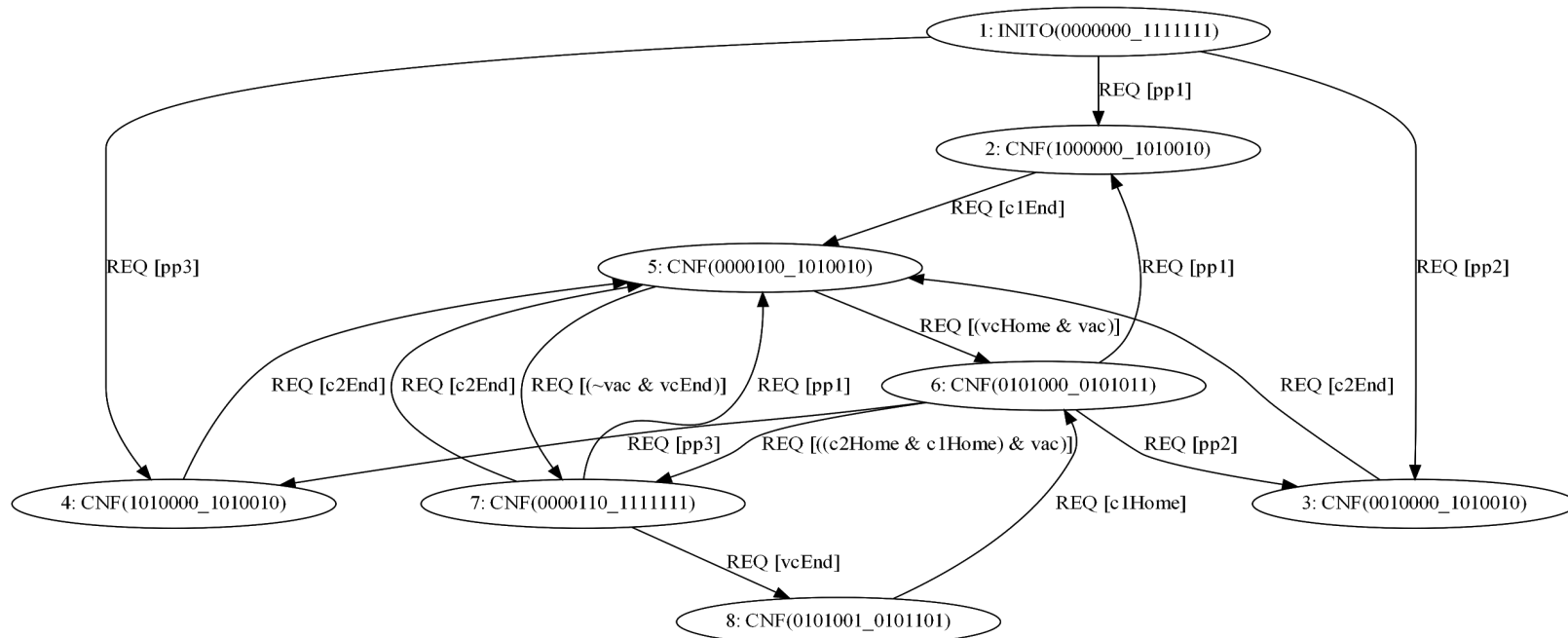
if $\mathcal{A} = \text{null}$ **then**

\perp **return** \mathcal{A}_{last}

else

\perp $\mathcal{A}_{last} \leftarrow \mathcal{A}$

Пример синтезированного автомата



Заключение

- ✓ Разработаны точные методы генерации минимальных конечно-автоматных моделей логических контроллеров по примерам поведения
- ✓ Предложенные методы реализованы в виде программного средства: github.com/ctlab/fbSAT
- ✓ Получен диплом «За лучший научно-исследовательский доклад» на VII Конгрессе молодых ученых в рамках IX сессии научной школы «Технологии программирования, искусственный интеллект, биоинформатика»
- ✓ Получена рецензия от начальника лаборатории НПО «Аврора» с оценкой «ОТЛИЧНО»

Спасибо за внимание!

Студент: Чухарев Константин Игоревич, P3435, lipen00@gmail.com

Руководитель: Ульянов Владимир Игоревич, к.т.н., доцент кафедры КТ

Консультант: Чивилихин Даниил Сергеевич, к.т.н., научный сотрудник кафедры КТ

IT's *MO*re than a
UNIVERSITY