

Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

# **Методы построения конечных автоматов на основе эволюционных алгоритмов**

Царев Федор Николаевич

Диссертация на соискание ученой степени  
кандидата технических наук

Специальность 05.13.11 – «Математическое и программное  
обеспечение вычислительных машин, комплексов и  
компьютерных сетей»

Научный руководитель – докт. техн. наук,  
профессор А. А. Шалыто

# Актуальность темы (1)

- Автоматное программирование – парадигма программирования, в рамках которой поведение программ предлагается описывать конечными автоматами
- Достоинства автоматного программирования:
  - Возможность генерировать до 70% исходного кода
  - Простота верификации на основе метода *Model checking*
- Поисковая инженерия программного обеспечения – область исследований, посвященная применению методов поисковой оптимизации для решения задач программной инженерии

# Актуальность темы (2)

- Часто эвристическое построение автоматов затруднено
- Построенные вручную автоматы могут быть не оптимальны, а их построение очень трудоемко
- Одно из решений – автоматизированная генерация конечных автоматов с помощью поисковой оптимизации, в частности, эволюционных алгоритмов
- Решение этой задачи позволит существенно повысить уровень автоматизации генерации автоматных программ

# Задачи диссертационной работы

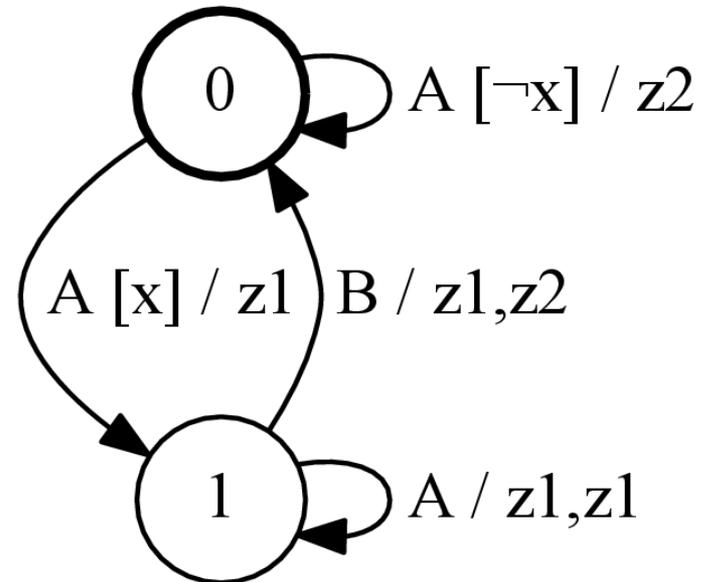
1. Разработать метод построения автоматов по обучающим примерам на основе эволюционных алгоритмов
2. Разработать метод выполнения операции скрещивания для генетических алгоритмов, учитывающий поведение автоматов на обучающих примерах
3. Разработать метод построения автоматов по обучающим примерам и темпоральным свойствам на основе эволюционных алгоритмов и верификации
4. Разработать технологию построения автоматов по обучающим примерам и темпоральным свойствам
5. Разработать инструментальное средство для автоматизации построения автоматов
6. Внедрить разработанные методы при построении автомата управления моделью беспилотного самолета и в учебный процесс

# Научная новизна

- Новый метод построения управляющих конечных автоматов по обучающим примерам на основе эволюционных алгоритмов. В эволюционные алгоритмы **добавлен новый шаг** «Расстановка выходных воздействий», который выполняется перед вычислением функции приспособленности
- Новый метод выполнения операции скрещивания для генетических алгоритмов, учитывающий поведение автоматов на обучающих примерах. Показано, что генетический алгоритм, использующий разработанный метод выполнения операции скрещивания, **осуществляет построение автоматов по обучающим примерам быстрее**, чем генетический алгоритм, использующий известный метод выполнения операции скрещивания, эволюционная стратегия и метод спуска на основе случайных мутаций
- Новый метод построения автоматов по обучающим примерам и темпоральным формулам на основе эволюционных алгоритмов и верификации. В нем для вычисления функции приспособленности **совместно применяются обучающие примеры и верификация**

# Основные понятия

- Управляющий конечный автомат:
  - входные события
  - входные переменные
  - выходные воздействия
- Обучающий пример (тест):
  - Входная последовательность пар  $\langle e, f \rangle$ 
    - $e$  – входное событие
    - $f$  – условие перехода – логическая формула от входных переменных
  - эталонная последовательность выходных воздействий
- Автомат на рисунке удовлетворяет тесту  $[(\langle A, \neg x \rangle, \langle A, x \rangle), (z_2, z_1)]$  и не удовлетворяет тесту  $[(\langle A, x \rangle), (z_2)]$

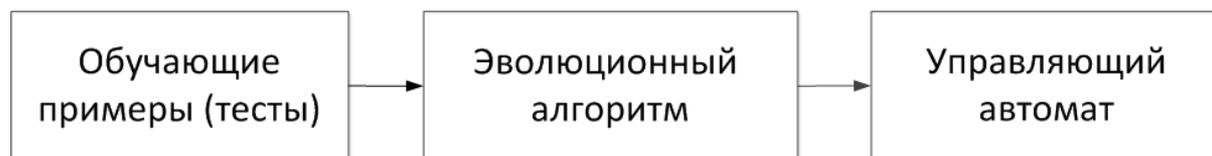


# Рассматриваемые эволюционные алгоритмы

- Метод спуска на основе случайных мутаций
- (1+1)-эволюционная стратегия
- Генетический алгоритм
- Компоненты:
  - Метод представления автоматов в виде особей эволюционного алгоритма
  - Функция приспособленности
  - Операция мутации
  - Операция скрещивания

# Метод построения автоматов по обучающим примерам (тестам) на основе эволюционных алгоритмов

- Входные данные:
  - множество событий
  - множество входных переменных
  - множество выходных воздействий
  - множество тестов
  - максимальное число состояний в искомом автомате



- Функция приспособленности основана на сравнении последовательности выходных воздействий, сгенерированной автоматом, и эталонной

$$FF_1 = \frac{\sum_{i=1}^n \left(1 - \frac{ED(\text{Output}[i], \text{Answer}[i])}{\max(|\text{Output}[i]|, |\text{Answer}[i]|)}\right)}{n}$$

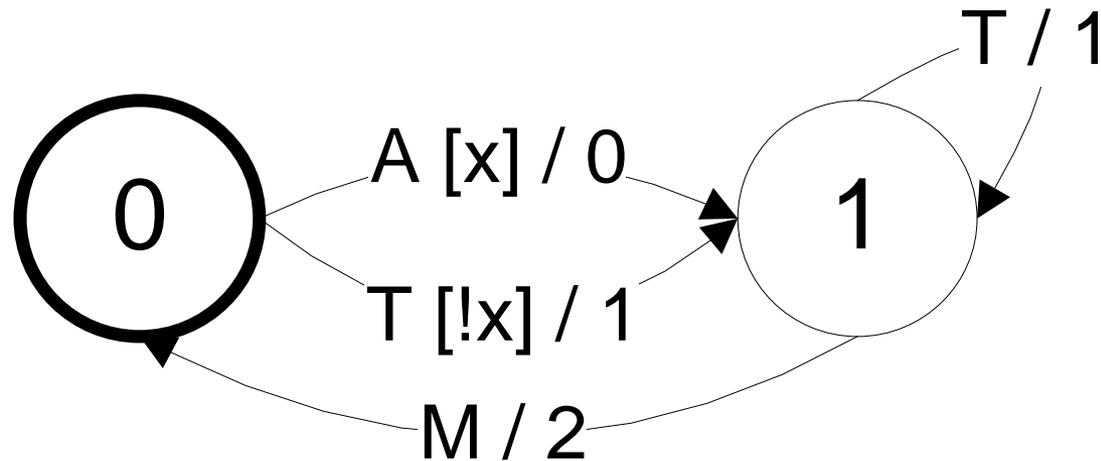
$$FF_2 = \begin{cases} 10 \cdot FF_1 + \frac{1}{M} \cdot (M - \text{cnt}), & FF_1 < 1 \\ 20 + \frac{1}{M} \cdot (M - \text{cnt}), & FF_1 = 1 \end{cases}$$

ED – редакционное расстояние

# Предварительная обработка ВХОДНЫХ ДАННЫХ

- Для каждой пары условий, встречающихся в тестах, вычисляется:
  - Являются ли они эквивалентными
  - Имеют ли общую выполняющую подстановку
- Используется в алгоритме удаления дублированных и противоречивых переходов
- Вычислительная сложность:
  - $O(2^{2r'} \cdot l^2)$ , где  $r'$  – максимальное число входных переменных, от которых зависит некоторое условие перехода, а  $l$  – суммарная длина входных последовательностей тестов

# Представление особи в эволюционном алгоритме

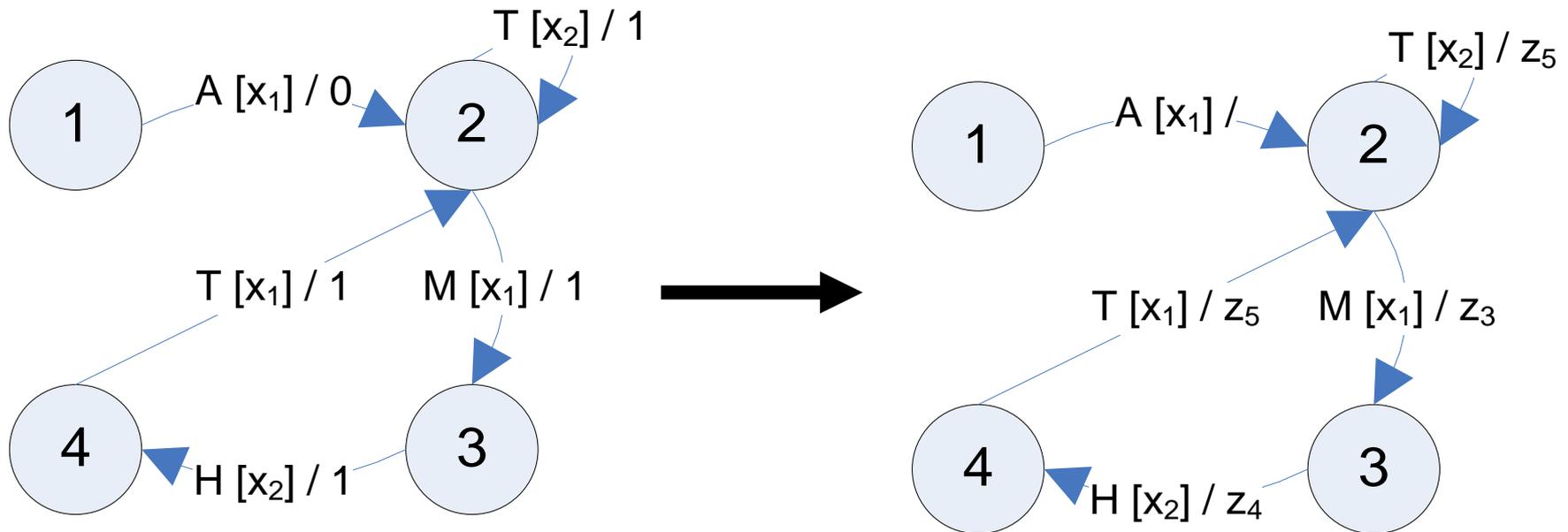


**{2, {{A, x, 1, 0}, {T, !x, 1, 1}}, {{T, true, 1, 1}, {M, true, 0, 2}}}**

- В особи задается только «каркас» конечного автомата
- Все конечные автоматы, рассматриваемые в рамках одного запуска эволюционного алгоритма, имеют одинаковое число состояний

# Алгоритм расстановки выходных воздействий

- В эволюционные алгоритмы предлагается добавить **новый шаг**, который выполняется перед вычислением функции приспособленности



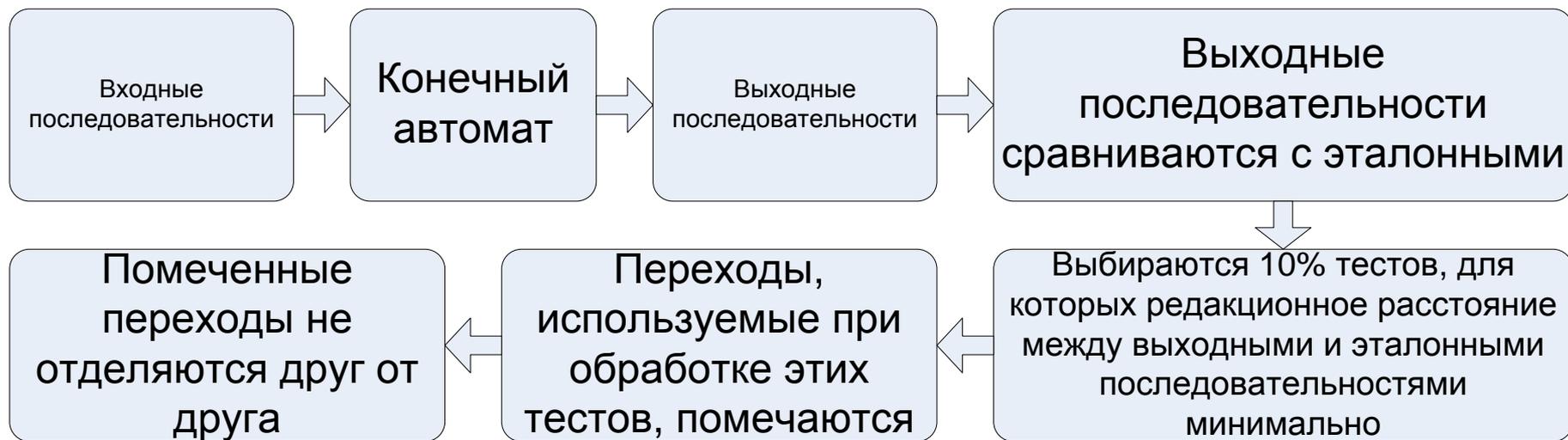
# Операция мутации

- В генетическом алгоритме и методе спуска на основе случайных мутаций
  - Изменение перехода
    - Состояние, в которое он ведет
    - Событие
    - Условие перехода
    - Число выходных воздействий
  - Добавление или удаление перехода
- В эволюционной стратегии:
  - $(x + 1)$  раз применяется мутация, используемая в методе спуска и эволюционной стратегии, где  $x$  выбрано из геометрического распределения с вероятностью успеха, равной 0.5

# Операция скрещивания

- Два варианта – обычное и с учетом тестов
- **Обычное** – для каждого номера состояния проводятся следующие операции:
  - Переходы автоматов-родителей объединяются в общий список
  - Элементы списка переставляются случайным образом
  - Элементы списка распределяются по автоматам-потомкам
- **С учетом поведения конечных автоматов на тестах** – переходы, которые используются при обработке нескольких тестов, которые автоматы проходят лучше всего, не затрагиваются

# Метод выполнения операции скрещивания с учетом поведения конечных автоматов на тестах

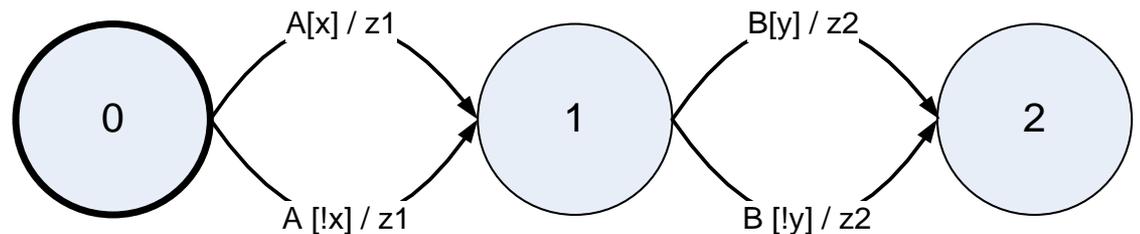


# Пример работы предлагаемой операции скрещивания (1)

- Набор тестов:

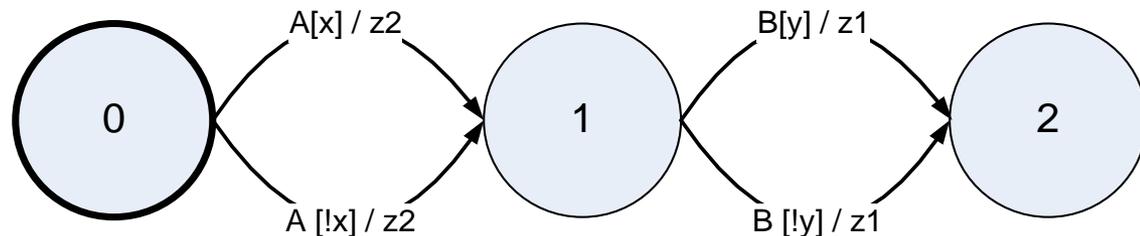
- Тест 1:

- $A [x], B [y]$
    - $z1, z2$



- Тест 2:

- $A [!x], B [!y]$
    - $z2, z1$



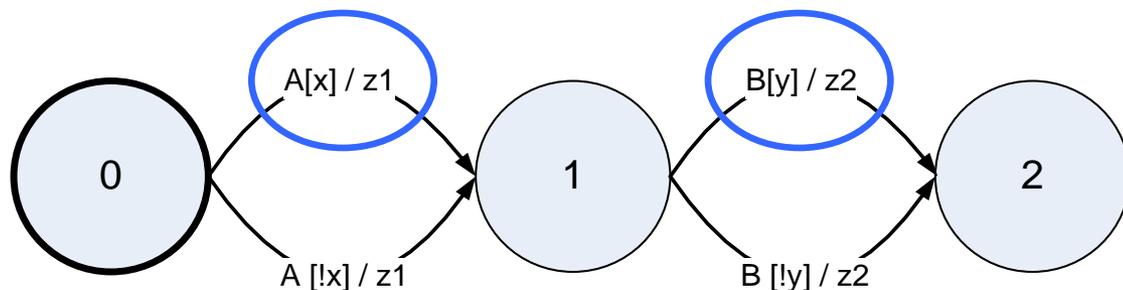
- ...

# Пример работы предлагаемой операции скрещивания (2)

- Набор тестов:

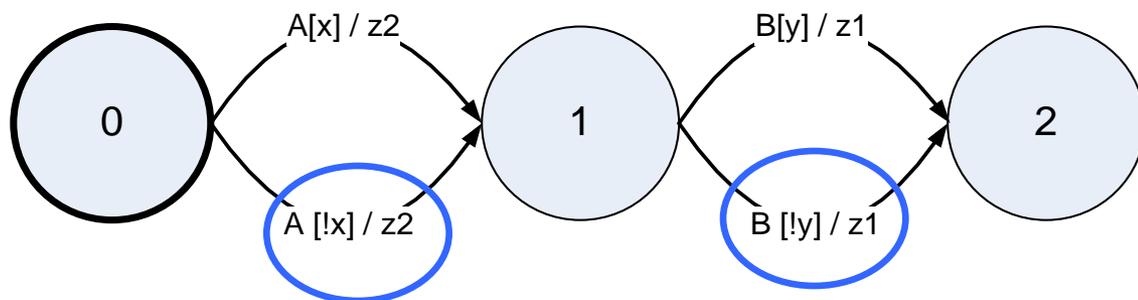
- Тест 1:

- $A [x], B [y]$
    - $z1, z2$



- Тест 2:

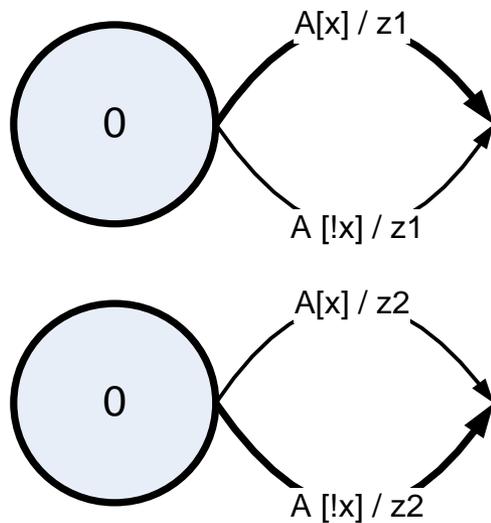
- $A [!x], B [!y]$
    - $z2, z1$



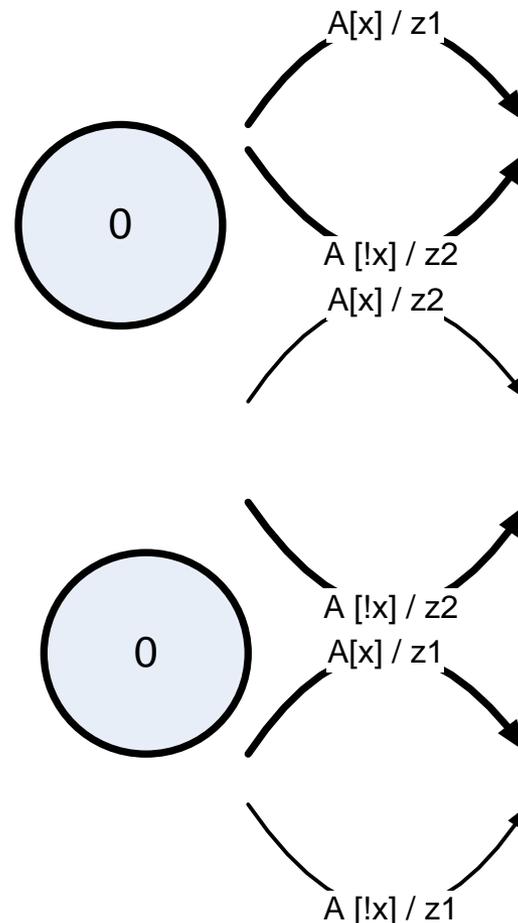
- ...

# Пример работы предлагаемой операции скрещивания (3)

«Родительские»  
особи

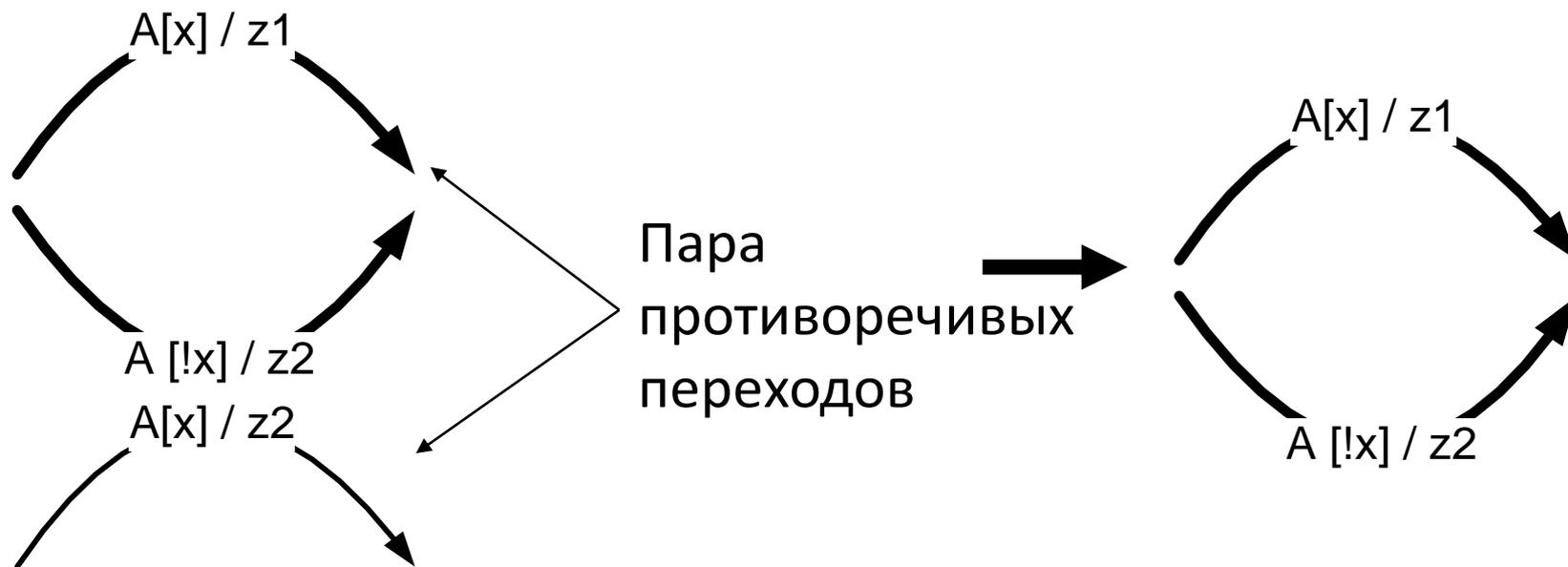


«Особь»-потомки



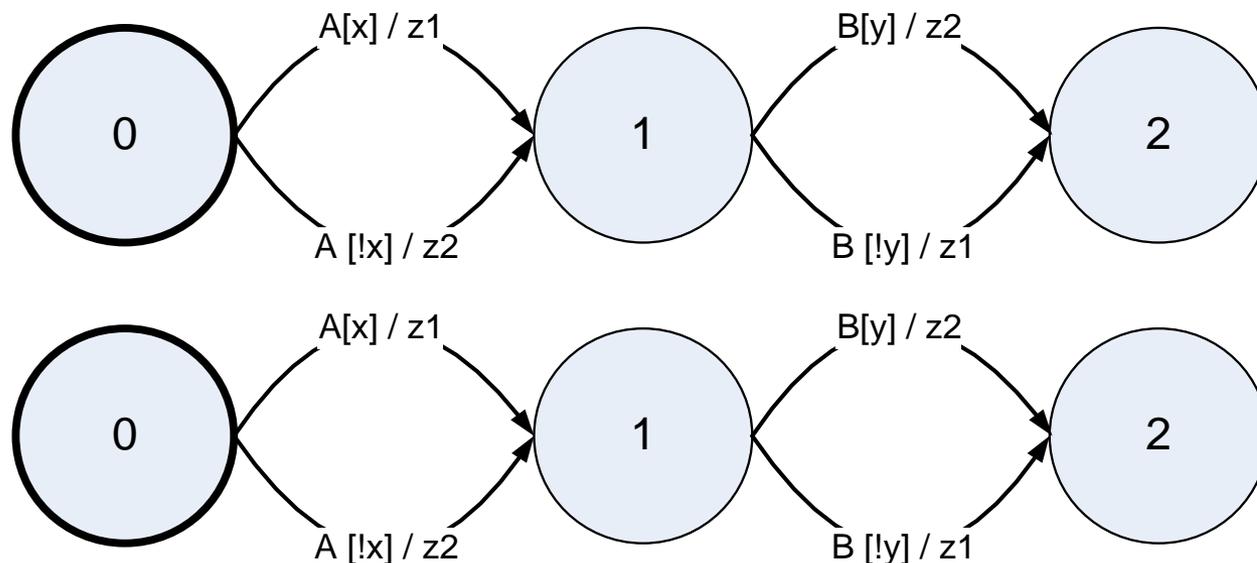
# Пример работы предлагаемой операции скрещивания (4)

- Алгоритм удаления дублированных и противоречивых переходов
- Показано для состояния «0» первой особи-«потомка»



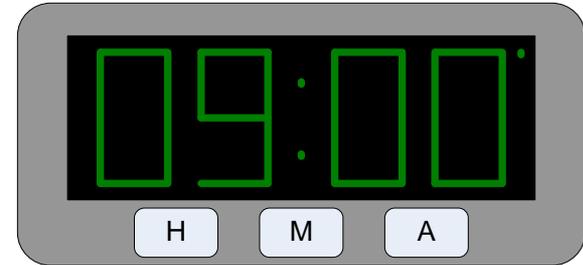
# Пример работы предлагаемой операции скрещивания (5)

- Оба автомата, задаваемые особями-«потомками», проходят оба теста



# Вычислительные эксперименты (1)

- Построение автомата управления часами с будильником
  - Четыре события
  - Две входные переменные
  - Семь выходных воздействий
  - 38 обучающих примеров
    - Суммарная длина входных последовательностей: 242
    - Суммарная длина выходных последовательностей: 195
- Тесты, сгенерированные случайным образом:
  - Был сгенерирован набор автоматов с 4, ..., 10 состояниями
  - По каждому из них был построен набор тестов - суммарная длина входных последовательностей 150k для автомата из  $k$  состояний



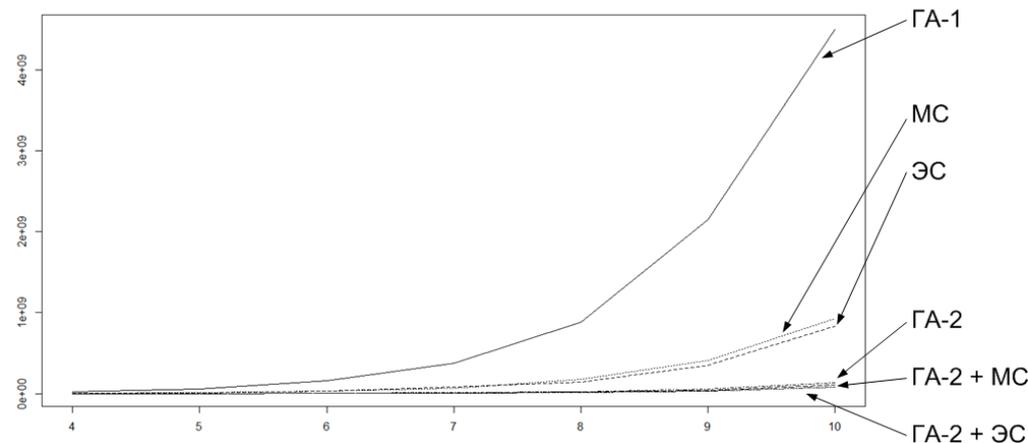
## Вычислительные эксперименты (2)

Название	Тип	Известный метод скрещивания	Предложенный метод скрещивания
МС	МС	-	-
ЭС	ЭС	-	-
ГА-1	ГА	+	-
ГА-2	ГА	-	+
ГА-2 + МС	ГА + МС	-	+
ГА-2 + ЭС	ГА + ЭС	-	+

# Вычислительные эксперименты (3)

- Число запусков каждого алгоритма:
  - Часы с будильником – 1000
  - Построение по тестам, сгенерированным случайным образом, – 100
- Вывод (подтверждается результатами статистического теста) – *применение метода скрещивания с учетом поведения автоматов на обучающих примерах позволяет существенно повысить скорость работы генетического алгоритма*

Алгоритм	Минимум	Максимум	Среднее	Медиана
ГА-1	1093938	41794531	6783215	5014202
МС	1387	9710090	1275439	792481
ЭС	1325	9915947	1317674	901615
ГА-2	51977	1196233	205451	142013
ГА-2 + МС	46311	780469	127712	103904
ГА-2 + ЭС	38458	2714324	129330	84778



# Метод построения конечных автоматов по обучающим примерам и темпоральным формулам на основе эволюционных алгоритмов и верификации

- Входные данные:
  - множество событий
  - множество входных переменных
  - множество выходных воздействий
  - множество тестов
  - множество темпоральных формул
  - максимальное число состояний в искомом автомате
- Функция приспособленности учитывает результаты не только тестирования, но и верификации

$$FF = FF_2 \cdot \left(1 + \frac{nf_1}{nf_2}\right)$$

- $nf_2$  – общее число темпоральных формул в спецификации
- $nf_1$  – число формул, которые выполняются для рассматриваемого автомата.
- в случае  $nf_1 = nf_2 = 0$ , величина  $\frac{nf_1}{nf_2}$  также равна нулю. Таким образом, если число темпоральных формул равно нулю, то функция приспособленности совпадает с определенной ранее

# Модельная задача – построение автомат управления дверьми лифта

- Пять событий, три выходных воздействия
- Девять тестов, 11 темпоральных LTL-формул
- 1000 экспериментов

	Лифт	
	Тесты	Тесты и формулы
Среднее	$7.479 \times 10^4$	$7.246 \times 10^5$
Минимальное	$2.184 \times 10^4$	$7.054 \times 10^4$
Максимальное	$2.999 \times 10^5$	$5.492 \times 10^6$
Число автоматов, соответствующих спецификации	7	1000

# Технология автоматизированного построения управляющих конечных автоматов (1)



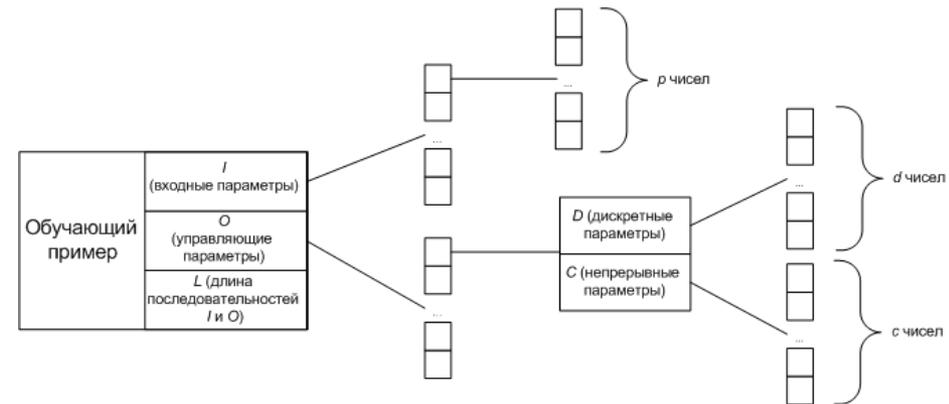
- В случае изменения спецификации необходимо выполнить построение автомата заново
- В ряде случаев требуется экспертная оценка построенного автомата

# Инструментальное средство для поддержки технологии

- Исходный код размещен в открытом доступе в сети Интернет по адресу:  
<http://code.google.com/p/gabp/>
- Реализует разработанные методы и технологию построения конечных автоматов
- Входные данные – описание тестов и темпоральных формул в XML-формате
- Выходные данные – описание конечного автомата в XML-формате инструментального средства *UniMod*

# Построение автомата управления моделью беспилотного самолета (1)

- Для моделирования самолета используется симулятор *FlightGear*
- Записаны обучающие примеры – как самолетом управляет человек
- Учитываются не только дискретные, но и непрерывные выходные воздействия



# Построение автомата управления моделью беспилотного самолета (2)

- Функция приспособленности обобщена на случай непрерывных выходных воздействий

$$Fitness = 1 - \sqrt{\frac{1}{N} \sum_{i=1}^n \frac{\rho^2(O_i, \tilde{O}_i)}{\rho^2(O_i, 0)}}$$

$$\rho(O_i, \tilde{O}_i) = \sqrt{\sum_{t=1}^{L_i} \left( \sum_{k=1}^d [D_{i,t,k} \neq \tilde{D}_{i,t,k}] + \sum_{k=1}^c (C_{i,t,k} - \tilde{C}_{i,t,k})^2 \right)}$$

- Расстановка выходных воздействий осуществляется с помощью решения набора систем линейных уравнений

$$S = \sum_{i=1}^n A_i \sum_{t=1}^{L_i} \left( C_{i,t,m} - \sum_{j=1}^{\text{cnt}} \alpha_{i,j}[t] u_j \right)^2$$

# Построение автомата управления моделью беспилотного самолета (3)

- Было записано 10 обучающих примеров выполнения самолетом «мертвой петли»
- Было проведено 50 запусков генетического алгоритма, построенные 50 автоматов были просмотрены экспертом, из них был выбран лучший
- По итогам этой части работы принята статья в журнал «Известия РАН. Теория и системы управления»

# Внедрение в учебный процесс

- Разработаны две виртуальные лаборатории для обучения эволюционным алгоритмам построения конечных автоматов
- За время использования виртуальных лабораторий в учебном процессе лабораторные работы были выполнены более чем 150 студентами кафедры «Компьютерные технологии» НИУ ИТМО. Часть отчетов по лабораторным работам опубликована на сайте <http://is.ifmo.ru/labs/>

# Результаты работы (1)

- Метод построения автоматов по обучающим примерам на основе эволюционных алгоритмов
- Метод выполнения операции скрещивания для генетических алгоритмов, учитывающий поведение автомата на обучающих примерах
- Метод построения автоматов по обучающим примерам и темпоральным свойствам на основе эволюционных алгоритмов и верификации
- Разработана технология построения автоматов по обучающим примерам и темпоральным формулам. Разработано инструментальное средство для автоматизации построения автоматов

# Результаты работы (2)

- Разработанные методы применены при построении автомата управления моделью беспилотного самолета и внедрены в учебный процесс на кафедре «Компьютерные технологии» НИУ ИТМО
- Опубликовано шесть статей, из которых пять в журналах из перечня ВАК
- Получены три свидетельства о регистрации программ для ЭВМ
- Сделано 17 докладов на конференциях, в том числе, три доклада на международной конференции GECCO и **пленарный доклад** на третьей российской конференции с международным участием «Технические и программные средства систем управления, контроля и измерения» (Институт проблем управления имени В. А. Трапезникова РАН, 2012)

**Спасибо за внимание!**

Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

# **Методы построения конечных автоматов на основе эволюционных алгоритмов**

Царев Федор Николаевич

Диссертация на соискание ученой степени  
кандидата технических наук

Специальность 05.13.11 – «Математическое и программное  
обеспечение вычислительных машин, комплексов и  
компьютерных сетей»

Научный руководитель – докт. техн. наук,  
профессор А. А. Шалыто