

Генерация тестов для олимпиадных задач по теории графов с использованием эволюционных алгоритмов

Буздалов Максим Викторович
Научный руководитель: докт. техн. наук,
проф. Шалыто Анатолий Абрамович

15 июня 2011 г., СПбГУ ИТМО

Содержание

Цель работы

Общий метод

Пример: задача „Work for Robots“

Пример: Задача о максимальном потоке

Актуальность

Автоматизация генерации тестов для олимпиадных задач

- ▶ Упростит труд членов жюри
- ▶ Повысит качество тестов и соревнования в целом
- ▶ В данной работе рассматриваются:
 - ▶ задачи по теории графов
 - ▶ тесты против неэффективных решений (время, память)

Новизна

- ▶ Эволюционные алгоритмы для генерации юнит-тестов?
 - ▶ Цель — покрытие кода
 - ▶ Производительность не учитывается
- ▶ Microsoft PeX?
 - ▶ Использует перебор путей выполнения
 - ▶ Производительность не учитывается
 - ▶ „Убивается“ типичными конструкциями из решений
- ▶ Бакалаврская?
 - ▶ Новый класс задач
 - ▶ Новые виды функций приспособленности для новых целей
 - ▶ Новые эволюционные алгоритмы

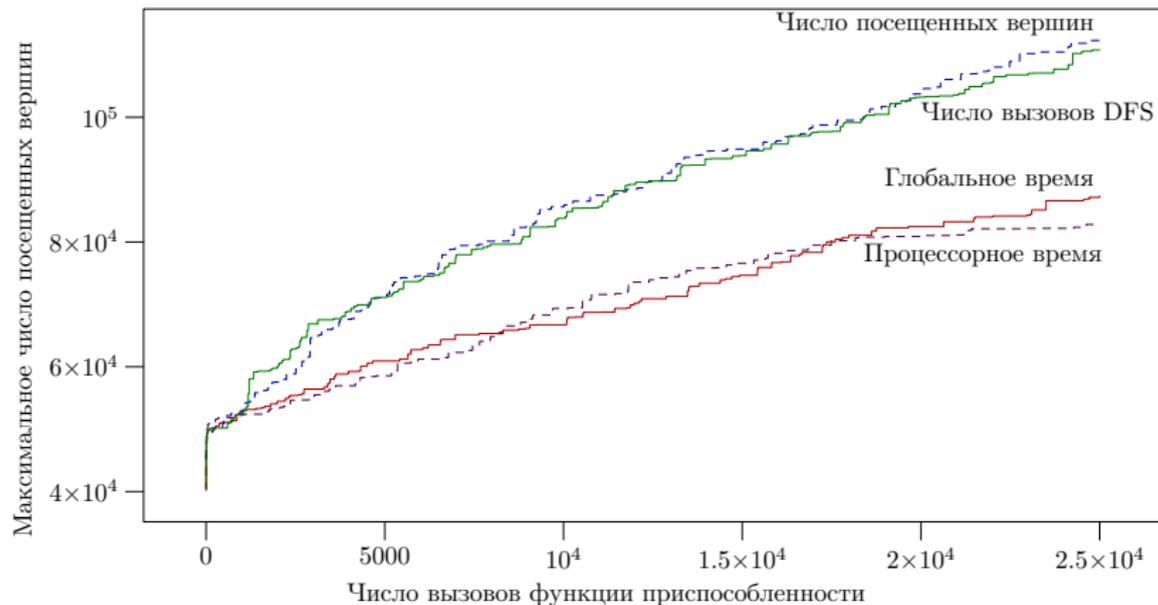
Генерация тестов как задача оптимизации

- ▶ Качество теста выражается количественно:
 - ▶ время работы решения на тесте
 - ▶ число вызовов функции
 - ▶ заполненность структур данных
 - ▶ ...
- ▶ Применяется эволюционный алгоритм:
 - ▶ Тест — особь алгоритма
 - ▶ „Качество“ теста — функция приспособленности
 - ▶ Условие останова — время выполнения на тесте больше ограничения

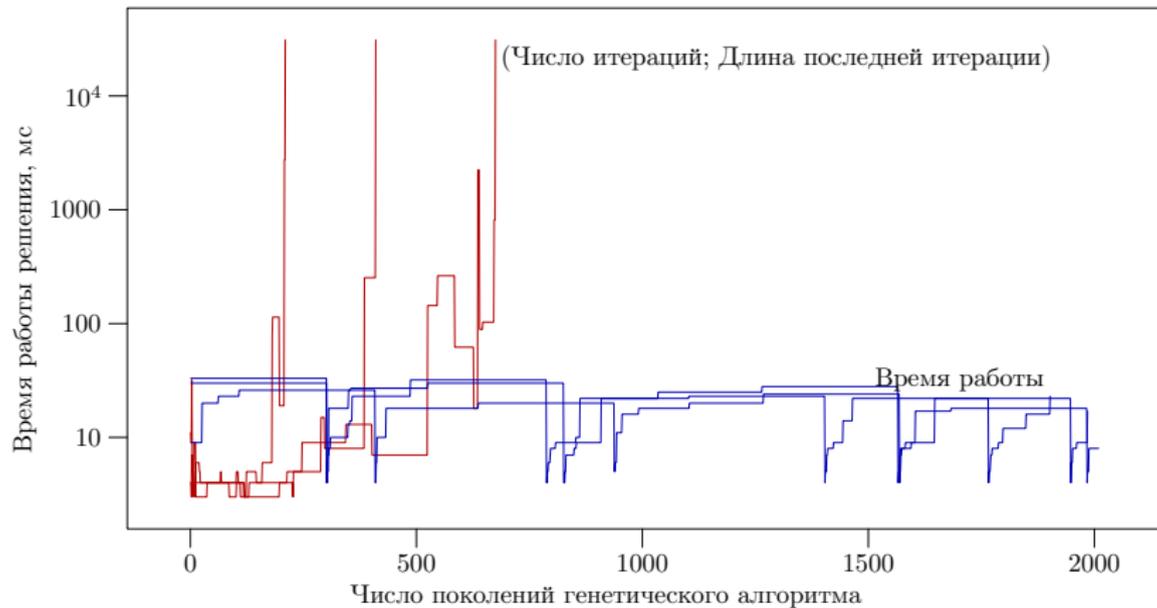
Функция приспособленности

- ▶ Исходный код решения модифицируется:
 - ▶ Вычисление времени выполнения
 - ▶ Число вызовов функций
 - ▶ Число итераций алгоритма
 - ▶ Специфические для решения величины
- ▶ Функция приспособленности — комбинация величин
 - ▶ Линейная комбинация
 - ▶ Вектор величин, лексикографическое сравнение
 - ▶ Может подбираться индивидуально

Функция приспособленности (1)



Функция приспособленности (2)



Кодирование теста

- ▶ Задание графа эффективными способами
 - ▶ матрица смежности
 - ▶ список ребер
 - ▶ может зависеть от задачи
- ▶ Индивидуальный подход к каждой задаче
 - ▶ увеличение доли „хороших“ тестов за счет выбора способа кодирования
 - ▶ эффективный учет ограничений задачи

Задача „Work for Robots“

- ▶ <http://acm.timus.ru>, №1695
- ▶ Дано:
 - ▶ Неориентированный граф
 - ▶ Число вершин N , $1 \leq N \leq 50$
- ▶ Требуется:
 - ▶ Найти число клик в графе
 - ▶ Клика — полный подграф
 - ▶ Ограничение по времени — 2 с
 - ▶ Ограничение по памяти — 64 Мб
- ▶ Задача NP-трудна, есть решение за $O(2^{N/2})$
- ▶ Множество эвристических переборных решений

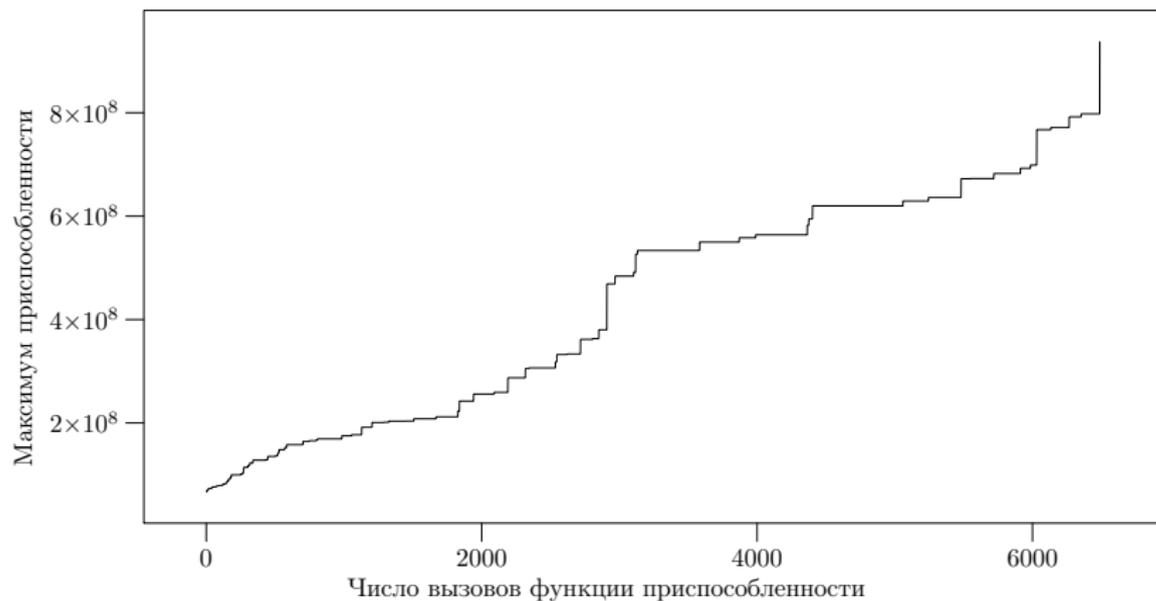
Генерация тестов

- ▶ Используется $(1 + 1)$ -эволюционная стратегия.
 - ▶ В поколении один тест
 - ▶ Из него мутацией создается новый тест
 - ▶ В следующее поколение проходит лучший из них
- ▶ Граф задается матрицей смежности
- ▶ Функция приспособленности варьируется в зависимости от типа решения

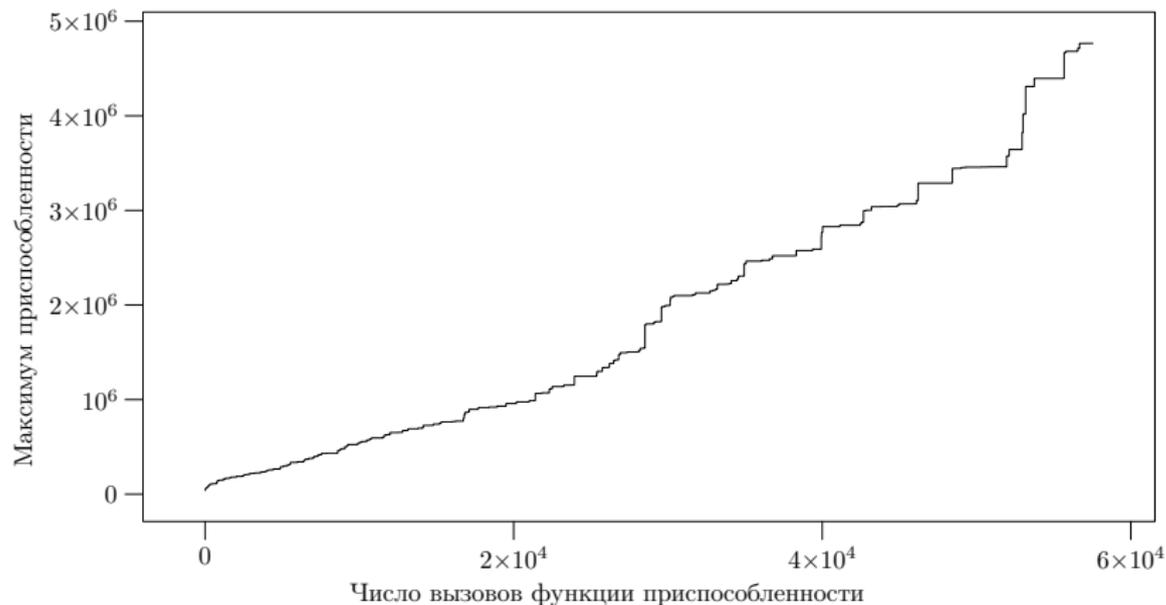
Типы решений

- ▶ Перебор с возвратом:
 - ▶ Кеширование младших бит
 - ▶ Использование ассоциативного массива
 - ▶ Использование «большого массива»
- ▶ Максимальное независимое подмножество и ДП
- ▶ Предварительное переупорядочивание вершин

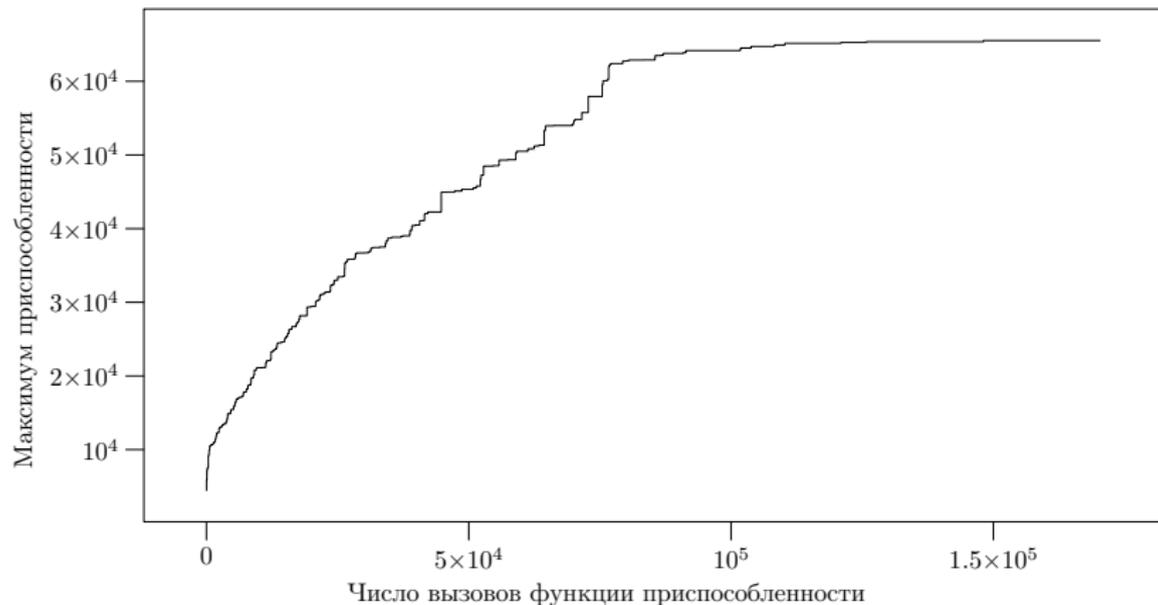
Пример: кэширование младших бит



Пример: Ассоциативный массив



Пример: «Большой» массив



Результаты

- ▶ Было сгенерировано 10 новых тестов
- ▶ Из 86 ранее зачтенных решений 45 не прошли новые тесты

Задача о максимальном потоке

- ▶ Дан граф с V вершинами и E ребрами
- ▶ Выделены вершины: s — исток, t — сток
- ▶ Пропускная способность i -того ребра c_i
- ▶ Найти максимальный поток $F = \{f_i\}$
 - ▶ $0 \leq f_i \leq c_i$
 - ▶ входящий поток во все вершины, кроме s и t , равен исходящему
 - ▶ исходящий поток из s максимален

Ограничения

- ▶ $2 \leq V \leq 100$
- ▶ $0 \leq E \leq 5000$
- ▶ $0 \leq c_i \leq 10000$
- ▶ Допускаются петли и кратные ребра
- ▶ Граф неориентированный

Решения

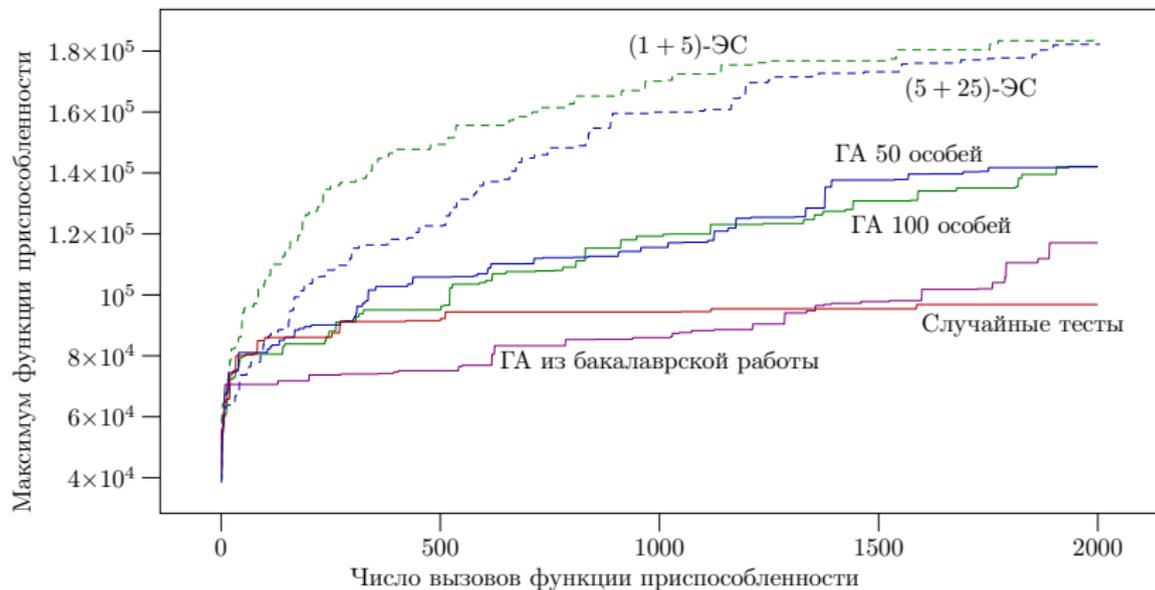
Тестируемые решения:

- ▶ Алгоритм Форда-Фалкерсона
- ▶ Алгоритм Форда-Фалкерсона с масштабированием

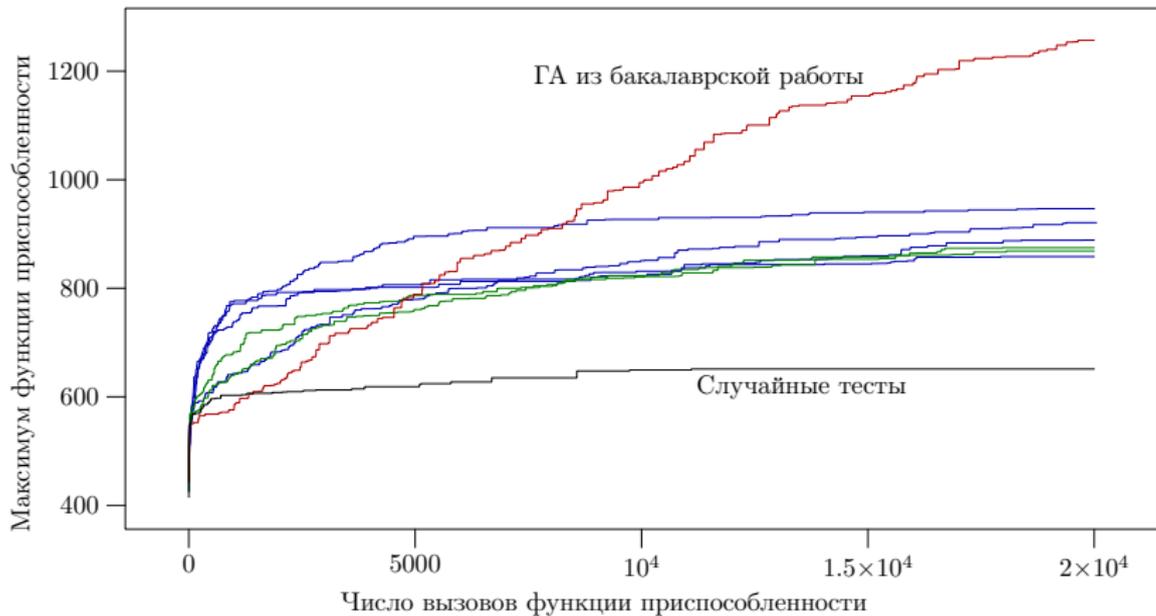
Эволюционные алгоритмы:

- ▶ Эволюционные стратегии из библиотеки Watchmaker
- ▶ Генетические алгоритмы из библиотеки Watchmaker
- ▶ Генетический алгоритм из бакалаврской работы

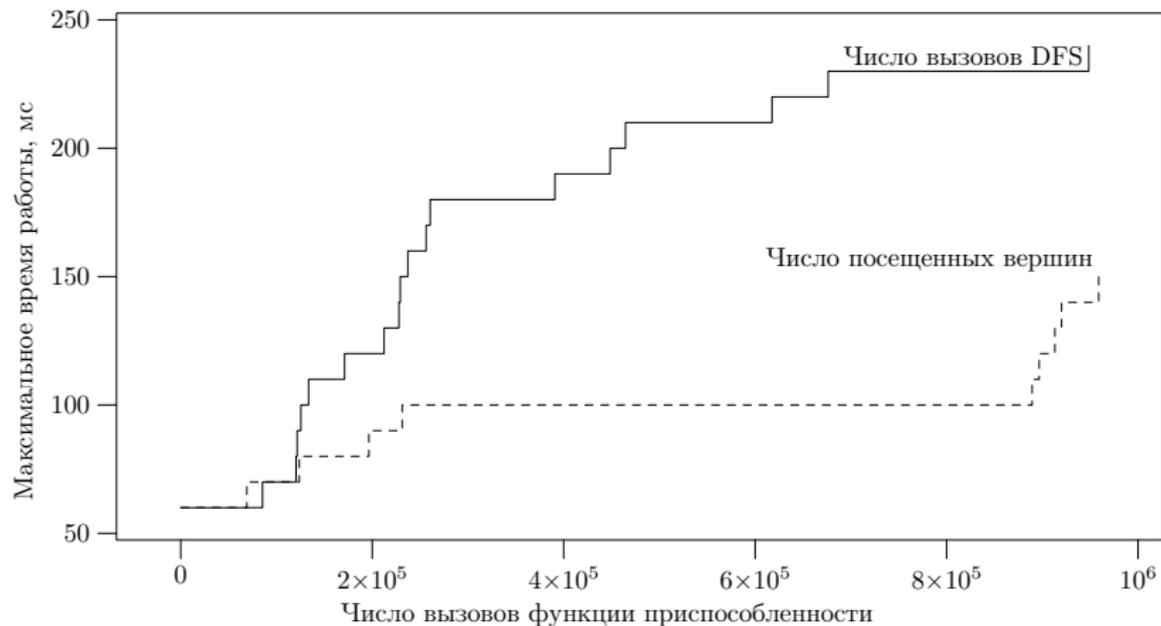
Эксперимент: Форд-Фалкерсон



Эксперимент: Форд-Фалкерсон с масштабированием



Продолжение эксперимента



Выводы

- ▶ В простых случаях следует использовать эволюционные стратегии
- ▶ В более сложных — специально разработанные генетические алгоритмы

Заключение

- ▶ Изложен метод генерации тестов для олимпиадных задач по теории графов с использованием эволюционных алгоритмов
- ▶ Приведены примеры использования метода
 - ▶ Задача о числе клик в графе
 - ▶ Задача о поиске максимального потока

Спасибо за внимание!

Вопросы?

