РОССИЙСКИЙ ФЕДЕРАЛЬНЫЙ ЯДЕРНЫЙ ЦЕНТР
всероссийский научно-исследовательский
институт экспериментальной физики

XIII ЗАБАБАХИНСКИЕ НАУЧНЫЕ ЧТЕНИЯ

АЛГОРИТМ НАХОЖДЕНИЯ ПЕРЕСЕЧЕНИЯ ВЫПУКЛЫХ МНОГОУГОЛЬНИКОВ С ВЫСОКОЙ ТОЧНОСТЬЮ ДЛЯ ИСПОЛЬЗОВАНИЯ В ЗАДАЧАХ ГАЗОДИНАМИКИ

Власов К.О.
ФГУП «РФЯЦ-ВНИИЭФ» ИТМФ

2017

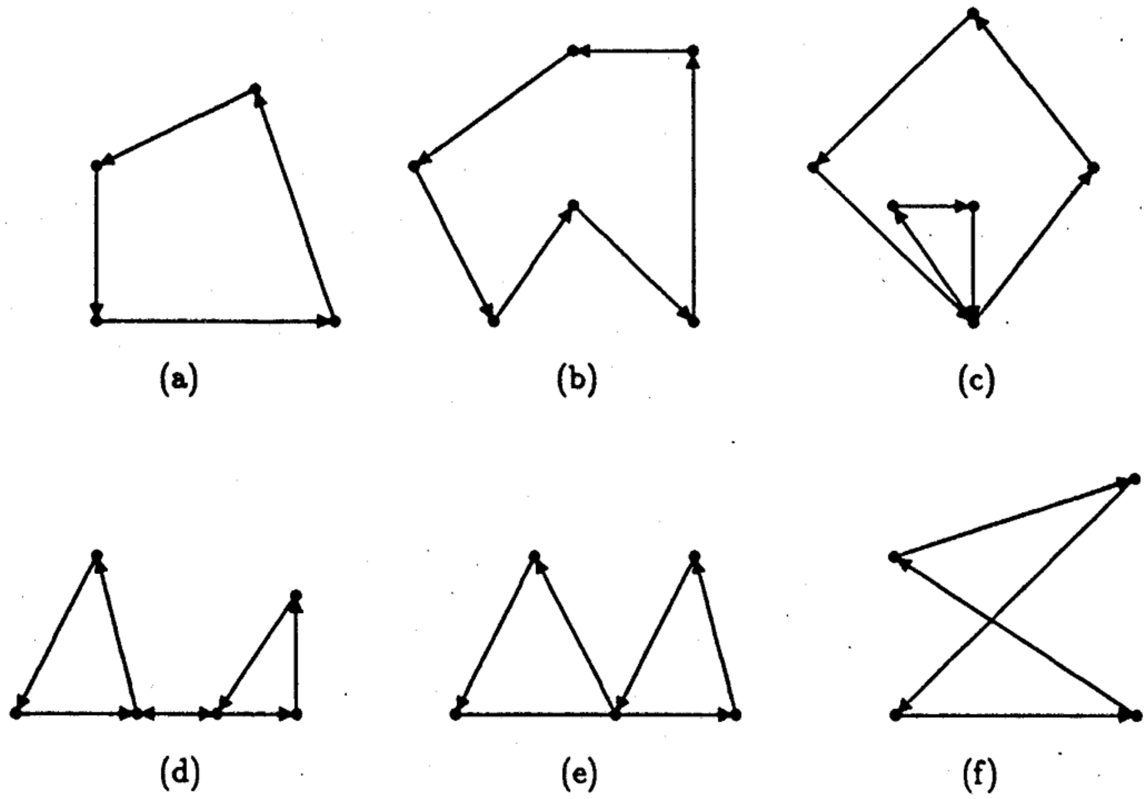
План доклада:

- Введение. Актуальность работы
- Обзор существующих алгоритмов
- Описание алгоритма поиска пересечений
- Обзор существующих библиотек
- Сравнение библиотек по производительности
- Сравнительное тестирование алгоритма
- Заключение

Введение. Актуальность работы

На сегодняшний день решение такой вычислительной задачи, как поиск пересечения плоских многоугольников, весьма востребовано. Это обусловлено тем, что в приложениях, использующих компьютерную графику, в автоматизированных системах различного типа (CAD, CAE, CAM), в геоинформационных системах (GIS), в векторных графических редакторах и многих других объекты описываются с помощью плоских многоугольников.

Классы многоугольников



a) простой выпуклый; b) простой; c) регулярный вершинно-полный;
d) вершинно-полный; e) ориентированный; f) общий (обыкновенный)

Алгоритм Сазерленда-Ходжмана

История создания:

Алгоритм был опубликован в **1974** году Сазерлендом и Ходжманом и является одним из первых алгоритмов отсечения многоугольников.

Область применения:

Алгоритм находит пересечения общего многоугольника и произвольного выпуклого многоугольника.

Особенности:

Плюсы:

простота реализации и малая требовательность к памяти

Минусы:

высокая временная сложность: $O(n_1 n_2 \log n_2)$

невозможность обобщения для поиска объединения и разности многоугольников

Алгоритм О'Рурка

История создания:

Алгоритм был опубликован в **1982** году. За основу взят алгоритм, который предложили Шеймос и Хоуи в 1976 году.

Область применения:

Алгоритм находит пересечения только выпуклых многоугольников.

Особенности:

Плюсы:

простота реализации; низкая временная сложность: $O(N)$

Минусы:

неспособность обрабатывать вырожденные пересечения ребер (исходный алгоритм)

Алгоритм Ватти

История создания:

Алгоритм был опубликован в **1992** году. Является модификацией алгоритма Вейлера–Азертона, который работал только с регулярными многоугольниками.

Область применения:

Наиболее универсальный алгоритм поиска пересечений многоугольников. Способен находить пересечения многоугольников любого типа.

Особенности:

Плюсы:

Способен находить объединения и разности многоугольников

Минусы:

высокая временная сложность

не является устойчивым (не учтены некоторые крайние случаи)

Требования к алгоритму

- Высокая чувствительность
- Высокая скорость работы в скалярном режиме
- Определения точек пересечения с высокой точностью
- Устойчивость

Описание алгоритма поиска пересечений

Этапы работы алгоритма:

-Определение общих вершин и внутренних вершин для обоих многоугольников

-Определение точек пересечения ребер многоугольников

-Сортировка найденных точек для построения результирующего многоугольника

Область применения:

Алгоритм находит пересечения произвольных выпуклых многоугольников и звездных многоугольников (с ограничениями).

Задачи газодинамики; численные схемы, требующих перестроения расчетной сетки в процессе решения - схемы семейства ALE, метод VOF (Volume Of Fluid)

Особенности:

Плюсы:
простота реализации; высокая чувствительность; устойчивость
Минусы:
большие затраты памяти

Описание алгоритма поиска пересечений

Барицентрические координаты

В барицентрических координатах уравнение первого ребра имеет вид:

$$(x_1, y_1)^T = \lambda_1 (x_1^0, y_1^0)^T + (1-\lambda_1)(x_1^1, y_1^1)^T$$

Аналогично записывается уравнение для второго ребра:

$$(x_2, y_2)^T = \lambda_2 (x_2^0, y_2^0)^T + (1-\lambda_2)(x_2^1, y_2^1)^T$$

Решаем уравнение:

$$(x_1, y_1)^T = (x_2, y_2)^T$$

и находим значения (λ_1, λ_2)

Если $(\lambda_1, \lambda_2) \in [0,1]^2$, то рассматриваемые ребра имеют точку пересечения.

Тогда координаты точки пересечения являются решением уравнения:

$$(x_i, y_i)^T = \lambda_2 (x_2^0, y_2^0)^T + (1-\lambda_2)(x_2^1, y_2^1)^T$$

Обзор существующих библиотек

General Polygon Clipping Library (GPC)

Разработчик и условие распространения:

Alan Murta 1998 Манчестерский университет. Библиотека предоставляется бесплатно (некоммерческое использование).

Использованный алгоритм и область применения:

В основе лежит **алгоритм Ватти**.

Многоугольники, имеющие дыры и/или самопересечения,

сложные многоугольники, состоящие из нескольких раздельных контуров.

Ссылка:

http://www.cs.man.ac.uk/ ~toby/alan/software/gpc.html

JTS Topology Suite (JTS)

Разработчик и условие распространения:

Компания **Vivid Solutions**. Предоставляется бесплатно, исходный код недоступен.

Родственная библиотека: Geos.

Использованный алгоритм и область применения:

Многие подходы, использованные в JTS, являются уникальными. Родственные алгоритмы: Ватти, Шутте, Леонова и Чана.

Многоугольники, имеющие дыры и/или самопересечения, сложные многоугольники, состоящие из нескольких раздельных контуров.

Ссылка:

http://www.vividsolutions.com/ jts/JTSHome.htm

Clipper

Разработчик и условие распространения:

Энгес Джонсон. Полностью бесплатная библиотека с открытым исходным кодом. Родственная библиотека: Boost.Geometry

Использованный алгоритм и область применения:

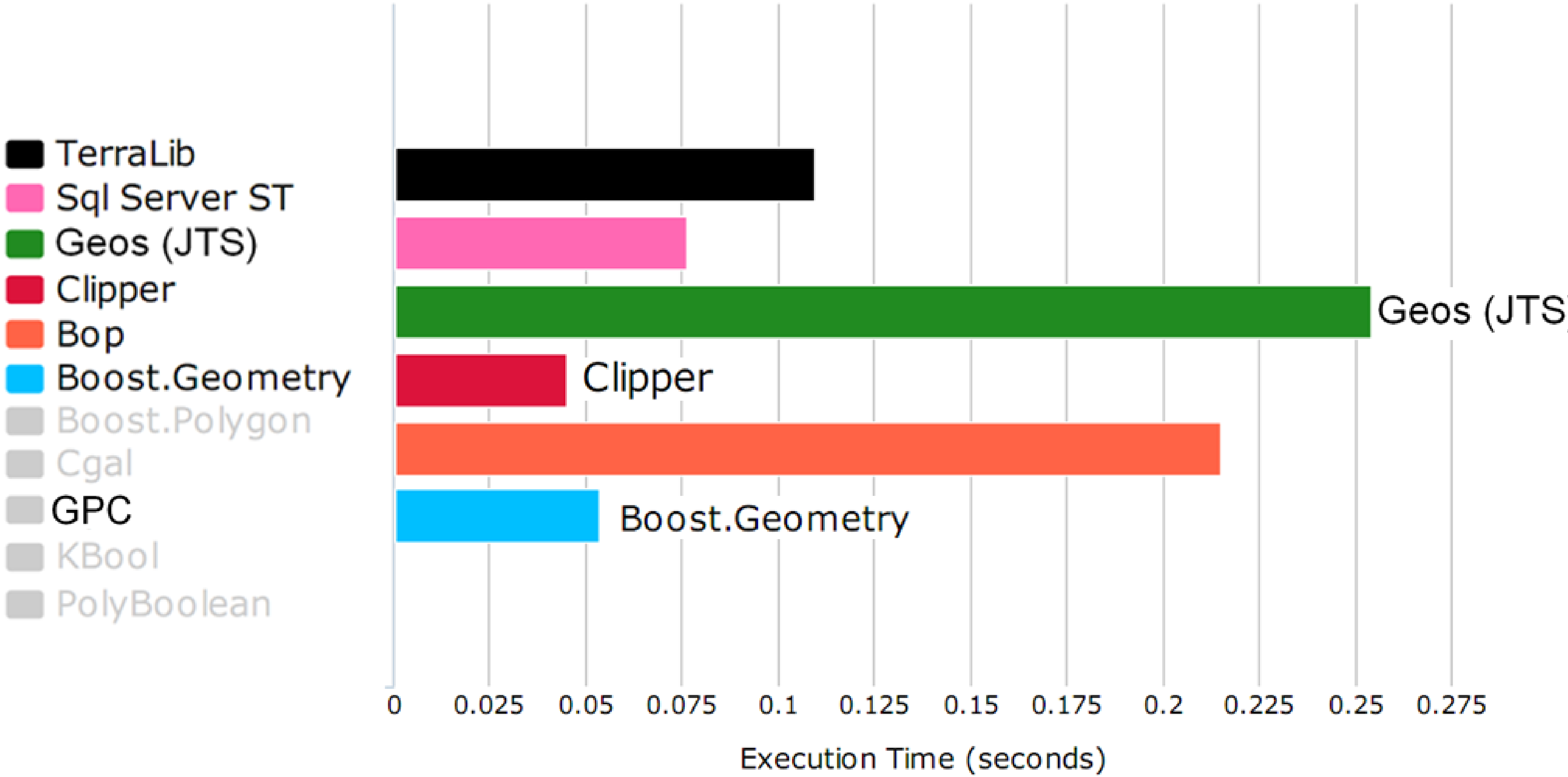
В основе лежит **алгоритм Ватти**.

Многоугольники, имеющие дыры и/или самопересечения, сложные многоугольники, состоящие из нескольких раздельных контуров.

Ссылка:

http://www.angusj.com/delphi/clipper.php

Сравнение библиотек по производительности

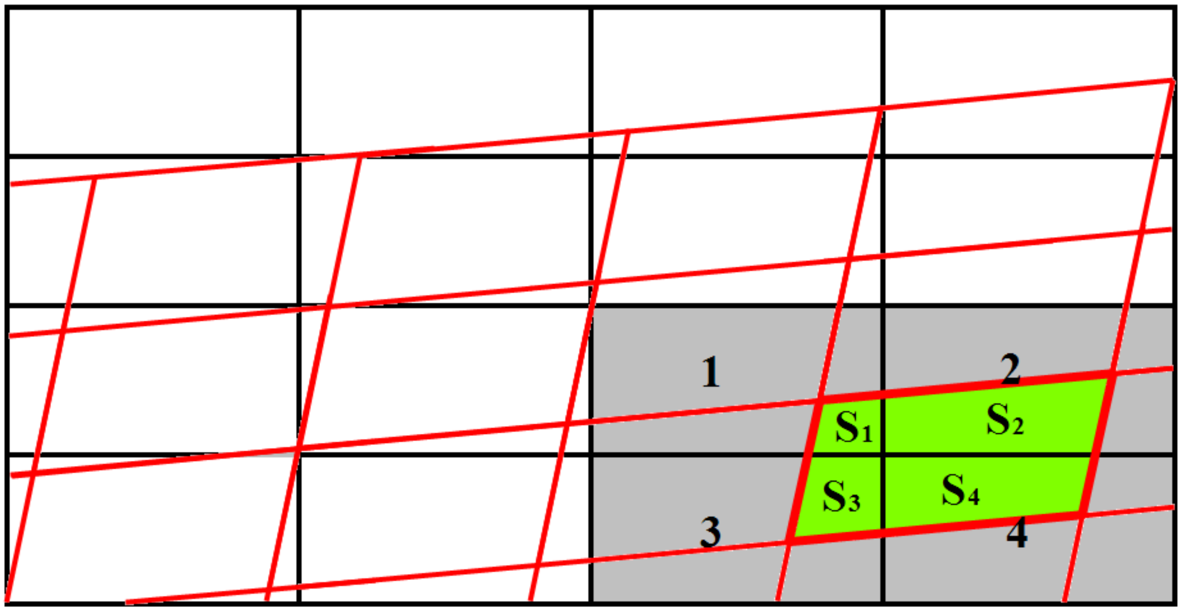


Ссылка: Polygon Clipping: a Wrapper, a Benchmark (http://rogue-modron.blogspot.ru/2011/04/polygon-clipping-wrapper-benchmark.html)

Сравнительное тестирование алгоритма

Тестовый набор 1:

В качестве тестового набора многоугольников использовались координаты ячеек двух расчетных сеток, которые имеют взаимное пересечение.
Данный тест повторяет ситуацию из реальной многофазной задачи газодинамики.



Тестовый набор 1.
Результаты

MAX относительная погрешность (%)

Библиотека Clipper (ver.6.1.3): 5.6×10^{-5}

Разработанный алгоритм: 1.9×10^{-13}

MAX абсолютная погрешность (ед. площади)

Библиотека Clipper (ver.6.1.3): 1.5×10^{-12}

Разработанный алгоритм: 4.9×10^{-24}

Тестовый набор 2:

Данный тестовый набор многоугольников представляет из себя сетку размерностью 100x100, ячейки которой являются квадратами со стороной **А**, на которую произвольным образом накладываются 50 правильных N-угольников со стороной **А**. Наложение происходило таким образом, чтобы все 50 N-угольников не выходили за пределы квадратной сетки.

N варьировалось от 3 до 15.

MS Visual Studio 2010 (x64 release build)
Intel Compiler

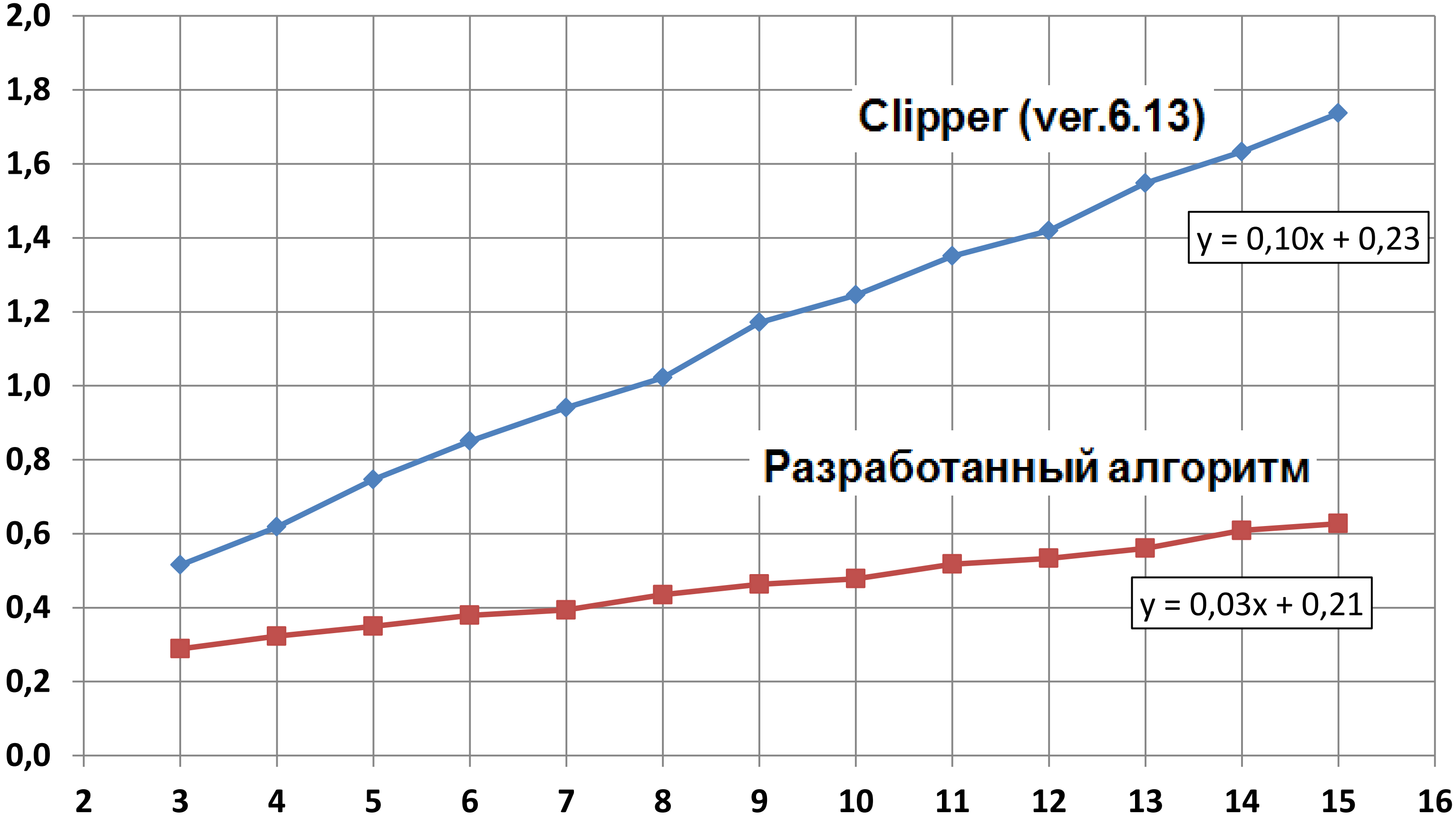


График зависимости времени расчета от числа вершин

MS Visual Studio 2010 (x32 release build)
Intel Compiler

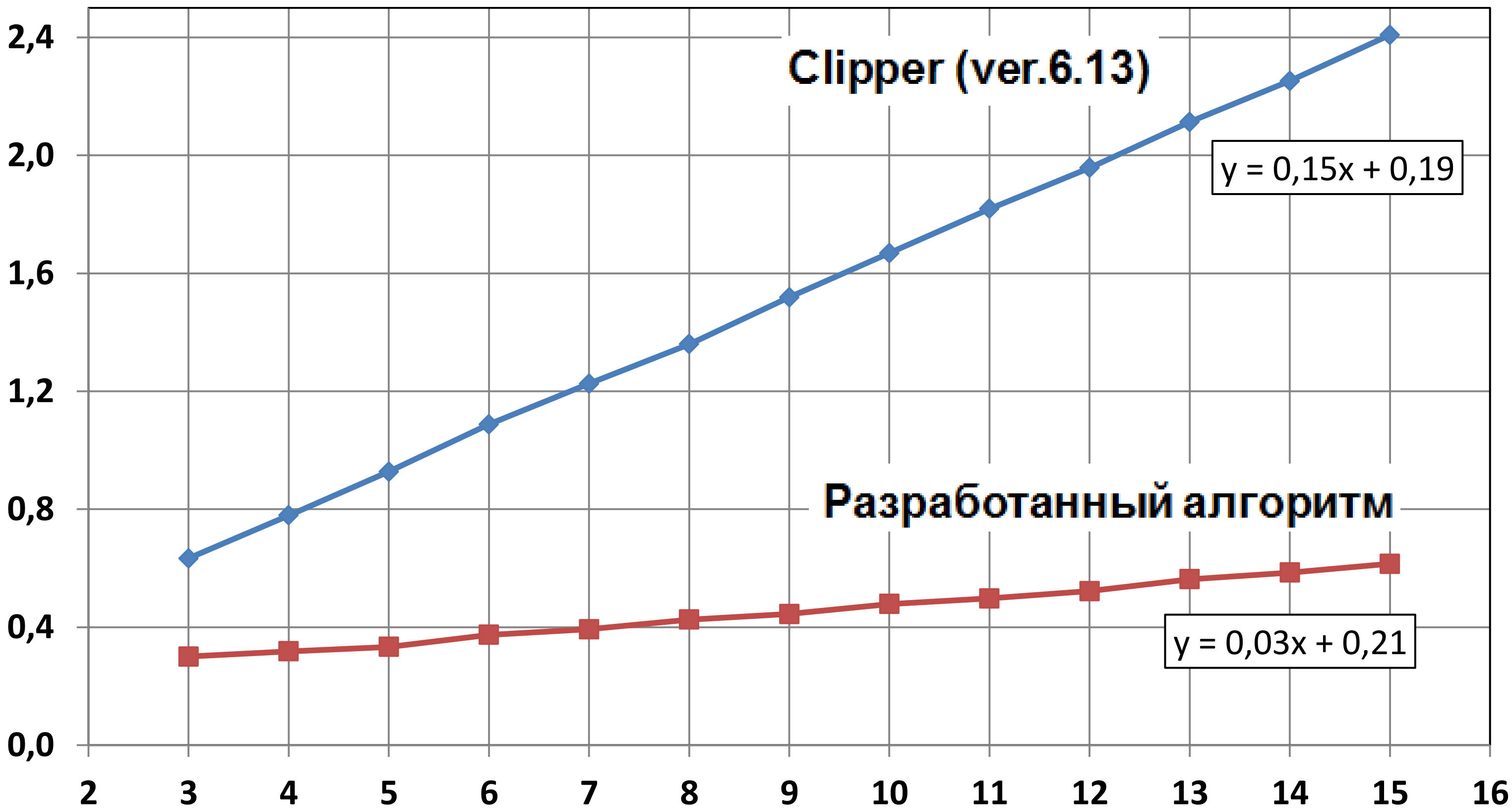


График зависимости времени расчета от числа вершин

Заключение

Разработанный алгоритм поиска пересечений выпуклых многоугольников отвечает изначальным требованиям:

- Высокая чувствительность
- Высокая скорость работы в скалярном режиме
- Определения точек пересечения с высокой точностью
- Устойчивость

В сравнении с некоммерческими программными библиотеками алгоритм показывает лучшую чувствительность и эффективность в скалярном режиме.

Рекомендуется для использования в задачах газодинамики; численных схемах, требующих перестроения расчетной сетки в процессе решения - схемы семейства ALE, метод VOF (Volume Of Fluid).

Спасибо за внимание!