

## Алгоритмы обнаружения коллизий плоских двумерных объектов произвольной формы

*Р.А. Файзрахманов, Р.Т. Мурзакаев, М.Д. Швецов, А.С. Мехоношин,  
А.Ф. Хабибулин*

*Пермский национальный исследовательский политехнический университет*

**Аннотация:** В работе перечислены наиболее распространенные методы обнаружения пересечений и предложен алгоритм для разрешения коллизий плоских двумерных объектов сложной невыпуклой формы. Разработанный алгоритм отличается отсутствием выпуклой декомпозиции невыпуклых объектов. Приведены примеры разрешения коллизий двух невыпуклых объектов. Произведена оценка среднего времени разрешения коллизий. Показана эффективность разработанного алгоритма, который работает более чем в три раза быстрее рассматриваемых аналогов.

**Ключевые слова:** алгоритм обнаружения коллизий, плоские двумерные объекты, невыпуклые контуры, выпуклая декомпозиция, алгоритм GJK/EPA, сумма Минковского, теорема о разделяющих осях.

Задача обнаружения и разрешения коллизий (пересечений объектов) возникает во множестве прикладных областей, таких как САПР, моделирование, физические движки, разработка тренажеров и др. [1-5].

Существующие алгоритмы разрешения столкновений не позволяют с высокой точностью и скоростью обнаружить пересечение двумерных объектов невыпуклой формы [6-8].

Целью статьи является разработка алгоритма разрешения коллизий как выпуклых, так и невыпуклых плоских объектов, отличающегося от существующих меньшими временными затратами.

Для решения данной задачи требуется постоянно измерять расстояние между двумя объектами. Если оно обращается в ноль, то тела соприкоснулись. В этом случае необходимо реализовать «скольжение» контуров без их взаимного проникновения. Поскольку контроль расстояния осуществляется с некоторым временным шагом, то вместо соприкосновения пары тел может возникнуть их взаимное проникновение. В этом случае

возникает задача устранения проникновения или иной реакции на это событие.

Перечислим наиболее распространенные методы обнаружения пересечений плоских двумерных объектов.

1. Набор инструментов «*XNA Game Studio*». Все объекты описываются ограничивающими прямоугольниками, и проверяются их пересечением [6], что негативно сказывается на точности определения коллизий.

2. Связка алгоритмов *GJK* (*Gilbert-Johnson-Keerthi algorithm*) и *EPA* (*Expanding Polytope Algorithm*), которая использует понятие суммы Минковского [7-9]. Тела пересекаются, когда их сумма Минковского содержит начало координат, т.е. нулевой вектор.

3. Метод, основанный на теореме о разделяющих осях (ТРО), согласно которой пара тел не пересекается, если для них существует хотя бы одна разделяющая ось, на которой проекции тел не пересекаются [10].

На практике, теорему применяют для фигур с малым количеством граней, иначе алгоритм требует больших временных и вычислительных затрат [11].

Невыпуклые многоугольники необходимо разбить на выпуклые (рис. 1), чтобы производить обработку столкновений для каждой выпуклой части [12]. Применение выпуклой декомпозиции для методов ТРО и *GJK/EPA* хорошо работает только при небольших пересечениях.

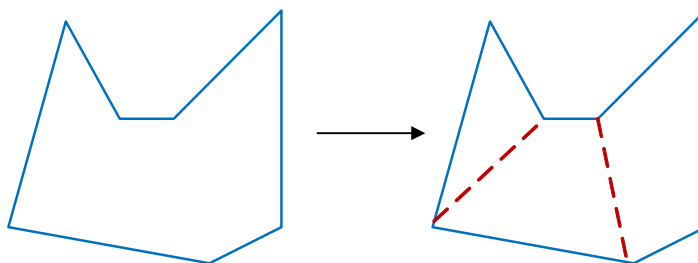


Рис. 1. – Разбиение вогнутого многоугольника на несколько выпуклых

Для разработки алгоритма обнаружения коллизий плоских двумерных объектов произвольной формы был выбран метод, основанный на теореме о разделяющих осях, так как он проще в реализации, чем алгоритм *GJK/EPА* и с высокой точностью позволяет разрешить столкновение объектов.

При проверке пересечений двух тел невыпуклой формы одно рассматривается как неделимый объект (без разбиения), а второе разбивается на выпуклые объекты, соответственно, поочередно проверяется пересечение первого тела с каждым выпуклым объектом второго.

Разработанный алгоритм для объектов произвольной формы состоит из следующих шагов:

1. Для каждого полигона  $PA$  и  $PB$  строим грани по заданным координатам точек фигур:

$$\text{Если } (i + 1 \geq n + 1) \ E = p[1] - p[i], \ i = 1 \dots n$$

$$\text{Иначе } E = p[i + 1] - p[i], \ i = 1 \dots n,$$

где  $i$  – номер точки, принимающий значение от 1 до  $n$  ( $n$  – количество точек полигона);

$p[i]$  – точка фигуры с координатами  $x$  и  $y$ , для которой строим грани;

$E$  – текущая грань.

2. Подсчитываем общее количество граней рассматриваемых объектов ( $k = 1 \dots m$ ,  $m$  – общее количество граней).

3. Берем грань с номером 1 ( $k = 1$ ).

4. Строим разделяющую ось перпендикулярную к текущей грани  $E[k]$ :

$$A = (-Y, X),$$

где  $A$  – разделяющая ось для грани  $E[k]$ ;

$Y$  – координата по  $y$  грани  $E[k]$ , которая используется в качестве координаты  $x$  разделяющей оси  $A$  с отрицательным знаком;

$X$  – координата по  $x$  грани  $E[k]$ , которая используется в качестве координаты  $y$  разделяющей оси  $A$ .

5. Нормируем длину полученной разделяющей оси  $A$ :

$$L = \sqrt{X * X + Y * Y}$$

$$X = \frac{X}{L}$$

$$Y = \frac{Y}{L},$$

где  $L$  – длинна разделяющей оси  $A$ ;

$X$  – координаты по  $x$  разделяющей оси  $A$ ;

$Y$  – координаты по  $y$  разделяющей оси  $A$ .

6. Получаем минимум и максимум проекции полигонов  $PA$  и  $PB$  на разделяющую ось  $A$ , для этого накладываем проекции фигур на данную разделяющую ось:

1) вычисляем скалярное произведение между первой точкой объекта и разделяющей осью  $A$ :

$$s = p[1].X * A.X + p[1].Y * A.Y$$

$$min = s$$

$$max = s,$$

где  $s$  – скалярное произведение между текущей точкой полигона и разделяющей осью  $A$ ;

$p[1].X, p[1].Y$  – координаты по  $x$  и  $y$  текущей точки объекта;

$A.X, A.Y$  – координаты по  $x$  и  $y$  разделяющей оси  $A$ ;

$min, max$  – минимальное и максимальное значение проекции полигона на разделяющую ось  $A$ .

2) вычисляем скалярное произведение между остальными точками объекта и разделяющей осью  $A$ , если полученное значение  $s$  будет меньше текущего минимума или больше текущего максимума, то переменной  $min$  и  $max$  присваиваем новые значения:

$$s = p[i].X * A.X + p[i].Y * A.Y, i = 2 \dots n$$

$$\text{Если } (s < min) \text{ } min = s$$

Иначе Если ( $s > \max$ )  $\max = s$

7. Вычисляем расстояние между проекциями полигонов  $PA$  и  $PB$ , с помощью переменных  $\min A$ ,  $\max A$ ,  $\min B$ ,  $\max B$ :

Если ( $\min A < \min B$ )  $I = \min B - \max A$

Иначе  $I = \min A - \max B$ ,

где  $I$  - дистанция между проекциями объектов  $A$  и  $B$ .

8. Если расстояние  $I$  между проекциями больше нуля, значит, полигоны не пересекаются, переходим на шаг 16, иначе на шаг 9.

9. Если текущая грань принадлежит объекту  $PA$  и данный объект является невыпуклым, то вычисляем расстояние между проекциями объекта  $PB$  и текущей грани  $E[k]$  на разделяющую ось  $A$ , иначе переходим на шаг 11:

1) в качестве минимума и максимума проекции грани  $E[k]$  возьмем следующие значения:

$$\min = p[k].X * A.X + p[k].Y * A.Y, k = 1 \dots n$$

$$\max = p[k].X * A.X + p[k].Y * A.Y, k = 1 \dots n,$$

где  $p[k].X$ ,  $p[k].Y$  - координаты по осям  $x$  и  $y$  точки полигона  $PA$  ( $k$  совпадает с номером текущей грани объекта  $PA$ ).

2) вычисляем расстояние между проекциями объекта  $PB$  и гранью:

$$E[k]: \text{Если } (\min < \min B) \quad I = \min B - \max,$$

$$\text{Иначе } I = \min - \max B$$

3) если расстояние  $I$  между проекциями больше нуля, значит, полигон  $PB$  и грань  $E[k]$  не пересекаются, в этом случае, если текущая грань является последней из общего количества граней объектов  $PA$ , переходим на шаг 10, иначе переходим к следующей грани ( $k + 1$ ) на шаг 4; если  $I \leq 0$  (есть пересечение) переходим на шаг 11.

10. Если ни одна из граней полигона  $PA$  (объект  $PA$  является невыпуклым) не пересекается с полигоном  $PB$ , то полигоны точно не пересекаются, переходим на шаг 16, иначе на шаг 11.

11. Проверяем, является ли вычисленное расстояние  $I$  для текущей разделяющей оси  $A$  минимальным по сравнению со значениями, полученными на других возможных разделяющих осях объектов  $PA$  и  $PB$ :

$$\text{Если } (I < D) \ D = I; \ V = A,$$

где  $D$  – минимальное расстояние между проекциями полигонов по умолчанию  $D = +\infty$ ;

$V$  – вектор, равный разделяющей оси, на которой наложение проекций полигонов минимально.

12. Если текущая грань является последней из общего количества граней объектов  $PA$  и  $PB$ , то переходим на шаг 13, иначе переходим к следующей грани ( $k + 1$ ) на шаг 4.

13. Получаем вектор перемещения, необходимый для вывода объектов  $A$  и  $B$  из состояния пересечения:

$$V = V * D$$

14. Определяем знак вектора перемещения:

$$C = CPA - CPB$$

$$\text{Если } (C.X * V.X + C.Y * V.Y < 0) \ V = -V,$$

где  $CA, CB$  - координаты центров объектов  $PA$  и  $PB$ ;

$C$  – разница между центрами объектов;

$C.X, C.Y$  – координаты по осям  $x$  и  $y$  переменной  $C$ .

$V.X, V.Y$  – координаты по осям  $x$  и  $y$  вектора перемещения  $V$ .

15. Смещаем полигон  $PA$  на полученный вектор перемещения  $V$  для вывода объектов из состояния пересечения.

16. Конец.

Было проведено тестирование работы алгоритма на нескольких наборах невыпуклых объектов. На рис. 2 представлен пример разрешения коллизий двух невыпуклых объектов разработанным алгоритмом.

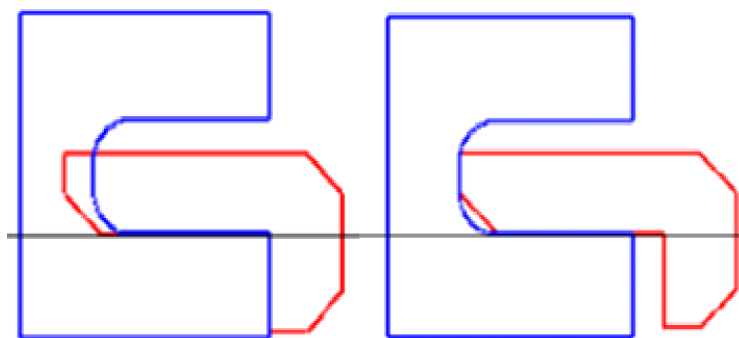


Рис. 2. – Пример разрешения коллизий двух невыпуклых объектов

Предложенный алгоритм, алгоритмы «ТРО» и *GJK/EPA* были протестированы на одинаковых наборах невыпуклых объектов в идентичных условиях (результаты представлены на рис. 3).

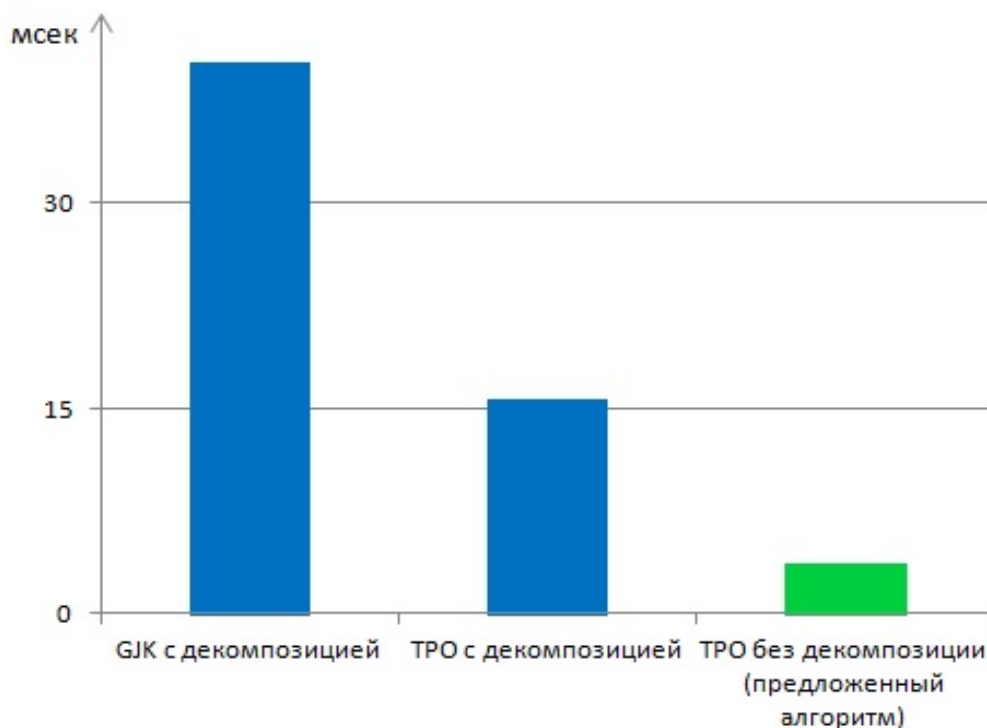


Рис. 3. – Среднее время разрешения коллизий (мсек)

Как видно из гистограммы, предложенный алгоритм работает в 4 раза быстрее алгоритма «ТРО» и 10 раз быстрее *GJK/EPA* с выпуклой декомпозицией.

Таким образом, в работе предложен алгоритм, отличающийся отсутствием выпуклой декомпозиции невыпуклых объектов, показана эффективность разработанного алгоритма, который работает в среднем в три - четыре раза быстрее по сравнению с алгоритмом «ТРО», использующего выпуклую декомпозицию, и в восемь – десять раз, чем алгоритм *GJK/EPA*.

### Литература

1. Файзрахманов Р.А., Бакунов Р.Р., Мехоношин А.С. Создание трехмерных моделей для системы визуализации тренажерного комплекса // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. С. 25-30.
  2. Файзрахманов Р.А., Мехоношин А.С., Бакунов Р.Р., Федоров А.Б., Бикметов Р.Р. Особенности разработки и реализации мобильных пультов тренажерного комплекса оператора портального крана // Инженерный вестник Дона, 2012, №4 URL: [ivdon.ru/magazine/archive/n4p1y2012/1267](http://ivdon.ru/magazine/archive/n4p1y2012/1267).
  3. Fayzrakhmanov R.A., Murzakaev R. T., Mezentsev A. S., Shilov V. S. Applying the greedy algorithm for reducing the dimensionality of the dynamic programming method in solving the one-dimensional cutting stock problem // Middle-East Journal of Scientific Research. – 2014. – № 19 (3). – pp. 412-416.
  4. Fayzrakhmanov R.A., Murzakaev R. T., Mezentsev A. S., Shilov V. S. Application of the Group Decoder for Solving the Orthogonal Materials Cutting Problem // World Applied Sciences Journal. – 2013. – № 28 (10). – pp. 1361-1365.
  5. Мурзакаев Р.Т., Шилов В.С., Буркова А.В. Основные методы решения задачи фигурной нерегулярной укладки плоских деталей // Инженерный вестник Дона, 2013, №4 URL: [ivdon.ru/magazine/archive/n4y2013/2043](http://ivdon.ru/magazine/archive/n4y2013/2043).
-



6. Взаимодействия 2D объектов (столкновения) // URL: [xnadev.ru/articles.php?article\\_id=72](http://xnadev.ru/articles.php?article_id=72) (Дата обращения: 26.09.15).

7. Yalmar Ponce Atencio, Claudio Esperanca и др. [всего 4 автора]. Collision Detection and Response Scheme for Simplified Physically Based Animation // URL: [lcg.ufrj.br/Members/esperanc/atencioy\\_collision.pdf](http://lcg.ufrj.br/Members/esperanc/atencioy_collision.pdf) (Date of check: 26.09.15).

8. Карасёв Р.Н. Мера невыпуклости плоских множеств и сумма Минковского URL: [rkarasev.ru/common /upload/minkowski\\_sum\\_rus.pdf](http://rkarasev.ru/common/upload/minkowski_sum_rus.pdf) (Дата обращения: 26.09.15).

9. Ермолин Е. Н. Методы определения и разрешения столкновений на полигональных моделях: дис. канд. физико-математических наук: 05.13.18. Новосибирск, 2011. – 155 с.

10. Johnny Huynh. Separating Axis Theorem for Oriented Bounding Boxes // URL: [jkh.me/files/tutorials/Separating%20Axis%20Theorem%20for%20Oriented%20Bounding%20Boxes.pdf](http://jkh.me/files/tutorials/Separating%20Axis%20Theorem%20for%20Oriented%20Bounding%20Boxes.pdf) (Date of check: 26.09.15).

11. Мурзакаев Р.Т., Швецов М.Д., Брюханова А.А. Алгоритмы распознавания столкновений // Международная научно-практическая конференция «Тенденции развития точных, естественных и гуманитарных наук - 2014» (сборник статей), Брянск, 21-23 июля 2014. – Брянск, 2014. – С. 3-8.

12. Jyh-Ming Lien, Nancy M. Amato. Approximate Convex Decomposition of Polygons // URL: [cs.gmu.edu/~jmlien/research/app-cd/p218-lien.pdf](http://cs.gmu.edu/~jmlien/research/app-cd/p218-lien.pdf) (Date of check: 26.09.15).

### References

1. Fayzrakhmanov R.A., Bakunov R.R., Mekhonoshin A.S. Vestnik Permskogo nacional'nogo issledovatel'skogo politehnicheskogo universiteta. Jeлектротехника, informacionnye tehnologii, sistemy upravlenija. 2012. № 6. pp. 25-30.

---

2. Fayzrakhmanov R.A., Mekhonoshin A.S., Bakunov R.R., Fedorov A.B., Bikmetov R.R. Inzhenernyj vestnik Dona (Rus), 2012, №4 URL: [ivdon.ru/magazine/archive/n4p1y2012/1267](http://ivdon.ru/magazine/archive/n4p1y2012/1267).
  3. Fayzrakhmanov R.A., Murzakaev R. T., Mezentsev A. S., Shilov V. S. Applying the greedy algorithm for reducing the dimensionality of the dynamic programming method in solving the one-dimensional cutting stock problem. RMiddle-East Journal of Scientific Research. 2014. № 19 (3). pp. 412-416.
  4. Fayzrakhmanov R.A., Murzakaev R. T., Mezentsev A. S., Shilov V. S. Application of the Group Decoder for Solving the Orthogonal Materials Cutting Problem. World Applied Sciences Journal. 2013. № 28 (10). pp. 1361-1365.
  5. Murzakaev R.T., Shilov V.S., Burkova A.V. Inzhenernyj vestnik Dona (Rus), 2013, №4 URL: [ivdon.ru/magazine/archive/n4y2013/2043](http://ivdon.ru/magazine/archive/n4y2013/2043).
  6. Vzaimodejstviya 2D obyektov (stolknoveniya) [Interactions 2D objects (collision)]. URL: [xnadev.ru/articles.php?article\\_id=72](http://xnadev.ru/articles.php?article_id=72).
  7. Yalmar Ponce Atencio, Claudio Esperanca etc. Collision Detection and Response Scheme for Simplified Physically Based Animation. URL: [lcg.ufrj.br/Members/esperanc/atencioy\\_collision.pdf](http://lcg.ufrj.br/Members/esperanc/atencioy_collision.pdf).
  8. Karasjov R. N. Mera nevyuklosti ploskih mnozhestv i summa Minkovskogo [The measure of non-convexity of plane sets and Minkowski sum] URL: [rkarasev.ru/common/upload/minkowski\\_sum\\_rus.pdf](http://rkarasev.ru/common/upload/minkowski_sum_rus.pdf).
  9. Ermolin E. N. Metody opredeleniya i razresheniya stolknovenij na poligonal'nyh modeljah [Methods for determining and resolving conflicts on polygonal models]: dis. kand. fiziko-matematicheskikh nauk: 05.13.18. Novosibirsk, 2011. 155 p.
  10. Johnny Huynh. Separating Axis Theorem for Oriented Bounding Boxes. URL: [jkh.me/files/tutorials/Separating%20Axis%20Theorem%20for%20Oriented%20Bounding%20Boxes.pdf](http://jkh.me/files/tutorials/Separating%20Axis%20Theorem%20for%20Oriented%20Bounding%20Boxes.pdf).
-



11. Murzakaev R.T., Shvecov M.D., Bryukhanova A.A. Mezhdunarodnaya nauchno-prakticheskaya konferentsiya «Tendentsii razvitiya tochnykh, estestvennykh i gumanitarnykh nauk - 2014» (sbornik statey), Bryansk, 21-23 iyulya 2014. Bryansk, 2014. pp. 3-8.

12. Jyh-Ming Lien, Nancy M. Amato. Approximate Convex Decomposition of Polygons. URL: [cs.gmu.edu/~jmlien/research/app-cd/p218-lien.pdf](http://cs.gmu.edu/~jmlien/research/app-cd/p218-lien.pdf).