Поиск корней

Сергей Зефиров

4 декабря 2004 г.

Аннотация

Рассмотрена проблема определения времени столкновения произвольно перемещающегося объекта и статического объекта, даны примеры функций «расстояний», приведены механизмы нахождения времени столкновения на основе интервальной и аффинной арифметик.

1 Введение

Все началось с того, что одним из источников ошибок в нашей физике являлось раздельное определение количества перемещения и количества поворота без столкновения (которые возвращались, как числа в диапазоне [0..1]), поскольку наша библиотека умела либо определять время столкновения – что использовалось для определения количества перемещения, – либо проверять факт проникновения коллижн-форм друг в друга, что и позволяло определить количество поворота[1].

Меня посетила мысль о том, что, по идее, можно определять столкновение с учетом одновременного поворота. Вопрос в том, как?

Некоторые товарищи используют аппроксимацию движения с поворотом – они представляют преобразование координат в процессе движения тела так называемым «винтовым движением»[2]. В процессе этого движения происходит перемещение тела по какому-то направлению $\overline{d}$ и поворот на угол α вокруг этой оси по радиусу R[3]. В результате получаются степенные уравнения 3-ей степени, которые решаются известными методами.

Я решил пойти другим путем. Можно сказать, инженерным. Во-первых, идеальная точность нам не нужна, скорее, нужна возможность выбора точности вычислений. Во-вторых, желательно оставить возможность произвольного представления траектории, а не ограничиваться каким-либо одним, пусть даже и «винтовым движением». Все это вместе привело к решениям, которые описаны ниже.

[1] Путем двоичного деления - если в диапазоне количества поворота $[a..b]$ в точке a не было проникновения, а в точке b – было, то бралась точка $c = (a + b)/2$, и формировался новый интервал количества поворота – $[a..c]$ или $[c..b]$. И так до тех пор, пока длина интервала не становилась достаточно малой.

[2] «screw motion»

[3] Подробнее можно узнать из творчества г-жи Redon[1]

2 Определение количества свободного движения в произвольном случае

Наш объект, пока неизвестной формы, движется по прямой (более того, он может двигаться и по параболе $\overline{r}(t) = \overline{r}_0 + \overline{v}t + \overline{a}\frac{t^2}{2}$) из точки $\overline{r}_0$ со скоростью $\overline{v}$ и вращается с угловой скоростью $\overline{\omega}$ от начальной ориентации $\overline{\alpha}_0$.

И, возможно, сталкивается с другим объектом, тоже неизвестной формы, но который стоит на месте.

Нам надо определить наличие столкновения на интервале времени $t \in [0..1]$[4] и время t_c при котором объекты столкнуться.

Для заданных угловых скоростей мы можем получить разложение матрицы ориентации в матрицу степенных рядов. Для двумерного случая $bfM = \begin{pmatrix} \cos(\alpha_0 + \omega t) & -sin(\alpha_0 + \omega t) \\ \sin(\alpha_0 + \omega t) & cos(\alpha_0 + \omega t) \end{pmatrix}$ это будет преобразовано в матрицу $bfM = \begin{pmatrix} \sum_{i=0}^{N_{11}} c_{11i}t^i & \sum_{i=0}^{N_{12}} c_{12i}t^i \\ \sum_{i=0}^{N_{21}} c_{21i}t^i & \sum_{i=0}^{N_{22}} c_{22i}t^i \end{pmatrix}$, где каждый ее элемент представляет собой степенной ряд, длина которого ограничена необходимой точностью.

Какими, примерно, могут быть длины этих рядов? Для угла $\alpha_0 = 2$ и скорости $\omega = 0.1$[5], $\sin(\alpha_0 + \omega t) = \sin(\alpha_0)\cos(\omega t) + \cos(\alpha_0)\sin(\omega t) = 0.909297426826 - 0.0416146836547t^1 + 0.00454648713413t^2 - 6.93578060913 * 10^{-05}t^3 \ldots$. Уже пятый элемент существенно меньше 0.001, то есть можно ожидать длину рядов в четыре-пять элементов.

Для двух объектов мы можем написать некоторую функцию расстояния $D(t)$, зависящую от положений объектов и их ориентаций $D(t) = d(\overline{r}_a(t), \overline{r}_b, \overline{\alpha}_a(t), \overline{\alpha}_b)$, которая больше нуля, если объекты разнесены в пространстве и меньше или равна нулю, если объекты проникли друг в друга или соприкасаются.

Например для двух сфер это будет функция, которая вычисляет разность между квадратом расстояния между центрами и квадратом суммы радиусов:

$$\begin{aligned} D(t) &= (\overline{r}_a(t) - \overline{r}_b)^2 - (R_a + R_b)^2 \\ &= [(x_a(t) - x_b)^2 + (y_a(t) - y_b)^2 + (z_a(t) - z_b)^2] - (R_a + R_b)^2. \end{aligned} \tag{1}$$

Для нахождения факта столкновения изначально разнесенных в пространстве ($D(t) > 0$) сфер (любых двух объектов) нам надо найти:

- Меняет ли функция $D(t)$ знак на диапазоне $[0..1]$. Если она меняет знак, то сферы (объекты) сталкиваются.
- Самое маленькое значение $t \in [0..1]$, при котором $D(t) = 0$. Это и будет время столкновения.

[4]Этот интервал представляет собой полное время одного цикла перемещения, а он меньше или равен 0.1 секунды для нашей физики.

[5]Напомню, что для временного интервала обновления физики в 0.1 секунды эта скорость соответствует 1.0 радиана в секунду, то есть, примерно 57 градусов в секунду. Это обычная скорость в наших обычных условиях.

Даже для столь простых объектов, как сферы, значительное вращение приводит к $D(t)$ в виде полинома большой степени - четвертой-пятой. Прямого метода нахождения корней таких полиномов нет, но можно воспользоваться итеративным способом. Один из таких методов называется метод Ньютона-Рафсона (Newton-Rafson method), и он состоит в нахождении одного из корней многочлена с последующим исключением этого корня из многочлена. Другой метод сводится к использованию интервальной арифметики и аффинной арифметики.

3 Интервальная арифметика

Интервальная арифметика оперирует с интервалами.

Интервал $t = [a..b]$ представлен парой $(a, b), a \leq b$.

Для интервалов определены обычные операции:

- Сложение: $[a..b] + [c..d] = [a + c..b + d]$
- Вычитание: $[a..b] - [c..d] = [a - d..b - c]$
- Умножение: $[a..b] \times [c..d] = [min(min(ac, ad), min(bc, bd))..max((max(ac, ad), max(bc, bd))]$
- Деление:
  $\frac{1}{[a..b]} = [1/b..1/a] \Leftarrow a > 0$ или $b < 0$,
  $\frac{[a..b]}{[c..d]} = [a..b] \times \frac{1}{c..d]}$.
- Взятие абсолютной величины:
  $abs([a..b]) = [a..b] \Leftarrow a \geq 0$
  $abs([a..b]) = [-b.. - a] \Leftarrow b < 0$
  $abs([a..b]) = [0..max(-a, b)]$ во всех остальных случаях.

Операции с интервалами дают результирующий интервал $[a..b]$, который включает в себя истинный интервал выражения $[A..B]$. Например, выражение $x - x^2$ для $x = [0..1]$ даст результирующий интервал $[0..1] - [0..1] * [0..1] = [-1..1]$, когда на самом деле результирующий интервал $[0..0.25]$ (максимум находится в $x = 0.5$).

Общий алгоритм отыскания наименьшего корня полинома $P(t)$ в диапазоне $t \in [0..1]$ для интервальной арифметики таков (половинное деление):

- Для текущего интервала t вычислить интервал изменений $d = [a..b] = P(t)$ с помощью заданных выше операций.
- Если $d.a > 0$, то есть, есть пересечение объектов отсутствует, то вернуть $false$.
- В противном случае, если длина интервала $|t| < \epsilon$, вернуть $true$ и время пересечения $(t.a + t.b)/2$.
- Иначе - продолжить работу:

– Разбить интервал t на два интервала $tlow = [t.a..(t.a + t.b)/2]$ и $thigh = [(t.a + t.b)/2..t.b]$.

– Проверить наличие корня в диапазоне $tlow$. Если есть, вернуть $true$ и вычисленное при проверке наличия корня значение времени пересечения.

– Проверить наличие корня в диапазоне $thigh$. Если есть, вернуть $true$ и вычисленное при проверке наличия корня значение времени пересечения.

– Вернуть $false$ – неуспех.

4 Определение времени пересечения параллепипедов(OBB)

Два OBB характеризуются положениями $\overline{r}_a$ и $\overline{r}_b$, направляющими векторами единичной длины $\overline{e}_i$ и $\overline{f}_j$ и половинными размерами по этим векторам a_i и b_i, $a_i, b_j \geq 0$.

Проверка факта пересечения двух OBB сводится к нахождению разделяющей плоскости. Нормаль разделяющей плоскости n должна удовлетворять условию:

$$|\overline{n}(\overline{r}_a - \overline{r}_b)| - (\sum_{i=1}^{3} a_i|\overline{ne}_i| + \sum_{j=1}^{3} b_j|\overline{n}\overline{f}_j|) > 0.$$

Нормаль $\overline{n}$ выбирается из множества $n \in \{\overline{e}_i, \overline{f}_j, \overline{e}_i \times \overline{f}_j; i, j \in [1..3]\}$ И всего получается пятнадцать проверок.

Мы снова запустим в пространстве параллепипед a, а второй параллепипед оставим на месте. Тогда положение и направляющие вектора будут зависеть от времени – $\overline{r}_a(t)$, $\overline{e}_i(t)$. И мы получим, соответственно, $n(t)$ и набор из пятнадцати «расстояний», но уже зависящих от времени:

$$D_k(t)|\overline{n}_k(t)(\overline{r}_a(t) - \overline{r}_b)| - (\sum_{i=1}^{3} a_i|\overline{n}_k(t)\overline{e}_i(t)| + \sum_{j=1}^{3} b_j|\overline{n}_k(t)\overline{f}_j|) > 0.$$

Отдельные слагаемые $D_k(t)$ константно равны 0 или 1. Их можно исключить из вычислений.

Отдельные варианты $D_k(t)$ могут быть положительны уже для интервала $t \in [0..1]$, и их тоже можно не учитывать.

Алгоритм нахождения факта и времени столкновения будет изменен:

- Для текущего интервала t вычислить интервалы изменений $d_k = [a..b] = D_k(t)$.

- Если для всех k, $d_k.a > 0$, то есть, есть пересечение объектов отсутствует, то вернуть $false$.

- В противном случае, если длина интервала $|t| < \epsilon$, вернуть $true$ и время пересечения $(t.a + t.b)/2$.
- Иначе - продолжить работу:
  - Разбить интервал t на два интервала $tlow = [t.a..(t.a + t.b)/2]$ и $thigh = [(t.a + t.b)/2..t.b]$.
  - Убрать из набора $\{k\}$ те k, для которых $d_k.a > 0$.
  - Проверить наличие корня в диапазоне $tlow$. Если есть, вернуть $true$ и вычисленное при проверке наличия корня значение времени пересечения.
  - Проверить наличие корня в диапазоне $thigh$. Если есть, вернуть $true$ и вычисленное при проверке наличия корня значение времени пересечения.
  - Вернуть $false$ – неуспех.

5 Несколько коллижн-форм

Поскольку у нас много объектов с несколькими коллижн-формами и у нас уже есть алгоритм для проверки нескольких расстояний за раз, то его легко можно обобщить для нескольких коллижн-форм. Добавим к нашим 15-ти $D_k(t)$ для OBB еще и шестнадцатую (или несколько $D_l(t)$) $D_16(t)$ для сферы.

Соответственно, мы можем легко проверить столкновение с несколькими треугольниками земли.

И тд., и тп.

6 Аффинная арифметика

Интервальная арифметика, хоть и удобное средство, страдает значительным недостатком: слишком большая величина ошибки. В частности, интервальная арифметика не учитывает зависимостей между переменными – выражение $x - x$ не даст 0 в результате, если интервал изменений x отличен от константы.

Аффинная арифметика[6][2] дает такую возможность. В аффинной арифметике слагаемое с ошибкой представлено в виде суммы «центра» и «смещений-ошибок»:

$$\tilde{x} = x + \sum_{i=1}^{N} x_i e_i.$$

В этой формуле-определении e_i – это ошибочный множитель, про который известно только, что он изменяется в диапазоне $[-1..1]$. x_i – это «амплитуда» ошибки, она может быть как положительной, так и отрицательной. x – это центр.

[6]Здесь рассматривается версия только для одной переменной.

$t \in [0..1]$ запишется в аффинной арифметике как $\tilde{t} = 0.5 + 0.5e_1$.

Вот сложение и вычитание для двух аффинных диапазонов:

$$\tilde{x} \pm \tilde{y} = (x \pm y) + \sum_{i=1}^{N} (x_i \pm y_i) e_i.$$

И выражение $x - x$, исходя из этого определения, в аффинной арифметике будет константно равно 0, что в бесконечность раз лучше, чем $[-1..1]$ интервальной арифметики.

Умножение добавляет к нашим ошибочным множителям еще один (γ), поскольку, по идее, умножение двух переменных некогерентно ни одному сомножителю:

$$\tilde{x} \times \tilde{y} = xy + y \sum_{i=1}^{N} y_i e_i + x \sum_{j=1}^{N} x_j e_j + \gamma e_{N+!}$$

Как вычислить γ? Распишем наше произведение менее подробно:

$$\tilde{x} \times \tilde{y} = (x + x_e) \times (y + y_e), \text{где } x_e \text{ и } y_e \text{ – суммы ошибок,}$$

$$\tilde{x} \times \tilde{y} = xy + xy_e + yx_e + x_e y_e = xy + xy_e + yx_e + \gamma e_{N+1}.$$

Получается, что $\gamma = x_e y_e = (\sum_{i=1}^{N} x_i e_i)(\sum_{j=1}^{N} y_j e_j)$. Наиболее простая форма оценки для $\gamma = (\sum_{i=1}^{N} |x_i|)(\sum_{j=1}^{N} |y_j|)$

Возьмем пример $x - x^2$ где $x \in [0..1]$. Аффинное представление для $x = 0.5 + 0.5e_1$. Квадрат x: $x^2 = 0.25 + 0.5e_1 + 0.25e_2$. Вычтем x^2 из x:

$$(0.5 + 0.5e_1) - (0.25 + 0.5e_1 + 0.25e_2) = 0.25 - 0.25e_2.$$

Переведя полученное представление обратно в интервальное представление, получим:

$$x - x^2 = 0.25 - 0.25e_2 \in [0..0.5].$$

Длина этого интервала 0.5, что в два раза хуже, чем длина реального интервала (0.25, см. выше), но в четыре раза лучше, чем вычисление с помощью интервальной арифметики.

В нашей ситуации – нахождение корней в фиксированном диапазоне функции от одной переменной, – возможны разные оптимизации. Во-первых, нам не надо брать модули при вычислении новых ошибок при умножении, поскольку «степень» $t = t_0 + \Delta t e_1$ будет иметь только положительные коэффициенты при ошибках. Во-вторых... Можно придумать что-нибудь еще.

Операция взятия модуля для аффинной арифметики нетривиальна (она действительно сложна). Поэтому можно перед взятием модуля перевести значение из аффинного представление в интервальное, взять модуль и дальше работать в интервальном представлении. Все равно нам надо получить результат в интервальном представлении, а потеря точности не так уж и важна.

7 Заключение

Итак, вашему вниманию была представлена идея с определением времени столкновения объекта, испытывающего произвольное перемещение и статического объекта. Она позволяет, до некоторой степени, унифицировать формулы определения времени столкновения вообще.

На данный момент я затрудняюсь оценить затраты времени выполнения на такую схему. Единственное, на что хочу обратить внимание, что интервальная арифметика дает **слишком** большую ошибку вычислений – например, на тестовых примерах при ошибке определения времени $\epsilon = 0.001$, что дает глубину поиска 10 или 20 интервальных вычислений в идеальном случае, интервальная арифметика давала 160-280 интервальных вычислений.

Оценки на реализацию: в процессе изучения самой возможности осуществления определения времени столкновения произвольно движущегося объекта афинная арифметика[7] была реализована за два дня, вместе с процедурой нахождения корня в интервале (что служило проверкой правильности). После нахождения функций «расстояния» для объектов разных типов добавление их в общую схему становится тривиальным.

В-общем, на это стоит посмотреть пристальней, наверное, стоит попробовать. Когда будет время.

Список литературы

[1] Continuos Collision Detection for Rigid and Articulated Bodies, Stephanie Redon, 2004.

[2] Поиск в Google терминов «affine interval arithmetic» дает на первой странице всю необходимую информацию.

[7] Точнее, афинное представление с одним ошибочным членом, что было неправильно, но почему-то успешно работало на тестовых задачах.