

Эффективные интервальные алгоритмы нахождения пересечений поверхностей в компьютерной графике.

П. С. ВЕНГЕРСКИЙ

Львовский национальный университет им.И.Франко

e-mail: p_vengersky@franko.lviv.ua

Одной с основных задач компьютерной графики является аппроксимация формы поверхности или кривой набором простых фигур. Задача более усложняется при комбинации поверхностей, когда результат не представляется явно заданием функции результата. Прямым методом решения такого рода задач является полный перебор всех пикселей, которые удовлетворяют заданному условию. Используя методы интервального анализа, можно достигнуть улучшения быстродействия алгоритмов, поскольку они работают не со всеми пикселями из рассмотренной области определения, а только с граничными пикселями этой области.

Использование интервальных вычислений разрешает эффективно работать со сложными дискретными моделями представления поверхностей неявно заданными функциями трех переменных[1,2]. Интервальные операции используют для дискретного представления модели объекта с помощью сильно уплотненной сетки. Сетка сохраняется в виде восьмиарного дерева для экономии памяти компьютера и акселерации процесса визуализации поверхности. Учитывая, что элемент сетки достаточно мал, он считается трехмерной точкой, которая может быть отображена как объемный пиксель(воксель) или исследована с целью генерации треугольных примитивов внутри элемента, что позволяет построить полигональное представление заданной функции.

На этапе уточнения решения задачи использовались интервальные итерационные методы Ньютона и типа Рунге[3]. Проведено сравнения эффективности рассмотренных методов, а также быстродействие разработанных алгоритмов с другими подходами решения такого рода задач. Разработано программное обеспечение для реализации этих алгоритмов.

1. Постановка задачи

Дано некоторую аналитическую неявно заданную кривую, как пересечение двух поверхностей $F(x,y,z)=0$ и $G(x,y,z)=0$ где $(x, y, z) \in D = \{(x, y, z) : x \in D_1, y \in D_2, z \in D_3\}$ т.е. необходимо решить систему:

$$\begin{cases} F(x, y, z) = 0 \\ G(x, y, z) = 0 \end{cases}$$

Таким образом, на экране монитора нам надо активизировать точки, принадлежащие данной функции и лежат в заданном интервале. Задачу разделим на два этапа: первый этап заключается в аппроксимации формы кривой, второй - в уточнении решения.

Алгоритм аппроксимации формы кривой состоит из следующих трех шагов: нахождение допустимой области, тест на глобальную параметризацию, нахождение граничных пикселей области кривой. Рассмотрим каждый из этих шагов.

2. Нахождение допустимой области функции

Неявно заданная кривая - это решение системы ограничений. Систему ограничений можно представить как функцию $F(x) = 1$, $x \in X \subset R^n$; $F : R^n \rightarrow R$ принимающая значения один, если условия выполняются и ноль в противном случае. Если функция включения $\otimes F$ для F в области $X \in I^n$ принимает значение один, тогда X называют допустимой областью. Для этого на каждой под области необходимо вычислить функции включения для $F(x, y, z)$ и $G(x, y, z)$. Функция включения для $F(x, y, z)$ определяется так:

$$\otimes f(Y) \equiv f(c) +_{\otimes} \otimes f'(Y) *_{\otimes} (Y -_{\otimes} c), \quad c \in Y \quad , \quad \otimes f'(Y) \equiv \left[\otimes \frac{\partial f_i}{\partial x_i}(Y) \right], \quad i=1,2,3.$$

$$\otimes g(Y) \equiv g(c) +_{\otimes} \otimes g'(Y) *_{\otimes} (Y -_{\otimes} c), \quad c \in Y \quad , \quad \otimes g'(Y) \equiv \left[\otimes \frac{\partial g_i}{\partial x_i}(Y) \right], \quad i=1,2,3.$$

Функция включения для $G(x, y, z)$ определяется аналогично. Они используются для того, чтобы проверить содержится ли в данном интервале сегмент кривой. Если полученные интервалы функций включения содержат 0 и интервал удовлетворяет ограничениям, т.е. выполняется точность, то это означает, что он содержит кривую. Каждый сегмент кривой в этой подобласти можно приблизить простым интерполяционным сегментом между парой точек решения.

Приведем алгоритм обработки интервалов кривой пересечения.

Имеем входной интервал Y .

помещаем Y в список L

пока L не пустой

выбираем следующую область Y из L

вычисляем функции включения на Y (для проверки содержит ли интервал сегмент кривой)

если весь интервал содержит кривую **то** добавляем Y к решению

иначе если кривая содержится вне интервала **то** отвергаем Y

иначе если интервал достаточно мал **то** добавляем Y к решению

иначе разделяем Y на $Y1$ и $Y2$ и вставляем в L

3. Тест на глобальную параметризацию

Неявно заданную кривую, находящийся на интервале, называют глобально параметризованной по параметру i , если в интервале есть не более одной точки, принадлежащей кривой для каждого параметра i . Для этого нужно вычислить функцию включения $\otimes F(x, y)$ для F' . Чтобы неявно заданная кривая была глобально параметризованных на аппроксимирующей интервале необходимо, чтобы $\otimes F(x, y)$ не содержала нуля на этом интервале. Если она не является глобально параметризованных в интервале для любого параметра, тогда рекурсивно разделяется и проверяется снова.

Глобальная параметризация определяет, каким образом кривая проходит через каждый аппроксимирующих интервал, найденный на предыдущем шаге. Если функция пересекает аппроксимирующий интервал в двух местах, то мы можем просто соединить их без всякой проверки. Поэтому целью глобальной параметризации является разделение аппроксимирующего интервала до тех пор, пока в каждом из новообразованных интервалов кривая не будет проходить однозначно, т.е. пересекать лишь две границы каждого из интервалов. Для этого, как уже было упомянуто выше, интервалы, в которых проходит кривая, должны быть глобально параметризованы одновременно по всем параметрам.

4. Нахождение пересечений кривой с границами аппроксимационных интервалов

На третьем шаге алгоритм находит пересечения неявно заданной кривой с границами каждого интервала. Для этого необходимо вычислить функцию включения $\otimes F(x, y)$ на каждой границе аппроксимирующего интервала. Если полученный интервал содержит 1, то это означает, что кривая пересекает его. Таким образом, для каждого интервала алгоритм на данном шаге находит все решения системы ограничений, образуя множество интервалов, которые называются интервалами граничных пересечений. Интервалы граничных связей распределенные вдоль краев и углов смежных аппроксимирующих интервалов должны совпадать. Таким образом, найдя множество пограничных связей, мы имеем набор интервалов в пределах которых проходит заданная кривая.

После этого шага алгоритм выполняет отбор интервалов. Если интервал Y не содержит ни одного пересечения на границах, он отвергается, поскольку условие глобальной параметризации включает в себя то, что решением не может быть замкнутая кривая, которая полностью содержится в Y . По предположению, решением задачи не может быть сегмент кривой, который не пересекает границ интервала.

5. Уточнения решения задачи

В результате предыдущих шагов полученные граничные сечения является интервалами, в которых содержатся точные координаты кривой пересечения. На этом этапе эти координаты уточняются интервальным методом Ньютона или типа Рунге[3]. Для этого на каждом граничном пересечении, применяя итерационную формулу (1), мы уточняем решение

$$X^{(k+1)} = \{m(X^k) - M^{-1} \cdot f(m(X^k))\} \cap X^k, k \geq 0, \quad M \in f'(X^k) \quad (1)$$

и все действия производятся над интервалами, $m(X^k)$ — точка, которая выбирается из интервала X^k . Обычно эту точку выбирают в середине этого интервала. В процессе уточнения мы предполагаем устойчивым один из параметров задания кривой, а по другому параметру задаем входной интервал, который может быть уточнен данным интервальным методом. Интервальный метод на выходе дает интервал, ширина которого меньше заранее заданного достаточно малого числа ε . Тогда полученные точки на границах ограничивающих интервалов могут быть соединены отрезками или аппроксимированы кривой.

Для уточнения решения на найденных граничных интервалах, вместо метода Ньютона, был использован интервальный метод типа Рунге[3]. Для этого на каждом граничном пересечении, мы уточняем решение с помощью итерационной формулы (1), где M вычисляется следующим образом:

$$M \in \frac{1}{4}f'(m(X^{(k)})) + \frac{3}{4}f'\left(m(X^{(k)}) + \frac{2}{3}(X^{(k)} - m(X^{(k)}))\right), \quad m(X^k) \in X^k \quad (2)$$

Процесс вычислений похож аналогичному в методе Ньютона. В отличие от метода Ньютона, в интервальном методе типа Рунге матрица производных M представляет

линейную комбинацию вида (2). Преимуществами данного метода, по сравнению с интервальным методом Ньютона, есть высший порядок сходимости, что обеспечивает более высокую скорость сходимости итерационного процесса к искомому решению, быстрое отбрасывание интервалов, через которые не проходит кривая решения задачи. Для этих методов общее количество интервальных вычислений не выше, чем у подобных им методах, таких как интервальный метод Ньютона, метод разделения отрезка пополам и другие.

6. Визуализация решений

Метод визуализации, который используется для генерации изображения, основанный на использовании сильно уплотненного дискретного пространства. Вокселей хранятся в 8-арной дереве, позволяющий работать с большим дискретным пространством без использования большого объема памяти компьютера. Все воксели считаются точками и могут быть отображены с помощью программного интерфейса Direct3D или OpenGL. Существует две разновидности процесса визуализации, основанные на дополнительном обходе полного 8-арного дерева с целью построения целочисленных координат существующих листовых элементов и превращение их в координаты точек, находящихся в пределах заданного трехмерного интервала. Полученные координаты определяют слоенные элементы (вокселей), которые могут быть отображены на экране компьютера как точки (point sprites), либо исследованы с целью создания полигональной поверхности с помощью алгоритма "marching cubes" чтобы определить точки прохождения поверхности в пределах данного слоенного элемента.

7. Тестовый пример

Поверхности зададим следующим образом

$$\begin{aligned} F(x, y, z) &= x^2 + y^2 + z^2 - 25 = 0 \\ G(x, y, z) &= y - z - x^2 = 0 \end{aligned}$$

Покажем нахождения вокселей кривой пересечения поверхностей на двух этапах общей схемы алгоритма, а именно: на первом этапе нахождения интервалов "подозреваемых" на наличие границ кривой (Рис.1) и последнем этапе уточнения решения задачи интервальными методами (Рис.2).

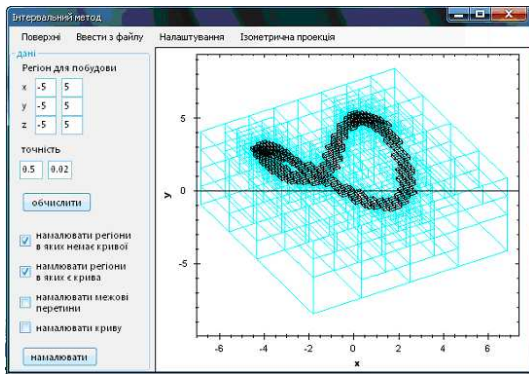


Рис. 1. Множество аппроксимируемых интервалов для кривой пересечения.

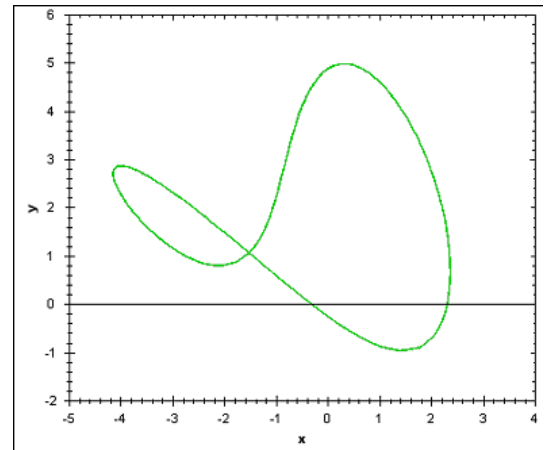


Рис. 2. Уточнения кривой после параметризации и использования интервального метода типа Рунге.

Разработано программное обеспечение для реализации алгоритма пересечения двух поверхностей с использованием уточнения кривой интервальными итерационными методами типа Ньютона и Рунге. Показано эффективность алгоритма решения этой задачи на многих примерах.

Список литературы

- [1] Snyder John M. Interval Analysis for Computer Graphics. Computer Graphics. 1992. — 26, № 2. — Р. 121—129.
- [2] Венгерський П.С., Смушак Н.Я. Інтервальний підхід для розв'язування деяких задач комп'ютерної графіки. Деп. В ДНТБ України. — 1998. — № 352 Ук98. — 27 с.
- [3] Венгерський П.С. Застосування інтервального аналізу для розв'язування деяких класів задач. Автореф. дис. канд. фіз.-мат. наук. — Львів, 1993. — 16 с.