

МЕТОДЫ ПАРАМЕТРИЗАЦИИ И АППРОКСИМАЦИИ ВЕЩЕСТВЕННЫХ ПОВЕРХНОСТЕЙ

Национальный авиационный университет

Представлены избранные методы параметризации и аппроксимации вещественных поверхностей и выпуклых оболочек точечных множеств

Введение

Построение и визуализация поверхностей является важным разделом компьютерной графики и анимации, геометрического моделирования, компьютерно-ориентированного (КАД) проектирования, других как фундаментальных, так и прикладных наук [1–2]. Условимся об используемых понятиях. Среди параметризации поверхностей мы различаем алгебраическую параметризацию, топологическую параметризацию и аналитическую параметризацию. Алгебраической называем параметризацию (если она возможна) алгебраической поверхности [3–4]. Это достаточно разработанный раздел, где предложено и апробировано большое число методов [3–4]. Топологической называем параметризацию, которая основывается на триангуляции поверхности и последующей аппроксимации элементов триангуляции. Топологическая параметризация, которая в [5] называется параметризацией (название, которое конфликтует с термином параметризация для рациональных поверхностей) может быть представлена, следуя [5], такой схемой:

- 1) построение планарной триангуляции;
- 2) построение триангуляции поверхности;
- 3) параметризация триангуляции поверхности.

Аналитической называем параметризацию, которая задается аналитическими (в том числе и неявно заданными) функциями. В данной работе мы используем аналитическую параметризацию.

Имеется ряд связанных понятий аппроксимации поверхности. В одном из

вариантов глобальной аппроксимации по множеству точек строится B -сплайновая поверхность бистепени (p, q) , такая, что расстояние точек B -сплайновой поверхности от аппроксимируемого множества точек поверхности минимально в смысле метода наименьших квадратов [5]. Мы используем понятие численной аппроксимации поверхности, когда точки аналитически заданной параметризованной поверхности численно аппроксимируются значениями аналитических функций, задающих параметризацию поверхности.

Одним из методов визуализации точечных множеств является метод построения выпуклой оболочки точечного множества.

Содержание работы следующее. В первом разделе описывается поверхность Минковского-Кона (М-К) и приводится её параметризация. Даны основные на этой параметризации методы численной аппроксимации поверхности М-К. Эти результаты принадлежат первому из авторов. В последующих разделах, написанных вторым из авторов, приводятся используемые нами методы построения выпуклых оболочек точечных множеств, основанные на работах Грэхема, Джарвиса и книге [2]. Результаты представленного исследования кратко обсуждаются и суммируются в заключении.

Поверхность Минковского-Кона

Поверхностью *Минковского-Кона* (М-К) называем вещественное многообразие размерности два, задаваемое урав-

$$\Delta(p, \sigma) = (\tau + \sigma)(1 + \tau^p)^{1/p} (1 + \sigma^p)^{1/p},$$

которое определено в области

$$\gamma: p > 1, 1 \leq \sigma \leq \sigma_p = (2^p - 1)^{1/p},$$

плоскости $\{p, \sigma\}$, где σ – некоторый вещественный параметр; здесь $\tau = \tau(p, \sigma)$ – функция, однозначно определяемая условиями

$$A^p + B^p = 1, 0 \leq \tau \leq \tau_p, \quad (1)$$

где

$$A = A(p, \sigma) = (1 + \tau^p)^{-1/p} - (1 + \sigma^p)^{-1/p},$$

$$B = B(p, \sigma) = \tau(1 + \tau^p)^{-1/p} + \sigma(1 + \sigma^p)^{-1/p},$$

τ_p определяется уравнением

$$2(1 - \tau_p)^p = 1 + \tau_p^p, 0 < \tau_p < 1. \quad (2)$$

Поверхность Минковского-Кона была неявно введена Г. Минковским в монографии [6] и явно определена в другой параметризации в работе [7]. Мы определяем поверхность Минковского-Кона и её параметризацию [8 – 9], следуя А.В. Малышеву.

Многообразие М-К является поверхностью в пространстве переменных Δ, p, σ . Поверхность Минковского-Кона связана с областями Минковского [6]

$$D_p \subset R^2 = \{x, y\},$$

$$|x|^p + |y|^p < 1, p > 1.$$

Имеет место.

Предложение. Поверхность Минковского-Кона параметризует все допустимые решетки области D_p имеющие по три пары точек на границе этой области.

Области Минковского параметризуются вещественным параметром $p > 1$.

Ввиду неограниченности параметра p речь может идти только о визуализации конечных частей поверхности Минковского-Кона.

Методы аппроксимации поверхности М-К

Пусть часть поверхности М-К задана в компактной области

$$L: p_1 \leq p \leq p_2, \sigma_1 \leq \sigma \leq \sigma_2.$$

Для того, чтобы построить численную аппроксимацию части поверхности М-К в области L в переменных p, σ , нужно решать уравнения (2) и (1), задающие τ_p и τ как неявные функции параметров p и σ . Предварительно, или в процессе аппроксимации части поверхности М-К, строится конечная аппроксимация области L путем её разбиения на полосы, как горизонтальные, так и вертикальные, с последующим подразделением полосы с шагом $\delta\sigma$ (горизонтальная полоса), или δp (вертикальная полоса).

Основной алгоритм.

Шаг 1. Строится аппроксимация $A(L)$ области L .

Шаг 2. Вычисление значения функции $\tau = \tau(p, \sigma)$ и, при необходимости, τ_p , в точках аппроксимации $A(L)$.

Шаг 3. Вычисление значения функции $\Delta(p, \sigma)$ в точках аппроксимации $A(L)$.

В зависимости от используемого метода численного решения уравнений (2) и (1), строится соответствующая аппроксимация.

Для целей последующей визуализации построенной аппроксимации целесообразно строить выпуклые оболочки точечных множеств численной аппроксимации поверхностей.

Метод обхода Грэхема

Грэхем показал, что, выполнив предварительно сортировку точек, крайние точки можно найти за линейное время.

Предположим, что внутренняя точка уже найдена, и она есть началом координат. Упорядочим лексикографически N точек в соответствии со значениями полярного угла и расстояния от начала координат. При выполнении сортировки, естественно, не нужно вычислять действительное расстояние между двумя точками, так как требуется лишь

две величины. Сравнение расстояний необходимо выполнять лишь в случае, если две точки имеют один и тот же полярный угол. Но тогда они лежат на одной прямой с началом координат, и сравнение в этом случае тривиально.

Если точка не является вершиной выпуклой оболочки, то она является внутренней точкой для некоторого треугольника (Opq) , где p и q – последовательные вершины выпуклой оболочки. Суть метода Грэхема состоит в однократном просмотре упорядоченной последовательности точек, в процессе которого удаляются внутренние точки. Оставшиеся точки являются вершинами выпуклой оболочки, представленными в требуемом порядке. Просмотр начинается с точки, помеченной как НАЧАЛО, в качестве которой можно взять самую правую с наименьшей ординатой точку из данного множества, заведомо являющуюся вершиной выпуклой оболочки. Тройки последовательных точек многократно проверяются в порядке обхода против часовой стрелки с целью определить, образуют или нет, они угол, больший или равный π . Если внутренний угол больше или равен π , то говорят, что $p_1p_2p_3$ образуют «правый поворот» ($\Delta > 0$), иначе они образуют «левый поворот» ($\Delta < 0$). Это можно легко определить, воспользовавшись формулой

$$x = \begin{vmatrix} x & y & 1 \\ x' & y' & 1 \\ x'' & y'' & 1 \end{vmatrix}$$

Из выпуклости многоугольника непосредственно следует, что при его обходе будут делаться только левые повороты. Если $p_1p_2p_3$ образуют правый поворот, то p_2 не может быть крайней точкой, так как она является внутренней для треугольника $(Op_1 p_3)$. В зависимости от результата проверки угла, образуемого текущей тройкой точек, возможны два варианта продолжения просмотра (рис. 1):

1. $p_1p_2p_3$ образуют правый поворот. Удалить вершину p_2 и проверить тройку $p_0p_1p_3$.

2. $p_1p_2p_3$ образуют левый поворот. Продолжить просмотр, перейдя к проверке тройки $p_2p_3p_4$.

Просмотр завершается, когда, обойдя все вершины, вновь приходим в вершину НАЧАЛО.

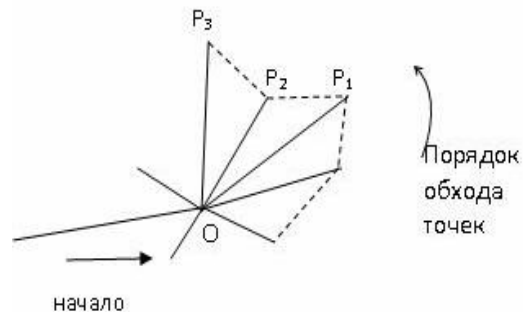


Рис. 1. Начало обхода точек в методе Грэхема. Вершина p_2 удаляется, если угол $p_1p_2p_3$ оказывается вогнутым

Ниже дано более точное описание алгоритма Грэхема. В алгоритме S – это исходное множество из N точек на плоскости.

procedure GREMHULL(S)

1. Найти внутреннюю точку q .
 2. Используя q как начало координат, упорядочить точки множества S лексикографически в соответствии с полярным углом и расстоянием от q . Организовать точки множества в виде кольцевого дважды связанного списка со ссылками СЛЕД и ПРЕД и указателем НАЧАЛО на первую вершину. Значение *true* логической переменной f указывает на то, что вершина НАЧАЛО оказалась достигнутой при прямом продвижении по оболочке, а не в результате возврата.

3. (Обход)

```

begin  $v := \text{НАЧАЛО}; \omega := \text{ПРЕД}[v];$ 
 $f := \text{false};$ 
while ( $\text{СЛЕД}[v] \neq \text{НАЧАЛО}$  or
 $f = \text{false}$ ) do
  begin if  $\text{СЛЕД}[v] = \omega$  then
 $f := \text{true};$ 

```

```

if (три точки  $v$ , СЛЕД[ $v$ ]
    [СЛЕД[ $v$ ]];
    образуют левый поворот) then
 $v :=$ СЛЕД[ $v$ ];
    else begin УДАЛИТЬ СЛЕД[ $v$ ];
 $v :=$ ПРЕД[ $v$ ]
        end
end
end.

```

По окончании выполнения алгоритма список содержит упорядоченные нужным образом вершины оболочки.

Теорема 1. Выпуклая оболочка N точек на плоскости может быть найдена за время $O(N \log N)$ при памяти $O(N)$ с использованием только арифметических операций и сравнений.

Несмотря на то, что алгоритм Грэхема является оптимальным, по-прежнему имеется много причин для продолжения исследования задачи о выпуклой оболочке.

1. Рассмотренный алгоритм является оптимальным в худшем случае.

2. Алгоритм не имеет обобщения на случай пространств более высокой размерности.

3. Алгоритм не является открытым алгоритмом, так как все точки множества должны быть известны до начала работы алгоритма.

4. При возможности параллельной обработки более предпочтительным является рекурсивный алгоритм, допускающий разбиение исходной задачи и данных на меньшие подзадачи.

Обход методом Джарвиса

Многоугольник с одинаковым успехом можно задать упорядоченным множеством, как его вершин, так и его ребер. В задаче о выпуклой оболочке обращали внимание главным образом на изолированные крайние точки. Определим ребра выпуклой оболочки. Если задано множество точек, то довольно трудно быстро определить, является или нет некоторая точка крайней. Однако если даны две точки, то непосредственно можно проверить, является или нет со-

единяющий их отрезок ребром выпуклой оболочки.

Теорема 2. Отрезок l , определяемый двумя точками, является ребром выпуклой оболочки тогда и только тогда, когда все другие точки заданного множества лежат на l или с одной стороны от него.

Предположим, наименьшая в лексикографическом порядке точка p_1 заданного множества точек. Эта точка заведомо является вершиной оболочки, и теперь хотелось бы найти следующую за ней вершину p_2 выпуклой оболочки. Точка p_2 – это точка, имеющая наименьший положительный полярный угол относительно точки p_1 как начала координат.

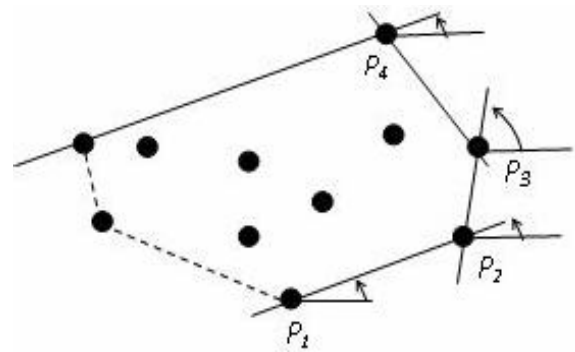


Рис. 2. Построение выпуклой оболочки методом Джарвиса

Алгоритм Джарвиса находит последовательные вершины оболочки путем многократного вычисления угла поворота. Каждая новая вершина определяется за время $O(N)$.

Аналогично следующая точка p_3 имеет наименьший полярный угол относительно точки p_2 как начала координат, и каждая последующая точка выпуклой оболочки может быть найдена за линейное время. Алгоритм Джарвиса обходит кругом выпуклую оболочку, порождая в нужном порядке последовательность крайних точек, по одной точке на каждом шаге (рис. 2). Таким образом строится часть выпуклой оболочки (ломаная линия) от наименьшей в лексикографическом порядке точки (p_1 на рис. 2) до наибольшей в лексикографическом порядке точки (p_4 на том же рисунке). Построение выпуклой

завершается нахождением другой ломаной, идущей из наибольшей в лексикографическом порядке точки в наименьшую в лексикографическом порядке точку. Ввиду симметричности этих двух этапов необходимо изменить на противоположные направления осей координат и иметь дело теперь с полярными углами, наименьшими относительно отрицательного направления оси x .

Так как все N точек множества могут лежать на его выпуклой оболочке (быть ее вершинами), а алгоритм Джарвиса затрачивает на нахождение каждой точки оболочки линейное время, то время выполнения алгоритма в худшем случае равно $O(N^2)$, что хуже, чем у алгоритма Грэхема. Если в действительности число вершин выпуклой оболочки равно h , то время выполнения алгоритма Джарвиса будет $O(hN)$, и он очень эффективен, когда заранее известно, что значение h мало.

Быстрые методы построения выпуклой оболочки

FASTHULL – метод разбивает множество S из N точек на два подмножества, каждое из которых будет содержать одну из двух ломаных, соединение которых дает многоугольник выпуклой оболочки. Начальное разбиение множества определяется прямой, проходящей через две точки l и r , имеющие соответственно наименьшую и наибольшую абсциссы (рис 3).

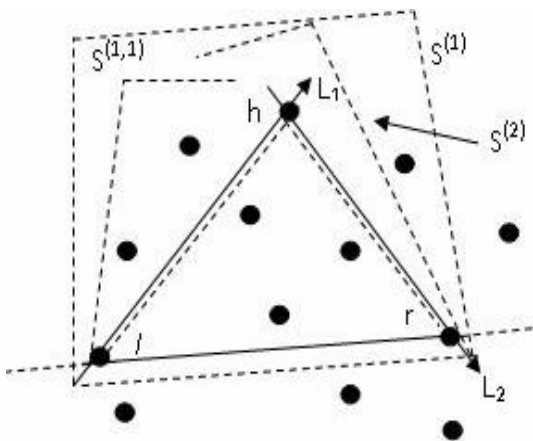


Рис. 3. Точки l , r и h определяют разбиение множества $S^{(1)}$

Обозначим через $S^{(1)}$ подмножество точек, расположенных выше или на прямой, проходящей через точки l и r , а через $S^{(2)}$ симметричным образом определенное подмножество точек, расположенных ниже или на той же самой прямой. На каждом последующем шаге обработка множеств, подобных $S^{(1)}$ и $S^{(2)}$, выполняется следующим образом (для конкретности мы рассмотрим множество $S^{(1)}$ на рис. 3).

Определим точку h , для которой треугольник (h, l, r) имеет максимальную площадь среди всех треугольников, а если таких точек имеется более одной, то выбираем ту из них, у которой угол (l, r, h) больше. Заметим, что точка h гарантированно принадлежит выпуклой оболочке.

Затем строятся две прямые: одна L_1 направленная из l в h , другая L_2 – из h в r . Для каждой точки множества $S^{(1)}$ определяется ее положение относительно этих прямых. Ясно, что ни одна из точек не находится одновременно слева как от L_1 , так и от L_2 , кроме того, все точки, расположенные справа от обеих прямых, являются внутренними точками треугольника (l, r, h) и поэтому могут быть удалены из дальнейшей обработки. Точки, расположенные слева от L_1 или на ней (и расположенные справа от L_2), образуют множество $S^{(1,1)}$, аналогично образуется множество $S^{(1,2)}$. Вновь образованные множества $S^{(1,1)}$ и $S^{(1,2)}$, передаются на следующий уровень рекурсивной обработки.

Возьмем соответствующий выбор начальных значений $\{l_0, r_0\}$ точек l и r может быть выполнен следующим образом. Выберем в качестве l точку (x_0, y_0) , имеющую наименьшую абсциссу, в качестве r_0 возьмем точку $(x_0, y_0 - \varepsilon)$, где ε – произвольное малое положительное число. Это приводит к тому, что в качестве исходной прямой, разбивающей множество на части, выбирается вертикальная прямая, проходящая через l_0 . После завершения алгоритма точка r_0 –

ляется (положив $\varepsilon = 0$, делаем точки l_0 и r_0 совпадающими).

Предположим, что множество S содержит по крайней мере две точки, а функция $MOSTFARPOINT(S; l, r)$ вычисляет точку h , которая содержится в S , писаным выше способом; кроме того, $FASTHULL$ возвращает в качестве результата упорядоченный список точек, а «*» обозначает операцию сцепления (конкатенации) списков.

function $FASTHULL(S; l, r)$

begin if $S = \{l, r\}$ **then return** (l, r)

(* выпуклая оболочка состоит из единственного ориентированного ребра *)

else begin $h := MOSTFARPOINT(S; l, r)$;

$S^{(1)} :=$ точки множеств S , расположенные слева от прямой lh или на ней;

$S^{(2)} :=$ точки множества S , расположенные слева от прямой hr или на ней;

return $FASTHULL(S^{(1)}; l, r) * (FASTHULL(S^{(2)}; h, r) - h)$

end

end

Таким образом, если уже имеется функция $FASTHULL$, то поставленную задачу можно решить с помощью следующей простой программы:

begin $l_0 = (x_0, y_0) :=$ точка множества S с наименьшей абсциссой;

$r_0 := (x_0, y_0 - \varepsilon); FASTHULL(S; l_0, r_0)$

удалить точку r_0 (*это эквивалентно тому, чтобы положить $\varepsilon = 0$ *)

end.

Для того чтобы выделить из множества S подмножества $S^{(1)}$ и $S^{(2)}$ требуется $O(N)$ операций. Затем следует рекурсивное обращение к функции для обработки $S^{(1)}$ и $S^{(2)}$. Теперь, если мощность каждого из этих множеств не превосходит мощности множества S , умноженной на некоторую константу, меньшую 1, и это условие имеет место на каждом уровне рекурсии, то время выполнения алгоритма равно $O(N \log N)$.

Выводы

Предложен класс методов численной аппроксимации и последующей визуализации неявно заданных поверхностей. Для сглаживания аппроксимации предложено использовать методы построения выпуклых множеств. Приведены методы построения выпуклых оболочек точечных множеств.

Список литературы

1. *Farin, G.* Curves and Surfaces for Computer Aided Geometric Design, Academic Press, San Diego. – 1988. – 682 p.

2. *Препарата Ф., Шеймос М.* Вычислительная геометрия: Введение. – М.: Мир, 1989. – 478 с.

3. *Шафаревич И.П.* Основы алгебраической геометрии. – .1. – Т2. – М.: Наука, 1988. – С. 304–351.

4. *Schicho, J.* Rational Parametrization of Surfaces // J. Symbolic Computation. 1998. – 26. – P. 1–29.

5. *Floater M.S.* Parameterization and smooth approximation of surface triangulation // Computer Aided Geometric Design. – 1997. – 14. – P. 231–250.

6. *Minkowski H.* Diophantische Approximationen. – Leipzig: Teubner, 1907, Vol. 8. – 235 p.

7. *Cohn H.* Minkowski's conjectures on critical lattices in the metric $(|\xi|^p + |\eta|^p)^{1/p}$ // Annals of Math. – 1950. – 51. – N 3. – P. 738–734.

8. *Glazunov N.M.* Minkowski's Conjecture on Critical Lattices and the Quantifier Elimination // Algorithmic algebra and logic. Proc. of the A3L 2005, Passau, Germany. – P. 111–114.

9. *Glazunov N.M.* Interval computations and their Categorification. Numerical Algorithms. – 2004. – 37. – P. 159–164.

10. *Stoer J., Witzgall C.* Complexity and optimization in finite dimensions I. New-York: Springer-Verlag, 1970. – 324 p.