



УДК 517.518.85+517.27

ББК 22.161.6

МЕТОД ЦЕПЕЙ ДЛЯ ОРГАНИЗАЦИИ ХРАНЕНИЯ МНОГОМЕРНЫХ ТРИАНГУЛЯЦИЙ¹

Клячин Владимир Александрович

Доктор физико-математических наук,
заведующий кафедрой компьютерных наук и экспериментальной математики
Волгоградского государственного университета
klchnv@mail.ru
Проспект Университетский, 100, 400062 г. Волгоград, Российская Федерация

Попов Владимир Валентинович

Кандидат физико-математических наук,
доцент кафедры компьютерных наук и экспериментальной математики
Волгоградского государственного университета
porov_v_v@gambler.ru
Проспект Университетский, 100, 400062 г. Волгоград, Российская Федерация

Аннотация. Рассматривается задача о поиске наилучшего способа записи триангуляции конечного набора точек многомерного пространства. В работе дана формальная постановка задачи. Приведены примеры для сравнения различных способов представления триангуляций. Так же в статье приводится алгоритм, перебирающий все триангуляции произвольного конечного набора точек на плоскости.

Ключевые слова: триангуляция, симплекс, оценка объема памяти, число триангуляций, число Каталана.

О способах записи триангуляции

В настоящей статье под k -мерным симплексом S в \mathbb{R}^n мы понимаем выпуклую оболочку $k+1$ точек $A_i, i = 0, \dots, k \leq n$ таких, что векторы $A_1 - A_0, A_2 - A_0, \dots, A_k - A_0$ линейно независимы. Пусть $p_i, i = 1, \dots, N$ — некоторый набор точек $p_i \in \mathbb{R}^n$ таких, что любой симплекс в вершинах из $\{p_i\}$ является невырожденным. Триангуляцией T заданного набора точек называется такой набор n -мерных симплексов S_1, \dots, S_m , что:

1) каждая точка p_i заданного набора является вершиной одного из симплексов $S \in T$;

2) каждая вершина любого симплекса $S \in T$ является одной из точек $p_i, i = 1, \dots, N$;

3) внутренность пересечения любых двух симплексов пуста.

Один из первых алгоритмов триангуляции с использованием условия пустого шара был предложен Б.Н. Делоне в его работе [5] (перевод см.: [1]).

Нас будет интересовать вопрос об оценке объема памяти, необходимой для хранения триангуляции заданного набора точек. Отметим работу [4], в которой решается подобная задача для случая размерности $n = 2$. Объем памяти будем вычислять в битах. С этой целью введем функцию $b(x)$, равную минимальному числу бит, необходимому для записи натурального числа $x \in \mathbb{N}$ в двоичной системе счисления. Очевидно, что

$$b(x) = \lceil \log_2 x \rceil + 1.$$

Формально постановка задачи следующая. Обозначим через $V(P)$ совокупность всех триангуляций набора точек $P = \{p_1, p_2, \dots, p_N\}$. Тогда под способом записи триангуляции T мы понимаем отображение

$$f : V(P) \rightarrow \mathbb{N},$$

которое всякой триангуляции ставит в соответствие натуральное число. Причем, мы предполагаем, что отображение обладает свойством

$$\forall P' \neq P'' \in V(P) \Rightarrow f(P') \neq f(P'').$$

Тогда величину

$$\mu(f) = \max_{T \in V(P)} b(f(T))$$

можно считать мерой объема памяти, необходимой для запоминания триангуляции T при заданном способе хранения f . Приведем примеры.

Пример 1. Один из стандартных способов (см., например, [2]) хранения триангуляции состоит в том, что каждому симплексу $S_i \in T$ сопоставляется набор из $(n + 1)$ натурального числа $n_{i0}, n_{i1}, \dots, n_{in}$, такой, что

$$S_i = \text{conv}\{p_{n_{i0}}, p_{n_{i1}}, \dots, p_{n_{in}}\}.$$

Значит, для такого способа записи одного симплекса понадобится $(n + 1)b(N)$ бит, а для всей триангуляции получим величину $m(n + 1)b(N)$. В этом примере конструкция отображения f следующая. Значение $f(T)$ — это натуральное число, в двоичной системе записи которого первые $b(N)$ бит занимает число n_{10} , вторые $b(N)$ бит — n_{11} и т. д. Другими словами, матрицу натуральных чисел $\|n_{ij}\|, i = 1, \dots, N, j = 0, \dots, n$ записываем в итоговое число построчно. Полученное натуральное число $f(T)$ будет в памяти занимать $m(n + 1)b(N)$ бит. Обозначим построенное отображение в этом примере через f_0 . Заметим, что

$$\lim_{N \rightarrow \infty} \frac{\mu(f_0)}{m \log_2 N} = n + 1.$$

Мы предлагаем конструкцию построения отображения f , для которого

$$\lim_{N \rightarrow \infty} \frac{\mu(f)}{m \log_2 N} = 1. \tag{1}$$

Для ее описания нам понадобится понятие цепи. Цепью C в заданной триангуляции $T \in V(P)$ мы назовем совокупность различных симплексов S_1, S_2, \dots, S_k , такую, что для любого $i = 2, \dots, k$ симплексы S_{i-1} и S_i имеют единственную общую $(n - 1)$ -мерную грань. Число k называется длиной цепи C . Предположим, что триангуляция $T \in V(P)$ разбита на непересекающиеся цепи C_1, \dots, C_p длины l_1, \dots, l_p соответственно. Пусть C — некоторая цепь. Построим последовательность натуральных чисел

$$n_{10}, n_{11}, \dots, n_{1n}, x_1, n_2, x_2, n_3, \dots, x_{k-1}, n_k, x_k, \tag{2}$$

где $n_{10}, n_{11}, \dots, n_{1n}$ — номера вершин в диапазоне $1, 2, \dots, N$ первого симплекса цепи, x_1 — номер в диапазоне $0, 1, \dots, n$ вершины первого симплекса, напротив которого расположена общая грань со вторым симплексом, n_2 — номер вершины второго симплекса, расположенной напротив общей грани с первым симплексом, и т. д. Причем x_k всегда равно $n + 1$. Это число нам необходимо для указания конца цепи в последующих построениях. Несложно подсчитать, что для записи всей этой последовательности чисел понадобится $(n + k)b(N) + kb(n + 1)$ бит. Так что для каждой цепи C_i мы получим запись длины $(n + l_i)b(N) + l_ib(n + 1)$. Суммируя по всем цепям и используя равенство

$$\sum_{i=1}^p l_i = m,$$

получим, что для всей записи симплексов триангуляции $T \in V(P)$ будет затрачено $(pn + m)b(N) + mb(n + 1)$ бит. Конструкция отображения f сводится к упаковке последовательностей вида (2) в одно двоичное число, содержащее ровно $(pn + m)b(N) + mb(n + 1)$ бит. Следовательно, для нашего отображения f получаем

$$\mu(f) = (pn + m)b(N) + mb(n + 1).$$

Поэтому

$$\lim_{N \rightarrow \infty} \frac{\mu(f)}{m \log_2 N} = \lim_{N \rightarrow \infty} \frac{(pn + m)b(N) + mb(n + 1)}{m \log_2 N} = 1,$$

при условии $p = o(m)$, $N \rightarrow \infty$, что предполагает неограниченное и равномерное увеличение длины цепей при $N \rightarrow \infty$. На самом деле это условие несущественно, поскольку небольшие перестройки триангуляции позволяют записать ее в виде одной цепи. Однако представление триангуляции в виде одной цепи не разумно с точки зрения временных затрат на доступ к отдельному симплексу. Действительно, предложенная нами конструкция, скажем, в отличие от отображения f_0 примера 1, не позволяет осуществить прямой доступ к любому симплексу, поскольку номера вершин формируются динамически — в процессе побитового чтения числа $f(T)$. При этом, чем длиннее цепь, тем она более затратней для чтения. Попытаемся найти оптимальное значение для количества цепей в триангуляции. Для этого составим целевую функцию. Пусть c_1 — стоимость единицы одной операции доступа к памяти, c_2 — стоимость одного бита памяти. Средняя длина цепи равна m/p , соответственно среднее число операций чтения равно $m/2p$. Получаем следующее выражение для целевой функции

$$F(p) = c_1 \frac{m}{2p} + c_2(pn + m)b(N) + mb(n + 1).$$

Исследуя эту функцию на минимум, получим оптимальное значение для числа цепей

$$p^* = \sqrt{\frac{c_1}{c_2} \frac{m}{2nb(N)}}.$$

В этом случае

$$\mu(f) = \sqrt{\frac{c_1}{c_2} mnb(N)} + mb(N) + mb(n+1).$$

Так что и в оптимальном случае

$$\lim_{N \rightarrow \infty} \frac{\mu(f)}{m \log_2 N} = \lim_{N \rightarrow \infty} \frac{\sqrt{\frac{c_1}{c_2} mnb(N)} + mb(N) + mb(n+1)}{m \log_2 N} = 1.$$

Зададимся вопросом, насколько предложенный метод является наилучшим способом записи триангуляции. С этой целью необходимо получить нижние оценки величины $\mu(f)$, независимые от f . Пусть $K = |V(P)|$ обозначает мощность множества $V(P)$, то есть равно количеству триангуляций набора точек P . Несложно проверить, что

$$\mu(f) \geq \lceil \log_2 K \rceil + 1.$$

Действительно, поскольку отображение f взаимно-однозначно, то мощность множества значений f равно мощности множества области определения, то есть в нашем случае эта мощность равна K . Это требует как минимум K натуральных чисел для идентификации каждой триангуляции. Для записи такого числа необходимо как минимум $\lceil \log_2 K \rceil + 1$ бит.

Пример 2. Рассмотрим набор точек в \mathbb{R}^2 с координатами $p_{i0} = (i, 0)$ и $p_{i1}(i, 1)$, где $i = 0, 1, \dots, n$. Так что $N = 2(n+1)$. Триангуляцию будем строить так. Возьмем ребро $p_{00}p_{01}$. Для получения треугольника у нас два варианта: либо точка p_{10} , либо точка p_{11} . Выберем какую-нибудь из них. Допустим, это точка p_{10} . Для получения следующего треугольника у нас опять два варианта: либо точка p_{20} , либо точка p_{21} . Заметим, что во всех вариантах у нас будут две точки: одна на верхнем ряду, другая — на нижнем. Теперь отображение f можно построить так. Каждой триангуляции необходимо поставить в соответствие двоичное число длины $2n$ бит, в котором число единиц равно числу нулей. Единицы соответствуют выбору точки из верхнего ряда, а нули — из нижнего ряда. Так что всего комбинаций получается C_{2n}^n . Согласно формуле Стирлинга $C_{2n}^n = o(4^n)$. Отсюда получаем оценку для числа триангуляций $K = o(4^n)$, что, в свою очередь, учитывая, что число треугольников $m = 2n$, дает нулевое значение в пределе вида (1).

Пример 3. Рассмотрим $N = n + 2$ точки, лежащие в вершинах выпуклого N -угольника. Число триангуляций вершин выпуклого многоугольника дается числом Каталана (см., например, [7])

$$C_n = \frac{1}{n+1} C_{2n}^n.$$

Так что в этом случае $\mu(f) = o(4^n)$ и мы получаем, как и в предыдущем примере,

$$\lim_{N \rightarrow \infty} \frac{\mu(f)}{m \log_2 N} = 0. \quad (3)$$

Оказывается, что эти примеры носят более общий характер. Именно в работах [3; 6; 8–11] было показано, что найдется абсолютная постоянная c , независимая от набора точек и его мощности N , что число триангуляций не превосходит c^N . Наилучшая

на данный момент константа $c = 43$ была найдена в работе [10]. Следовательно, можно надеяться на существование такого способа записи триангуляции, при котором число бит растёт линейно при $N \rightarrow \infty$. Ну и, в частности, будет выполнено (3).

Триангуляции многоугольника на плоскости

Пусть F — многоугольник на плоскости и $S = \{p_1, p_2, \dots, p_N\}$ — такой набор точек плоскости, который включает все вершины многоугольника F , причем каждая точка p_i является или вершиной, или внутренней точкой F . Необходимо перечислить все триангуляции многоугольника F , вершинами которой являются точки множества S . Не теряя общности, считаем, что отрезок $p_1 p_2$ является стороной многоугольника F .

Обозначения:

Триангуляция T будет состоять из треугольников $\Delta_1, \Delta_2, \dots, \Delta_m$ (m — общее число треугольников в триангуляции).

Вершинами треугольника Δ_k служат точки множества S с порядковыми номерами $n1[k], n2[k]$ и $n3[k]$.

k — порядковый номер строящегося треугольника из триангуляции T . В начале построения $k = 1$.

Треугольник строим следующим образом: к отрезку с порядковыми номерами точек $n1[k], n2[k]$ («активная сторона строящегося треугольника Δ_k ») пытаемся подобрать вершину с порядковым номером $n3[k]$ (если такая вершина существует) из того же множества S , и полученный треугольник Δ_k будет включен в строящуюся триангуляцию T . После этого ребро с концами $n1[k], n3[k]$ будет принято за активную сторону треугольника Δ_{k+1} , а ребро с концами $n2[k], n3[k]$ («свободное ребро») будет запомнено и в дальнейшем может быть использовано как активное ребро для построения треугольника Δ_i при некотором $i > k + 1$ (если в этом возникнет необходимость).

Триангуляции многоугольника на плоскости

Пусть F — замкнутый многоугольник на плоскости и $S = \{p_1, p_2, \dots, p_n\}$ — такое его конечное подмножество, которое содержит все вершины F . Необходимо перечислить все триангуляции многоугольника F , вершинами которых являются точки множества S . Не теряя общности, считаем, что отрезок $[p_1, p_2]$ является стороной многоугольника F .

Обозначения:

Очередная триангуляция T будет состоять из треугольников $\Delta_1, \Delta_2, \dots, \Delta_m$, где m — общее число треугольников в триангуляции (оно одинаково для любой триангуляции). Это число будет найдено после построения первой триангуляции. В начале работы алгоритма полагаем $m = 0$. Вершинами треугольника Δ_k служат точки множества S с порядковыми номерами $n_1[k], n_2[k]$ и $n_3[k]$.

k — порядковый номер строящегося треугольника из триангуляции T . В начале построения $k = 1$.

Через $D(i_1, i_2, i_3)$ будем обозначать треугольник с номерами вершин i_1, i_2, i_3 . Пусть $k > 1$ и треугольники $\Delta_1, \Delta_2, \dots, \Delta_{k-1}$ уже построены. Пусть i_1, i_2, i_3 — целые числа от 1 до n . Тогда через $f(i_1, i_2, i_3)$ будем обозначать такое наименьшее целое число j , что $i_3 < j \leq n$ и треугольник $D(i_1, i_2, j)$ можно добавить к последовательности $\Delta_1, \Delta_2, \dots, \Delta_{k-1}$, то есть он удовлетворяет следующим условиям:

(а) $D(i_1, i_2, j)$ не содержит точек множества S , отличных от вершин этого треугольника;

(b) треугольник $D(i_1, i_2, j)$ не имеет общих внутренних точек ни с одним из треугольников $\Delta_1, \Delta_2, \dots, \Delta_{k-1}$ (но может иметь общую вершину или общую сторону).

Если числа j с перечисленными свойствами не существует, то полагаем $f(i_1, i_2, i_3) = 0$.

Определим функцию $f(i_1, i_2, i_3)$ и при $k = 1$. А именно, $f(i_1, i_2, i_3)$ — это наименьшее целое число от 1 до n , которое больше i_3 , и для которого треугольник $D(i_1, i_2, j)$ удовлетворяет условию (a). Если такого j не существует, то полагаем $f(i_1, i_2, i_3) = 0$.

В процессе работы алгоритма будут строиться также три одномерных массива $q_1[i]$, $q_2[i]$ и $fr[i]$, где $1 \leq i < n$. При построении треугольника Δ_k отрезок e_i , концами которого являются точки множества S с порядковыми номерами $q_1[i]$ и $q_2[i]$, $i = 1, 2, \dots, k-1$, в дальнейшем может быть использован для построения треугольника Δ_t , где $t \geq k$. Отрезки e_i содержатся в объединении границ треугольников $\Delta_1, \Delta_2, \dots, \Delta_{k-1}$ (e_i — «граничные отрезки»). Массив $fr[i]$ играет следующую роль: если $fr[i] = 0$, то отрезок e_i пока не использован для построения треугольника. Если же этот отрезок будет стороной некоторого треугольника Δ_t , $t > 0$, включенного в строящуюся триангуляцию, то $fr[i] = t$.

На множестве упорядоченных пар целых чисел рассматриваем обычный лексикографический порядок: если a, b, c, d — целые числа, то

$$(a, b) < (c, d) \iff a < c \text{ или } (a = c \text{ и } b < d).$$

Алгоритм построения триангуляций

(1) Полагаем $k = 1$, $n_1[1] = 1$, $n_2[1] = 2$ (активная сторона треугольника Δ_1).

Триангуляция T пока не содержит ни одного треугольника.

Полагаем $m = 0$ (число треугольников в триангуляции; оно будет определено после того, как будет построена первая триангуляция).

Полагаем $ktr = 0$ — общее число триангуляций.

(2) Начинаем строить треугольник Δ_k (в этом пункте $k \geq 1$ — целое число). Полагаем $n_3[k] = 0$.

(3) Полагаем $j = f(n_1[k], n_2[k], n_3[k])$.

Если $j = 0$, то идем к пункту (5), если же $j > 0$, то идем к пункту (4).

(4) Принимаем за Δ_k треугольник $D(n_1[k], n_2[k], j)$, где j — число, найденное в пункте (3), и определяем новую активную сторону с номерами вершин $n_1[k]$ и $n_3[k]$. Для этого полагаем

$$\begin{aligned} n_3[k] &= j, \\ n_1[k+1] &= n_1[k], \\ n_2[k+1] &= n_3[k]. \end{aligned}$$

Сторону с номерами вершин $n_2[k]$ и $n_3[k]$ запоминаем как граничную сторону. Для этого полагаем

$$\begin{aligned} q_1[k] &= n_2[k], \\ q_2[k] &= n_3[k], \\ fr[k] &= 0. \end{aligned}$$

Число k увеличиваем на 1.

Если $m = 0$, то идем к пункту (2).

Если $m > 0$ и $k \leq m$, то идем к пункту (2).

Если $m > 0$ и $k > m$, то идем к пункту (7) (очередная триангуляция построена).

(5) На этот пункт попадем, если $j = 0$ в пункте (3). Среди граничных сторон выбираем такую, которую можно принять за активную сторону. Для этого полагаем

$$E = \{l : f(q_1[l], q_2[l], 0) > 0, \text{ где } 1 \leq l < k\}.$$

Если множество E пусто, то идем к пункту (6). Если же $E \neq \emptyset$, то за l_0 принимаем такое число, что $(q_1[l_0], q_2[l_0])$ — наименьший элемент множества E (относительно введенного выше лексикографического порядка). После выбора l_0 полагаем

$$\begin{aligned} n_1[k] &= q_1[l_0], \\ n_2[k] &= q_2[l_0], \\ n_3[k] &= 0, \\ fr[l_0] &= k. \end{aligned}$$

Идем к пункту (3).

(6) На этот пункт попадем, если множество E в пункте (5) пусто, то есть уже нет граничных ребер, которые можно принять за сторону строящегося треугольника. Это будет означать, что или построена очередная триангуляция, или зашли в тупик, или алгоритм завершил работу и все триангуляции уже перечислены. Действия в этом пункте таковы:

Если $m = 0$, то полагаем $m = k - 1$ и идем к пункту (7).

Если $k = 0$, то выдаем число ktr найденных триангуляций и завершаем работу.

Если $1 < k \leq m$, то k уменьшаем на 1 и идем к пункту (8).

(7) Построена очередная триангуляция. Выдаем список треугольников $\Delta_1, \Delta_2, \dots, \Delta_m$. Увеличиваем число ktr (уже построенных триангуляций) на 1.

Переходим к построению очередной триангуляции. Для этого полагаем $k = m - 1$: в дальнейшем будем строить заново треугольник Δ_{m-1} . Тем самым мы удаляем два последних построенных треугольника Δ_{m-1} и Δ_m (удаление только последнего треугольника не даст новой триангуляции).

(8) Начинаем заново строить треугольник Δ_k , а затем и все треугольники с большими номерами. При построении «старых» треугольников Δ_k, Δ_{k+1} и т. д. могли быть использованы граничные ребра $q_1[l], q_2[l]$. Теперь эти ребра снова должны стать граничными. Это достигается так: просматриваем массивы $q_1[l], q_2[l]$, и если при некотором l верно $1 \leq l < k$ и $fr[l] \geq k$, то полагаем $fr[l] = 0$ — теперь при построении треугольника Δ_t , где $t \geq k$, сторону с номерами вершин $q_1[l], q_2[l]$ можно будет использовать при построении треугольников Δ_t , где $t \geq k$.

Полагаем $j = f(n_1[k], n_2[k], n_3[k])$.

Если $j > 0$, то идем на пункт (3).

(В противном случае) k уменьшаем на 1.

Если $k > 1$, то идем на пункт (8).

Полагаем $j = f(n_1[k], n_2[k], n_3[k])$.

Если $j > 0$, то идем на пункт (8).

(В противном случае) выдаем общее число ktr найденных триангуляций и завершаем работу.

ПРИМЕЧАНИЕ

¹ Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований, грант № 13-01-97034 р_поволжье_а.

СПИСОК ЛИТЕРАТУРЫ

1. Делоне, Б. Н. О пустой сфере. К мемуару Георгия Вороного / Б. Н. Делоне ; пер. с фр.: А. Ю. Игумнов // Записки семинара «Сверхмедленные процессы». — 2005. — Вып. 1. — С. 147–153.
2. Скворцов, А. В. Алгоритмы построения и анализа триангуляции / А. В. Скворцов, Н. С. Мирза. — Томск : Изд-во Томск. ун-та, 2006. — 168 с.
3. Ajtai, M. Crossing-free subgraphs / M. Ajtai, V. Chvátal, M. M. Newborn, E. Szemerédi // *Annals Discrete Math.* — 1982. — № 12. — P. 9–12.
4. De Floriani, L. Compressing Triangulated Irregular Networks / L. De Floriani, P. Magillo, E. Puppo // *Geoinformatica.* — 2000. — V. 1, № 4. — P. 67–88.
5. Delaunay, B. N. Sur la sphere vide. A la memoire de Georges Voronoi / B. N. Delaunay // *Изв. АН СССР.* — 1934. — № 6. — С. 793–800.
6. Denny, M. O. Encoding a triangulation as a permutation of its point set / M. O. Denny, C. A. Sohler // *Proc. 9th Canadian Conf. on Computational Geometry.* — 1997. — P. 39–43.
7. Pólya, G. On picture-writing / G. Pólya // *The American Mathematical Monthly.* — 1956. — № 63. — P. 689–697.
8. Santos, F. A better upper bound on the number of triangulations of a planar point set / F. Santos, R. Seidel // *J. Combinat. Theory, Ser. A.* — 2003. — № 102. — P. 186–193.
9. Seidel, R. On the number of triangulations of planar point sets / R. Seidel // *Combinatorica.* — 1998. — № 18. — P. 297–299.
10. Sharir, M. Random triangulations of planar point sets / M. Sharir, E. Welzl // *Proc. 22nd Ann. ACM Symp. on Computational Geometry.* — 2006. — P. 273–281.
11. Smith, W. S. *Studies in Computational Geometry Motivated by Mesh Generation: Ph.D. Thesis* / W. S. Smith. — Princeton University, 1989. — 488 p.

REFERENCES

1. Delaunay B.N. O pustoy sfere. K memuaru Georgiya Voronogo [On an empty sphere. To memoir of Georgiy Voronoy]. *Zapiski seminara «Sverkhmedlennye protsessy»* [Notes of Subslow Processes Seminar], 2005, issue 1, pp. 147–153.
2. Skvortsov A.V., Mirza N.S. *Algoritmy postroeniya i analiza triangulyatsii* [Algorithms for construction and analysis of triangulations]. Tomsk, Izd-vo Tomsk. un-ta Publ., 2006. 168 p.
3. Ajtai M., Chvátal V., Newborn M.M., Szemerédi E. Crossing-free subgraphs. *Annals Discrete Math.*, 1982, no. 12, pp. 9–12.
4. De Floriani L., Magillo P., Puppo E. Compressing Triangulated Irregular Networks. *Geoinformatica*, 2000, vol. 1, no. 4, pp. 67–88.
5. Delaunay B.N. Sur la sphere vide. A la memoire de Georges Voronoi. *Izv. AN SSSR* [Izvestiya: Mathematics], 1934, no. 6, pp. 793–800.
6. Denny M.O., Sohler C.A. Encoding a triangulation as a permutation of its point set. *Proc. 9th Canadian Conf. on Computational Geometry*, 1997, pp. 39–43.
7. Pólya G. On picture-writing. *The American Mathematical Monthly*, 1956, no. 63, pp. 689–697.
8. Santos F., Seidel R. A better upper bound on the number of triangulations of a planar point set. *J. Combinat. Theory, Ser. A.*, 2003, no. 102, pp. 186–193.
9. Seidel R. On the number of triangulations of planar point sets. *Combinatorica*, 1998, no. 18, pp. 297–299.
10. Sharir M., Welzl E. Random triangulations of planar point sets. *Proc. 22nd Ann. ACM Symp. on Computational Geometry*, 2006, pp. 273–281.
11. Smith W.S. *Studies in Computational Geometry Motivated by Mesh Generation: Ph.D. Thesis*. Princeton University, 1989. 488 p.

CHAINS METHOD FOR STORAGE OF MULTIDIMENSIONAL TRIANGULATION

Klyachin Vladimir Aleksandrovich

Doctor of Physical and Mathematical Sciences,
Head of Department of Computer Science and Experimental Mathematics
Volgograd State University
klchnv@mail.ru
Prospect Universitetsky, 100, 400062 Volgograd, Russian Federation

Попов Владимир Валентинович

Candidate of Physical and Mathematical Sciences,
Associate Professor, Department of Computer Science and Experimental Mathematics
Volgograd State University
popov_v_v@rambler.ru
Prospect Universitetsky, 100, 400062 Volgograd, Russian Federation

Abstract. This study is an attempt to obtain lower bounds for the amount of memory required to store the data of multi-dimensional triangulations. The author offers a method of organizing storage triangulation, allowing in $n + 1$ times to reduce the amount of memory in comparison with one of the standard methods. The essence of the method proposed by the author is the partitioning of the set of simplices in an array of chains adjacent simplices. Neighborhood relations can be written in a more compact structure of the triangulation. More accurately. If $\mu(f)$ denotes the number of bits occupied by storage triangulation N points in \mathbb{R}^n by one of the standard ways, then $\lim_{N \rightarrow \infty} \frac{\mu(f)}{m \log_2 N} = n + 1$, where m — number of simplexes. Whereas the proposed method of storage triangulation this limit is 1 for any dimension. To investigate the optimal method, the author formalises the problem and introduces the concept of triangulation as a way to store one-to-one mapping of the set of all triangulations of a set of points in \mathbb{R}^n on the set of natural numbers \mathbb{N} . Thus, the method of storage — a comparison of each triangulation identify the unique natural number. Study the accuracy of the asymptotic estimates leads to the problem of obtaining lower triangulations given set of points. This article discusses two examples where such a calculation is accurate. In both cases, there is the exponential growth for number of triangulations depended on the number of points. Reference to recent works by foreign mathematicians say about general behavior of the number of such triangulations in the plane. This allows the author to hypothesize the existence of such a storage method of triangulation, in which the memory capacity increases linearly with the number of points. In particular, we expect that the optimal value of the above limit is 0.

Key words: triangulation, simplex, memory volume estimate, number of triangulations, Catalan number.